City Research Online

City, University of London Institutional Repository

Citation: Bleakley, G.J. (1999). A methodology for the design of safety critical mechatronics. (Unpublished Doctoral thesis, City, University of London)

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/30510/

Link to published version:

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online: http://openaccess.city.ac.uk/ publications@city.ac.uk

A METHODOLOGY FOR THE DESIGN OF SAFETY CRITICAL MECHATRONICS

By

Graham John Bleakley B.Eng. Hon's

Submitted for

The Degree of Doctor of Philosophy

City University

Department of Civil Engineering

City University

London, EC1V OHB

September 1999

CONTENTS

CONTENTS 1

TABLE OF FIGURES 8

TABLE OF TABLES 13

ACKNOWLEDGEMENTS 15

ABSTRACT 16

1 THE DESIGN OF SAFETY CRITICAL MECHATRONICS: INTRODUCTION 17

1.1 BACKGROUND REVIEW 19

1.1.1 Engineering Design Methods *19*

1.1.1.1 Descriptive methods 19

1.1.1.2 Why a more structured and systematic approach is necessary 21

1.1.1.3 Prescriptive methods 22

1.1.1.4 Design Methods and Methodologies 26

1.1.1.5 Concurrent Design Methods 27

1.1.1.5.1 Integrated Product Development 29

1.1.1.5.2 Design Function deployment 29

1.1.1 Review of current mechatronics design methods *32*

1.1.1.1 Why a specific method is required for Mechatronics design ? 33

1.1.1.2 The laws and axioms of Buur 35

1.1.1.3 The Finnish Approach, 'Metamethodics' 37

1.1.1.4 Design Function Deployment (DFD) and Mechatronics 38

1.1.2 Review Summary and conclusions *40*

1.2 RESEARCH HYPOTHESES, OBJECTIVES AND OUTCOMES 42

1.2.1 Research Hypotheses *43*

1.2.2 Research objectives *43*

1.2.3 Scope of Research *43*

1.3 NOVELTY OF WORK 44

1.3.1 The integration of new computerised tools into EQFD *45*

1.3.2 Integration of combined design tools in QFD methods *45*

1.4 SCOPE OF WORK 47

1.4.1 Research into design tools and methods *47*

1.4.2 Application of the proposed methodology to a case study *48*

1.5 THESIS STRUCTURE 48

2 REVIEW OF DESIGN METHODS AND TOOLS 50

2.1 INTRODUCTION 50

2.2 FUNCTION BASED DESIGN TOOLS 50
2.2.1 Function trees, the law of vertical causality and function coupling *51*
2.2.2 Theory of domains *53*
2.2.3 Morphological charts *55*
2.2.4 Benchmarking techniques *55*
2.2.4.1 Pugh's concept selection 56
2.2.4.2 Weighted objectives method 57
2.3 INTRODUCTION TO QFD FAMILY OF METHODS 57
2.3.1 Quality Function Deployment (QFD) *58*
2.3.2 QFD House of Quality *58*
2.3.3 Macro procedure for basic QFD *61*
2.3.4 Enhanced Quality Function Deployment (EQFD) *62*
2.3.5 Requirements and Failure Causes, Analysis (RFCA) *64*
2.3.6 Design Function Deployment (DFD) *67*
2.4 SUMMARY OF REVIEW AND CONCLUSIONS 69

3 SYSTEMS MODELLING TECHNIQUES 72

3.1 INTRODUCTION 72
3.2 SOFTWARE TOOLS FOR SYSTEM DESIGN 74
3.3 SYSTEM MODELS 75
3.3.1 Function based modelling *75*
3.3.1.1 ACP diagrams and MASCOT 75
3.3.1.2 Structured Analysis and Design Technique (SADT) 77
3.3.1.3 Jackson Structured Programming 80
3.3.2 Object Orientated methods *81*
3.4 BEHAVIOURAL MODEL: LOGICAL VIEW 82
3.4.1 Finite state models *82*
3.4.2 Petri nets *84*
3.4.3 flow charts *85*
3.4.4 Textual Behavioural Models *86*
3.5 BEHAVIOURAL MODELLING: ENERGY BASED VIEW 88
3.5.1 Control Approach *89*
3.5.2 Bond Graph Approach *91*
3.6 SUMMARY OF MODELLING TECHNIQUES 93
3.7 REVIEW OF METHODS FOR COMPUTERISATION 95
3.7.1 Object orientated methods *95*
3.7.2 Expert System shells *95*
3.7.3 Graphical modeling tools *97*
3.7.4 Schemebuilder *98*
3.8 SUMMARY FOR COMPUTER REPRESENTATION METHODS 98

4 FAILURE ANALYSIS: TECHNIQUES AND COMPUTER TOOLS 100

4.1 INTRODUCTION 100
4.2 INTRODUCTION TO FAILURE ANALYSIS TOOLS 102
4.3 QUALITATIVE FAILURE ANALYSIS 102
4.3.1 Fault Tree Analysis (FTA) 103
4.3.2 Failure Modes Effects Criticality Analysis (FMECA) 104
4.3.3 Event Tree Analysis (ETA) 104
4.3.4 HAZard and Operability Analysis (HAZOP) 106
4.3.5 Consequence Led Analysis of Safety Hazards (CLASH) 107
4.4 COMPUTER BASED FAILURE ANALYSIS TOOLS 107
4.4.1 ES failure analysis models 108
4.4.2 Formal Methods for failure analysis 110
4.4.3 Dynamic, computer based modelling 111
4.5 CONCLUSIONS AND SUMMARY 113

5 DESIGN METHODOLOGY OVERVIEW 116

5.1 INTRODUCTION 116
5.2 METHODOLOGY OVERVIEW 116
5.3 INITIAL RISK ANALYSIS (CLASH) 120
5.4 CURRENT MARKET ANALYSIS 122
5.5 CUSTOMER REQUIREMENTS AND FUNCTION ANALYSIS 124
5.6 CONCEPT SELECTION 126
5.6.1 Function Mean's Trees 126
5.6.2 2nd Stage EQFD 127
5.6.3 Morphological charts 128
5.6.4 Benchmarking 129
5.7 SYSTEMS MODELLING 130
5.7.1 ACP diagrams 130
5.7.2 Constructing an ACP Diagram 130
5.7.3 ACP diagrams and the roof of the House of Quality 133
5.7.4 ACP and VISSIM 135
5.8 COMPUTER MODELS FOR COMPONENT SELECTION 138
5.8.1 System analysis and ACP diagrams 139
5.8.2 The use of VISSIM for component selection 142
5.8.3 Results 144
5.9 FAILURE MODELLING AND ANALYSIS USING VISSIM 145
5.9.1 System behaviour and modelling 146
5.9.2 Failure Mode identification and VISSIM modelling 147
5.9.3 Results and Analysis 150

5.10 SUMMARY 152

6 CASE STUDY, JUSTIFICATION, SPECIFICATION AND SCOPE OF WORK 154

6.1 CASE STUDY JUSTIFICATION 154
6.2 TASK DESCRIPTION AND SYSTEM SPECIFICATION 155
6.3 SAFETY SPECIFICATION 157
6.4 SCOPE OF THE CASE STUDY 158
6.5 SUMMARY 158

7 DEFINING CUSTOMER REQUIREMENTS AND SELECTING CONCEPTS: CLASH LED EQFD 160

7.1 INTRODUCTION 160
7.1 THE USE OF QFD FOR INITIAL CONCEPT SELECTION 160
7.1.1 Selection of Working Platform Concept 161
7.1.1.1 Conventional platforms 161
7.1.1.2 Climbing robots 162
7.1.1.3 Non- Climbing Robots 164
7.1.1.4 QFD Assessment of Platforms 166
7.1.2 Selection of robot configuration for collimator handling 168
7.1.2.1 Task Models for Gamma Radiography Robot 168
7.1.2.2 Relative Robot Performances 170
7.2 SAFETY BASED CUSTOMER REQUIREMENTS AND GENERAL SOLUTIONS 172
7.2.1 CLASH/FTA analysis 173
7.2.2 Initial FMEA 175
7.2.3 Summary of CLASH, FTA and FMEA 176
7.3 INTEGRATION OF GENERAL DESIGN TOOLS INTO EQFD 179
7.3.1 1st Stage EQFD chart and top level functional analysis 179
7.3.2 2nd Stage EQFD charts and sub function analysis 182
7.3.2.1 Development of complete concept solutions 185
7.3.2.2 Final concept selection 189
7.3.3 Mapping the design process 191
7.4 SUMMARY AND CONCLUSIONS 193

8 EMBODIMENT DESIGN: SOFTWARE MODELS FOR COMPONENT SELECTION 195

8.1 INTRODUCTION 195
8.2 SYSTEM SPECIFICATION 196
8.3 CORRELATION CHAINS AND ACP DIAGRAMS 197
8.3.1 The correlation chain 197
8.3.2 ACP systems model 199
8.3.3 Behavioural model 200
8.3.3.1 Motor behaviour 200

8.3.3.2 Gear box behaviour 205
8.3.3.3 Pulley and belt behaviour 208
8.3.3.4 Load and Bearing behaviour 210
8.4 DEVELOPMENT OF THE VISSIM MODEL FOR COMPONENT SELECTION 211
8.4.1 Creating the component databases 211
8.4.2 Creating the component combination file 214
8.4.3 Creating the VISSIM model of the drive system 217
8.4.3.1 Motor model 218
8.4.3.2 Gear box and Pulley model 220
8.4.3.3 Belt, Load and Pulley model 222
8.4.4 Combination selection and logging 223
8.4.4.1 Output analysis 223
8.4.4.2 Decision and Data Filter Blocks 224
8.4.4.3 Combination File 225
8.5 COMPUTERISED COMPONENT SELECTION 226
8.5.1 Input conditions 227
8.5.2 Results 228
8.5.2.1 Component Selection results 228
8.6 SUMMARY AND CONCLUSIONS 230

9 SOFTWARE MODELS FOR FAULT ANALYSIS 233

7.1 INTRODUCTION 233
9.1 VISSIM MODELS FOR FAILURE ANALYSIS 234
9.1.1 Upgrading the VISSIM model 235
9.1.2 Failure modelling within the VISSIM model 238
9.1.2.1 Failure Mode Analysis 238
9.1.2.2 Modelling perfect and failure behaviour 243
9.1.2.3 Generating the failure combination file 245
9.1.3 Operating sequence 245
9.1.4 Modelling of Failure criteria and data logging 246
9.2 FAILURE MODELS FOR FAULT ANALYSIS 249
9.2.1 1st Generation Model 249
9.2.2 2nd Generation Model 253
9.2.3 3rd Generation Model 256
9.2.4 4th Generation Model 258
9.2.4.1 Generating the FMEA chart 259
9.2.4.2 Morphological Analysis and Solution Selection 263
9.2.4.3 Failure Analysis Of Additional Components 270
9.2.4.4 Results from 4th Generation Model 272
9.3 SUMMARY AND CONCLUSIONS 276
9.3.1 Comparison of Results 278

9.3.2 Conclusions ... 279

10 SUMMARY OF RESULTS AND CONCLUSIONS ... **282**

10.1 INTRODUCTION ... 282

10.2 ASSESSMENT OF PRO-ACTIVE SAFETY LED DESIGN FOR MECHATRONICS ... 282

10.2.1 Method 1: DFD/ACP and intuitive FMEA, FTA ... 282

10.2.2 Method 2: CLASH led EQFD with computerised fault analysis ... 286

10.2.3 Comparison of methods 1 and 2 ... 289

10.3 RESULTS RELATED TO HYPOTHESES AND OBJECTIVES ... 291

10.3.1 Hypothesis 1: Integration of failure analysis and design methods ... 292

10.3.1.1 Benefits of applying CLASH led EQFD ... 294

10.3.2 hypothesis 2: Integration of computer tools with the design process ... 295

10.3.2.1 The benefits of computerised component selection ... 295

10.3.2.2 The benefits of computerised failure analysis ... 296

10.3.3 Cautionary observations ... 297

10.4 NOVELTY OF WORK ... 299

10.4.1 The integration of new computerised tools into EQFD. ... 299

10.4.2 Integration of combined design tools into EQFD ... 300

10.5 RECOMMENDATIONS FOR FUTURE WORK ... 301

REFERENCES AND BIBLIOGRAPHY ... **303**

APPENDIX A : GLOSSARY OF TERMS ... **321**

APPENDIX B: RESULT FILES GENERATED BY VISSIM MODELS FOR COMPUTERISED COMPONENT SELECTION AND FAILURE ANALYSIS ... **325**

SUCCESSFUL COMBINATION FILES FOR COMPONENT SELECTION ... 325

Successful combinations for X axis component selection. ... 326

Successful combinations for Y axis component selection. ... 328

FAILED COMPONENT COMBINATION DETAILS FOR 1ST, 2ND AND 3RD STAGE MODELS FOR THE X AND Y AXIS ... 330

Logged failed combinations for X axis component selection ... 332

Logged failed combinations for Y axis component selection ... 338

RESULTS OF VISSIM MODELS FOR FAILURE ANALYSIS ... 344

First Generation failure model ... 345

Second Generation failure model ... 346

Third Generation failure model ... 346

Fourth Generation failure model ... 348

APPENDIX C: VISSIM MODELS AND SUPPORTING FILES ... **350**

APPENDIX D: PAPERS AND REPORTS ... **356**

JOURNAL PAPERS 356
CONFERENCE PAPERS 356
RESEARCH REPORTS 357
PAPERS IN PREPARATION 357

TABLE OF FIGURES

FIGURE 1-1 DESCRIPTIVE MODEL OF DESIGN, (CROSS[89]) 19

FIGURE 1-2 THE FRENCH MODEL OF THE DESIGN PROCESS (FRENCH[71]) 20

FIGURE 1-3 DESIGN ACCORDING TO JONES[63] 22

FIGURE 1-4 DESIGN ACCORDING TO ARCHER[84] 23

FIGURE 1-5 PAHL & BEITZ'S DESIGN MODEL (PAHL & BEITZ[84]) 24

FIGURE 1-6 DESIGN ACCORDING TO VDI 2221 25

FIGURE 1-7 OVER THE WALL APPROACH TO ENGINEERING (EVBUOMWAN[95]) 28

FIGURE 1-8 COMPARISON OF TIME SCALES FOR TRADITIONAL ENGINEERING AND CONCURRENT ENGINEERING (BURR[90A]) 28

FIGURE 1-9 THE BUILDING BLOCKS OF INTEGRATED PRODUCT DEVELOPMENT (ANDREASAN[87]) 29

FIGURE 1-10 STAGES IN THE DESIGN FUNCTION DEPLOYMENT, DESIGN LIFE CYCLE 30

FIGURE 1-11 TOP LEVEL DFD CHART FOR AUTOMATED GAMMA RADIOGRAPHY ROBOT 32

FIGURE 1-12 THE POSITION OF PRODUCTS WITHIN THE MECHATRONICS DOMAIN 33

FIGURE 1-13 DOMAIN INTERACTION SPHERES FOR MECHATRONICS 35

FIGURE 1-14 FINITE STATE MACHINE MODEL OF A TELEPHONE (BUUR[90]) 36

FIGURE 1-15 SA/SD, CONTEXT DIAGRAM FOR A ROBOT GRIPPER (SALMINEM[90]) 37

FIGURE 1-16 SA/SD STATE TRANSITION DIAGRAM, FOR ROBOT GRIPPER, (SALMINEM[90]) 38

FIGURE 1-17 THE INTEGRATION OF FMEA AND FTA WITH DFD 39

FIGURE 1-18 THE INTERACTION OF DFD AND ACP THROUGH THE USE OF THE CORRELATION MATRIX (BLEAKLEY[97]) 40

FIGURE 2-1 FUNCTION STRUCTURE OF A WASHING MACHINE 51

FIGURE 2-2 FUNCTION MEANS TREE FOR AN AUTOMATIC TEA MAKER (TJALVE[1979]) 52

FIGURE 2-3 THE THEORY OF DOMAINS 54

FIGURE 2-4 TABULAR, MORPHOLOGICAL CHART, FOR A FORKLIFT TRUCK (CROSS[1989]) 55

FIGURE 2-5 PUGH'S CONCEPT SELECTION TABLE (PUGH[1981]) 56

FIGURE 2-6 WEIGHTED OBJECTIVES CHART (CROSS[1989]) 57

FIGURE 2-7 THE QFD HOUSE OF QUALITY (SONTOW[1993]) 60

FIGURE 2-8 THE QFD PROCESS (SONTOW[1993]) 61

FIGURE 2-9 EQFD PROCESS DIAGRAM (SONTOW(93)) 63

FIGURE 2-10 AMPLIFICATION SUB-TREE (PANDEY[1992]) 65

FIGURE 2-11 QUALITY LOSS FUNCTION (PHADKE[89]) 66

FIGURE 2-12 DFD DESIGN PROCESS (EVUOBUOMWAN(95)) 68

FIGURE 2-13 DFD THREE LAYER MODEL OF DESIGN TOOLS (SIVALOGANATHAN(95)) 69

FIGURE 3-1 SCHEMATIC OF HYDRAULIC RAM SYSTEM 73

FIGURE 3-2 TOP AND SECOND LEVEL SYSTEM DIAGRAM OF HYDRAULIC RAM SYSTEM 73

FIGURE 3-3 KEY FOR ACP DIAGRAMS 76

FIGURE 3-4 ACP MODEL OF COMPUTER CONTROLLED CHEMICAL PLANT (BENNET (90)) 77
FIGURE 3-5 SADT & DATA FLOW DIAGRAM SYMBOLOGY 77
FIGURE 3-6A CRUISE CONTROL SYSTEM, CONTEXT DIAGRAM (LAPLANTE(93)) 79
FIGURE 3-7 JSD SOFTWARE MODEL OF ELEVATOR CONTROLLER (LAPLANTE(93)) 80
FIGURE 3-8 FINITE STATE MACHINE REPRESENTATION OF AN EVEN PARITY CHECKER 82
FIGURE 3-9 STATE TRANSITION DIAGRAM OF A VARIABLE LIGHT LAMP (WARD(85)) 83
FIGURE 3-10 EXAMPLE OF A HAREL STATE CHART(LAPLANTE[93]) 84
FIGURE 3-11 PETRI-NET MODEL OF MACHINE SHOPORDER PROCESSING (PETERSEN[81]) 85
FIGURE 3-12 EXAMPLE OF A SIMPLE FLOWCHART (LAPLANTE[93]) 86
FIGURE 3-13 EXAMPLE OF PSEUDO CODE (FROM LAPLANTE[93]) 87
FIGURE 3-14 SECTION OF Z CODE DEFINING AN OPERATION (POTTER[91]) 88
FIGURE 3-15 SCHEMATIC OF A MOTOR SPEED CONTROLLER (GOLTEN[92]) 89
FIGURE 3-16 BLOCK DIAGRAM IN THE S DOMAIN FOR THE MOTOR SPEED CONTROLLER (GOLTEN[92]) ... 89
FIGURE 3-17 TIME RESPONSE OUTPUT OF A 1ST ORDER CONTROL SYSTEM TO A STEP INPUT 90
FIGURE 3-18 SCHEMATIC OF AN AIR PUMP (ATHERTON[98]) 91
FIGURE 3-19 BOND GRAPH MODEL OF AN AIR PUMP (ATHERTON[98]) 91
FIGURE 3-20 BOND GRAPH MODEL OF AN AIR PUMP, INCORPORATING SYSTEM LOSSES(ATHERTON[98])92
FIGURE 3-21 ACP MODEL OF ON/ EMERGENCY STOP LATCHED RELAY CIRCUIT 96
FIGURE 3-22 VISSIM MODELS OF THE ACP DIAGRAM IN FIGURE 3-21 97
FIGURE 4-1 DESIGN FOR SAFETY ACCORDING TO BSI EN 292 PART 1 100
FIGURE 4-2 DESIGN FOR SAFETY ACCORDING TO THE GUIDE FOR PES'S (HSE[93]) 101
FIGURE 4-3A TYPICAL FAULT TREE 103
FIGURE 4-3B RFT OF FIGURE 4-3A 103
FIGURE 4-4 EVENT TREE ANALYSIS FOR GAS ESCAPE HAZARD (BENTLEY[99]) 106
FIGURE 4-5 MODEL-OBSERVER CONTROL SYSTEM 114
FIGURE 4-6 SCHEMATIC OF SOFTWARE IMPLEMENTATION OF FAILURE MODE DETECTION 114
FIGURE 5-1 CLASH LED EQFD WITH INTEGRATED FAILURE ANALYSIS 117
FIGURE 5-2 SCHEMATIC FOR APPROACH TO AUTOMATIC COMPONENT SELECTION 119
FIGURE 5-3 SCHEMATIC OF SOFTWARE IMPLEMENTATION OF FAILURE MODE DETECTION 120
FIGURE 5-4 EXAMPLE OF CLASH ANALYSIS AS APPLIED TO A WOOD SPLITTER 121
FIGURE 5-5 1ST STAGE QFD ANALYSIS OF COMPETING ELECTRONIC THERMOMETERS 122
FIGURE 5-6 TOP LEVEL EQFD CHART FOR GAMMA RADIOGRAPHY ROBOT 125
FIGURE 5-7 FUNCTION MEANS TREE FOR "TRANSMIT SPOKEN MESSAGE OVER A DISTANCE" 127
FIGURE 5-8 STAGE 2 EQFD CHART 127
FIGURE 5-9 MORPHOLOGICAL CHART FOR "TRANSMIT SPOKEN MESSAGE OVER A DISTANCE" 128
FIGURE 5-10 EXAMPLE OF TOP LEVEL ACP DIAGRAM (MOTOR SPEED CONTROL SYSTEM) 131
FIGURE 5-11 BREAK DOWN OF PSU ACTIVITY 131
FIGURE 5-12 EQFD STAGE 2 CHART SHOWING THE INTERACTION MATRIX 134
FIGURE 5-13 TOP LEVEL ACP DIAGRAM FOR MOTOR DRIVE SYSTEM 134

FIGURE 5-14 MOTOR ACTIVITY SYSTEM 135
FIGURE 5-15 MOTOR ACTIVITY 135
FIGURE 5-16 VISSIM DIAGRAM OF MOTOR CONTROLLER 136
FIGURE 5-17 VISSIM MODEL OF PSU ACTIVITY 137
FIGURE 5-18 VISSIM MODEL OF ON/OFF SWITCH POOL 137
FIGURE 5-19 VISSIM MODEL OF +/-15 V PSU POOL 138
FIGURE 5-20 SCHEMATIC SOLUTION FOR LIFTING PROBLEM 139
FIGURE 5-21 2ND STAGE EQFD MATRIX FOR LIFTING PROBLEM 140
FIGURE 5-22 ACP MODEL OF LIFTING PROBLEM 140
FIGURE 5-23 VISSIM MODEL FOR COMPONENT SELECTION FOR LIFTING PROBLEM 142
FIGURE 5-24 PISTON ACTIVITY 142
FIGURE 5-25 COMPONENT DATABASE BLOCK 143
FIGURE 5-26 SOLUTION LOGGING BLOCK 143
FIGURE 5-27 VISSIM MODEL FOR LATCHED ON, EMERGENCY STOP OFF SWITCH 146
FIGURE 5-28 ACP DIAGRAM FOR LATCHED ON, EMERGECY STOP OFF, SWITCH 146
FIGURE 5-29 LATCHED RELAY ACTIVITY 147
FIGURE 5-30 FAILURE MODEL FOR ON SWITCH ACTIVITY 149
FIGURE 5-31 COMPLETE VISSIM MODEL FOR FAILURE SIMULATION AND ANALYSIS 149
FIGURE 5-32 REDUCED FAULT TREE ON/OFF SWITCH 152
FIGURE 7-1 SOUTH BANK CLIMBING ROBOT 162
FIGURE 7-2 MAGNETIC CRAWLER ROBOT 163
FIGURE 7-3 ROBUG II (COLLIE[90]) 164
FIGURE 7-4 FLYING ROBOT (NISHI[93]) 165
FIGURE 7-5 WORKING AREA OF CARTESIAN ROBOT 169
FIGURE 7-6 WORKING AREA OF PUMA ROBOT 169
FIGURE 7-7 IDEALIZED JOINT SPECIFICATIONS FOR PUMA AND CARTESIAN ROBOTS 170
FIGURE 7-8 TIME AGAINST ACCELERATION FOR BOTH TASKS, AND ROBOTS 171
FIGURE 7-9 COMPARISON OF ASYNCHRONOUS TO SYNCHRONOUS MOVES FOR TASK 2 171
FIGURE 7-10 CLASH TREE FOR GAMMA RADIOGRAPHY ROBOT 174
FIGURE 7-11 EXPANDED CLASH TREE/FTA FOR GAMMA RADIOGRAPHY ROBOT 175
FIGURE 7-12 TOP LEVEL EQFD CHART 180
FIGURE 7-13 FIRST STAGE EQFD CHART 181
FIGURE 7-14 STAGE 2 EQFD CHART 184
IGURE 7-15 EXPANDED STAGE 2 EQFD CHART 186
FIGURE 7-16 STAGE 2 EQFD/ WEIGHTED OBJECTS CHART FOR CONCEPT SELECTION 190
FIGURE 7-17 FUNCTION MEANS TREE FOR X AND Y AXIS 192
FIGURE 8-1 CORRELATION CHAIN FOR BELT PULLEY DRIVE 198
FIGURE 8-2 ACP DIAGRAM OF COMPLETE DRIVE AXIS DRIVE SYSTEM 200
FIGURE 8-3 ARMATURE CIRCUIT FOR A DC MOTOR 201

FIGURE 8-4 STANDARD CLTF BLOCK DIAGRAM 203
FIGURE 8-5 CLTF OF AN ELECTRIC MOTOR 203
FIGURE 8-6 MECHANICAL TRANSFER FUNCTION OF A MOTOR 204
FIGURE 8-7 ACP DIAGRAM OF MOTOR 204
FIGURE 8-8 SCHEMATIC OF MOTOR, GEARBOX AND LOAD 205
FIGURE 8-9 ACP DIAGRAM FOR MOTOR 207
FIGURE 8-10 ACP DIAGRAM FOR GEAR BOX 208
FIGURE 8-11 ACP DIAGRAM FOR LOAD AND BEARING 210
FIGURE 8-12 PULLEY DIMENSION NOTATION 214
FIGURE 8-13 EFFECT OF INT BLOCK ON A RAMPED SIGNAL 215
FIGURE 8-14 COMBINATION LOGGING TO SWEEP.MAP 215
FIGURE 8-15 SECTION OF SWEEP.MAP 216
FIGURE 8-16 VISSIM MODEL OF DRIVE SYSTEM 217
FIGURE 8-17 VISSIM MODEL OF AN ELECTRIC MOTOR 218
FIGURE 8-18 ELECTRICAL TRANSFER FUNCTION (1/(LAS + RA)) 219
FIGURE 8-19 MECHANICAL TRANSFER FUNCTION (1/(JL + JMS)) 220
FIGURE 8-20 GEARBOX MODEL 221
FIGURE 8-21 PULLEY MODEL 221
FIGURE 8-22 BELT MODEL 222
FIGURE 8-23 LOAD MODEL 222
FIGURE 8-24 OUTPUT ANALYSIS BLOCK 224
FIGURE 8-25 DECISION AND DATA FILTER BLOCKS 225
FIGURE 8-26 SECTION FROM A SUCCESSFUL COMBINATION FILE 226
FIGURE 8-27 VISSIM MODEL FOR COMPUTERISED COMPONENT SELECTION 227
FIGURE 9-1 SCHEMATIC OF VISSIM MODEL FOR FAILURE ANLAYIS 234
FIGURE 9-2 VISSIM MODEL X AXIS DRIVE SYSTEM MODEL 235
FIGURE 9-3 FUNCTIONAL DECOMPOSITION OF THE COMPUTER BLOCK 236
FIGURE 9-4 SERVO UNIT BLOCK 237
FIGURE 9-5 TACHOMETER FAILURE MODEL 243
FIGURE 9-6 TACHOMETER IDEAL BEHAVIOUR MODEL 244
FIGURE 9-7TACHOMETER INTERMITTENT FAILURE MODEL 244
FIGURE 9-8 FAILURE CRITERIA BLOCK 247
FIGURE 9-9 FAILURE LOGGING BLOCK 248
FIGURE 9-10 THE EFFECT OF A BREADTH FIRST SEARCH ON THE FAILURE SPACE 250
FIGURE 9-11 SECTION OF SWFILT2D.VSM SHOWING FAILURE MODE SELECTION FOR TRIPLE POINT FAILURES 251
FIGURE 9-12 RFT FOR MODEL 1, FC1 252
FIGURE 9-13 RFT FOR MODEL 1, FC2 252
FIGURE 9-14 INTERRUPT BASED FAILURE CHECKING 253

FIGURE 9-15 VISSIM MODELS FOR STAGED STARTUP 254
IGURE 9-16 RFT FOR MODEL 2, FC1 255
FIGURE 9-17 RFT FOR MODEL 2, FC2 255
FIGURE 9-18 RFT FOR MODEL 3, FC1 257
FIGURE 9-19 RFT FOR MODEL 2, FC2 258
FIGURE 9-20 OVERVIEW OF VISSIM 4TH GENERATION MODEL (STAGES4A.VSM) 272
FIGURE 9-21 RFT FOR MODEL 4, FC1 274
FIGURE 9-22 RFTS FOR MODEL 4, FC2 275
FIGURE 10-1 DESIGN METHODOLOGY USED FOR PILOT STUDY 283
FIGURE 10-2 INCOMPLETE STAGE 2A DFD CHART 284
FIGURE 10-3 CLASH LED EQFD WITH COMPUTERISED ANALYSIS FEATURES 287

TABLE OF TABLES

TABLE 1-1 THE INCOMPATIBILITY OF MECHATRONIC DESIGN DOMAINS (ULLMAN[92]) 34
TABLE 1-2 SUMMARY COMPARISON OF DESIGN METHODS FOR MECHATRONICS 41
TABLE 3-1 LOGICAL AND BEHAVIOURAL MODEL OF HYDRAULIC RAM 74
TABLE 4-1 EXAMPLE OF FMEA CHART FOR THE SAFE DESIGN OF AN REMOTE GAMMA RADIOGRAPHY CONTROL SYSTEM 105
TABLE 5-1CROSS'S WEIGHTED OBJECTIVES CHART 129
TABLE 5-2 SOLUTIONS TO LIFTING PROBLEM 144
TABLE 5-3 FMEA TABLE FOR HARDWIRED ON/OFF SWITCH WITH LATCHED RELAY 148
TABLE 5-4 LOGGED FAILURE COMBINATIONS FOR LATCHED, ON/OFF SWITCH 151
TABLE 5-5 SPF FAILURES FOR ON/OFF SWITCH 151
TABLE 6-1 GAMMA RADIOGRAPHY EQUIPMENT SPECIFICATION. 156
TABLE 7-1QFD ANALYSIS OF WORKING PLATFORMS FOR REMOTE GAMMA RRADIOGRAPHY 166
TABLE 7-7-2 OVERALL SPECIFICATIONS: PUMA AND CARTESIAN ROBOT 170
TABLE 7-3PROACTIVE FMEA CHART FOR GAMMA RADIOGRAPHY ROBOT (CONTINUED OVERLEAF)..... 177
TABLE 7-4 MORPHOLOGICAL CHART FOR Y AND X AXIS CONCEPT GENERATION 187
TABLE 7-5 MORPHOLOGICAL CHART FOR GEAR TRAIN DRIVE 188
TABLE 7-6 MORPHOLOGICAL CHART FOR SCREW DRIVE 188
TABLE 7-7 MORPHOLOGICAL CHART FOR PULLEY BELT DRIVE 189
TABLE 8-1 X AND Y AXIS SPECIFICATION 196
TABLE 8-2 MOTOR SPECIFICATIONS FOR VICKERS FAS T RANGE 212
TABLE 8-3 PULLEY SPECIFICATIONS 212
TABLE 8-4 STEGMAN PLANETARY GEAR BOX SPECIFICATIONS 212
TABLE 8-5 VISSIM MODEL FOR COMPUTERISED COMPONENT SELECTION 229
TABLE 8-6 VISSIM MODE FOR COMPUTERISED COMPONENT SELECTION 230
TABLE 9-1 FAILURE ANALYSIS FOR 1ST GENERATION MODEL (CONTINUED OVERLEAF) 238
TABLE 9-2 REDUCED FAILURE MODE BEHAVIOUR FOR 1ST GENERATION MODEL 242
TABLE 9-3 SECTION OF FAILURE COMBINATION FILE 245
TABLE 9-4 FMEA CHART FOR 2ND GENERATION MODEL FAILURES (CONTINUED ON THE NEXT 2 PAGES)260
TABLE 9-5 MORPHOLOGICAL ANALYSIS FOR THE DETERMINATION OF FAILURE SOLUTIONS (CONTINUED ON NEXT PAGE.) 265
TABLE 9-6 EQFD AND WEIGHTED OBJECTIVES CHART FOR FMEA ANALYSIS (CONTINUED BELOW)..... 268
TABLE 9-7 FAILURE MODE AND BEHAVIOURAL ANALYSIS 271
TABLE 9-8 COMPONENT AND FAILURE MODE IDENTIFICATION 273
TABLE 9-9 SUMMARY OF FAILURE ANALYSIS RESULTS 278
TABLE 10-1 SUMMARY OF RESULTS FOR COMPUTERISED FAILURE ANALYSIS 293

Table Appendix B-1 Contents of STAGEX1.MAP..... ..326

Table Appendix B- 2Contents of STAGEX2.MAP..... ..327

Table Appendix B- 3 Contents of STAGEX3.MAP 327

Table Appendix B- 4 Contents of STAGEY1.MAP..... ..328

Table Appendix B- 5 Contents of STAGEY2.MAP..... ..329

Table Appendix B- 6 Contents of STAGEY3.MAP..... ..329

Table Appendix B- 7 Column designation for Stage 1 result files.....330

Table Appendix B- 8 Column designation for stage 2 and 3 result files..... ..331

Table Appendix B- 9 Contents of FAIL1X.DAT..... ..332

Table Appendix B- 10 Contents of FAIL2X.DAT..... ..336

Table Appendix B- 11 Contents of FAIL3X.DAT 337

Table Appendix B- 12 Contents of FAIL1Y.DAT..... ..338

Table Appendix B- 13 Contents of FAIL2Y.DAT..... ..342

Table Appendix B- 14 Contents of FAIL3Y.DAT..... ..343

Table Appendix B- 15 Column position against component/function 1st, 2nd & 3rd models 344

Table Appendix B- 16, FS114.DAT , SPF, Model 1, FC1..... ..345

Table Appendix B- 17, FC116.DAT SPF, Model 1, FC2.....345

Table Appendix B- 18, FC126.DAT DPF, Model 1, FC2..... ..345

Table Appendix B- 19, FC214.DAT, SPF, Model 2, FC1..... ..346

Table Appendix B- 20, FC2244.DAT, DPF, Model 2, FC1..... ..346

Table Appendix B- 21, FC216.DAT, SPF, Model 2, FC2..... ..346

Table Appendix B- 22, FC314.DAT, SPF, Model 3, FC1..... ..346

Table Appendix B- 23, FC3244.DAT, DPF, Model 3, FC1..... ..346

Table Appendix B- 24, FC316.DAT, SPF, Model 3, FC1..... ..347

Table Appendix B- 25, FC3266.DAT, DPF, Model 3, FC2..... ..347

Table Appendix B- 26 Column position against component/function 4th generation model..... ..348

Table Appendix B- 27, FC414.DAT, SPF, Model 4, FC1..... ..349

Table Appendix B- 28, FC43FC1.DAT, TPF, Model 4, FC1..... ..349

Table Appendix B- 29, FC42FC2.DAT, DPF, Model 4, FC2..... 349

Table Appendix B- 30, FC44FC2.DAT, QPF, Model 4, FC2.....349

Acknowledgements

I wish to thank Mr Denis Chamberlain, who provided invaluable support and guidance throughout the completion of this thesis, including the onerous task of reading and editing my work.

I also wish to thank Dr Ronie Navon, who provided the priceless opportunity to study in Israel for six months, giving me access to one of the most extensive libraries I have had the pleasure of using.

Finally, I would like to thank Dr Robert Edney, Dr Mark Atherton, and Professor David Bradley who provided me with a sounding board for my thoughts, with regard to the ideas presented in this thesis.

Abstract

The general approach taken for the design of safety critical, mechatronic products is at present based upon creating a design and then incorporating safety features into the design, almost as an afterthought. Although guidelines exist, that detail basic processes which, should be carried out for design for safety, structured design methods, that could aid design for safety are not described. Additionally, the use of computers, to aid in the decision making process for the design and safety analysis of mechatronics, is underexploited.

In this thesis a structured, safety led, design methodology based around Extended Quality Function Deployment is developed. The basic methodology is integrated with a combination of safety analysis tools, and additional structured design methods which help control the methodology, reducing many of the problems of implementing Extended Quality Function Deployment (EQFD). Additionally, the development of computer based tools for component selection and failure analysis, based upon dynamic, energy based models is shown. These models are fully integrated with the design methodology.

The development of these methods and tools is described in relation to a Case Study, based upon the safety analysis, and conceptual design of a robot capable of carrying out Remote Gamma Radiography of Steel Storage Tanks. Safety analysis tools are applied in a pro-active manner, prior to starting the EQFD analysis, providing guidelines as to the safety requirements for the design. Further safety analysis is carried out during the design process using graphical dynamic computer models of the proposed design. These computer models originally being developed for automated component selection, which are in turn derived from the correlation roof .

This pro-active approach to design for safety led to many useful insights into possible guidelines and features which should be incorporated into the product at an early stage of the design process. Modelling the system using a computer based tool-improved understanding of the system behaviour and helped accelerate the design process. Firstly as an tool for automated component selection and secondly as mechanism, for extensive safety analysis.

1 The Design of Safety Critical Mechatronics: Introduction

The main objective of this research is to develop a design life cycle for conceptual design and modelling of safety critical mechatronic systems. In this, safety is taken as "the freedom of exposure to danger , or the exemption from hurt , injury or loss" (Bowen[93]) and critical as "important" (Collins[88]). A mechatronic system, is defined as "a functional and spatial integration of mechanical and electrical components with information technology", (Buur [90a]). According to these definitions, a safety critical mechatronic system could be defined as:

> A functional and spatial integration of mechanical, electrical and information technology components which are used to carry out a process where the possibility of exposure to danger, hurt, injury or loss is an important feature of the process.

Such systems are used in the offshore, construction and nuclear industries, where the environment is harsh, and the task sensitive and hazardous. The cost of system failure is expensive, in terms of injury to personnel and loss or damage of equipment.

At present, the majority of safety critical mechatronic systems are developed using guidelines set out by the British Standards Institute, BSI, (BSI[91a], BSI [92]), and the Health and Safety Executive, HSE (HSE[87], HSE[93], PES[92]), and are based on the philosophy that safety should be designed into a product. This philosophy is also expounded by Bowen and Stavriduo [93], who stated that, "Safety should be designed into a system and danger designed out". This is echoed by Ullman[92]. BS EN 292 part 1 (BSI[91a]) attempts to take this attitude into account, although the design methodology guidelines offered are unclear and seem reactive rather than proactive. Even in the later BS EN 954 (BSI[92]) for safety related control systems, safety measures seem to be considered as an add-on to the original machinery, instead of an integral part of it. Due to the high-tech nature of mechatronics, the safe use of information technology in the development of such systems is another problem, which has only been partly answered (HSE[87], HSE[93], PES[92]). The main solutions are based upon the application of a number of redundant back up systems, which, although covering system failure, do not analyse potential problems to prevent them from occurring. From this basic review of current legislation it is apparent that a systematic

design methodology for the development of safety critical systems is necessary. Furthermore, this is imperative in the light of current EC legislation (EC[93]), which needs to be complied with before a CE mark can be given and a product sold on the open market. In support of this, there are certain conditions, identified by Archer[84], which support the use of systematic design, design methods and design methodologies. This is when one or more of the following three conditions are met:-

1) The consequences of being wrong are grave.
2) The probability of being wrong is high.
3) The number of interacting variables is large.

All three of these conditions apply to the design of the remote gamma radiography system described in chapters 7, 8 and 9. The product is safety critical and, therefore, the risks associated with it, i.e. inadvertent radioactive source exposure, are high. The possibility of developing an inappropriate design is high, because this type of device does not exist at present. Finally, the problem of analysing the interactions of the various failure modes with the control software is extremely large. Based on this reasoning, the need for a systematic approach is indicated.

The main objective, stated on page 1, will be met by investigating tools for the development of safety critical mechatronic systems. This will cover risk analysis tools, engineering design methods and systems modelling tools. Tools will be assessed and specific methods tested on the design of a remote gamma radiography system for use on steel storage tanks. The system was chosen because it complies with the criteria previously given for a safety critical system. It is partly mechanical. In its present form it has a number of electrical components and computer based logic. In terms of being safety critical, the process is extremely hazardous due to the risk of operators being exposed to radiation (Chamberlain[96]).

The other justification for developing a fully remote gamma radiography system is the high demand. There are estimated to be over 1 million tanks of various types world wide. Some of the oldest steel storage tanks are understood to be over 40 years old. All tanks require inspection on a 5 to 10 cycle year and, in the case of new installations, 100% weld inspection (Chamberlain[95]). Gamma radiography is one of the most widespread methods used for the testing weld integrity and demand for it is increasing. Because of this increase in demand, and moves to reduce radiation exposure, the development of a fully remote automated system is interesting. The work will be based

upon a modified version of the methodology originally used to design the prototype. The methodology is detailed in Chamberlain[95] and summarised in Chamberlain[96].

1.1 Background review

1.1.1 Engineering Design Methods

Design models are descriptions of a sequence of activities performed, or expected in the design process. A broad classification of design models divides them into two categories, namely (a) descriptive models and (b) prescriptive models. Descriptive models are based upon rules of thumb, intuition and observations on the methods of, successful designers. They model heuristics.

A number of studies have been carried out into how designers work and a number of problems identified, see section 1.1.1.2. It was to overcome these problems that prescriptive models were developed. These describe the order of tasks and activities that the proposer feels necessary to carry out the design process in the best possible manner. They rationalise the design process and make it more systematic.

1.1.1.1 Descriptive methods

Cross[89] defines a simple descriptive model based upon generation, evaluation and communication. See figure 1-1.

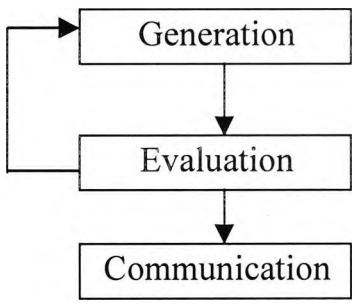


Figure 1-1 Descriptive Model of Design, (Cross[89])

During generation, a concept is developed from the original problem definition and this is evaluated against the design brief. The concept is then modified or changed completely, in an iterative manner, until it closely matches the requirements of the design brief. At this point the design is communicated to the manufacturer.

A more complex model of the design process developed by Professor Michael French, know as the French model (French[71]), Figure 1-2, is based on a number of set design activities.

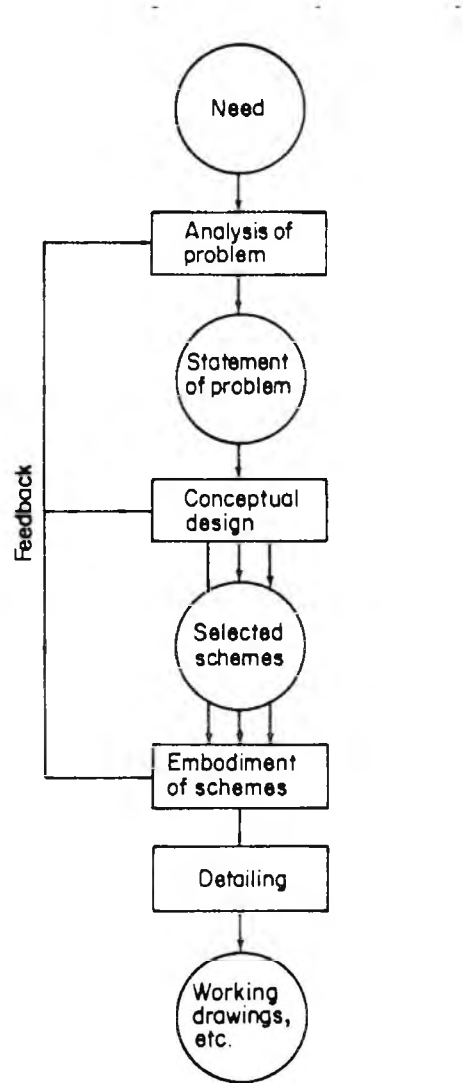


Figure 1-2 The French Model of the Design Process (French[71])

The French model encompasses the following design activities:-

- Analysis of the problem:- This involves identifying the problem to be solved as precisely as possible or desirable. This results in a statement of the true design problem, identifies limitations and constraints of the solution and defines the criteria of excellence to be worked to. This is a design specification developed from the original project brief.

- Conceptual design:- In this phase, numerous solutions (in the form of schemes) are generated from the design specification. Engineering science, practical knowledge, production methods, materials, commercial considerations and other factors are taken into account before selecting the final scheme(s).
- Embodiment of schemes:- This stage deals with developing the schemes from the previous section in more detail. At this stage, many problems will be found within the proposed scheme, so there is generally a lot of feedback to the conceptual design stage and the re-analysis of the problem.

  This is in agreement with Esterline[88] who found that the process of design could not be separated from the problem formulation process. From the initial statement of the problem, schemes are developed, which aid the designer in understanding deeper problems within the statement problem and current design. The iterative manner of this model is common to most other design models and the reduction of these iterations, to improve efficiency, led to the idea of "right first time design". The emphasis of this design philosophy is to look deeply at the problem statement and needs before carrying out any design work.
- Detail design:- The selected scheme is worked into finer details, and decisions made regarding very small but essential points. The result of this process is generally working drawings of the chosen scheme.

1.1.1.2 Why a more structured and systematic approach is necessary

Three main problems have been identified in the literature to support the need for a more prescriptive or systematic approach.

Firstly, designers tend to use familiar solutions and not explore alternatives or innovative ideas unless their new design fails badly and cannot be salvaged, Marples[61], cited in Finger and Dixon [89a]. This lack of flexibility, also reported by Ullman and Diettrerich [86, 87], is not confined to mechanical designers, as Adelson[84] found in his study of software designers. This goes against the second and third stage of the French model (French[71]), which imply that numerous solutions (schemes) should be conceptualised and evaluated. This is a large problem when the speed of technological advances in fields such as materials, processes and electronics is

considered. This is because various solutions may exist, due to the presence of new technology, but are not considered. This is due to the designer's lack of awareness of technological advances.

The second problem is the human difficulty of considering more than a few solution variants within given time constraints (Ullman [92]). This reflects on the limit of short term memory to hold between 5 and 9 pieces of information at any one time (Miller [56]).

The final problem is due to the lack of understanding shown by designers in the design approach taken, and discipline in applying the design model to the process (Tebay[84] and Hykin[75]).

The adoption of a prescriptive model can help combat the first problem by ensuring that a series of rational and logical decisions are made. The second problem is overcome by detailing the design process through various methods such as interaction matrices, morphological charts and record keeping, design methods common to the prescriptive model. Unfortunately, the final problem can only be surmounted by improving design education by applying structured design methods to real problems. This is starting to happen with the wide acceptance of methods such as Quality Function Deployment(QFD) (Akao[90]). It is also apparent in the number of design courses now available in the United Kingdom.

1.1.1.3 Prescriptive methods

As stated above, prescriptive models were developed to improve the way that designers work. They offer a more algorithmic, systematic procedure for the designer to follow and are regarded as providing a particular methodology (Cross[89]).

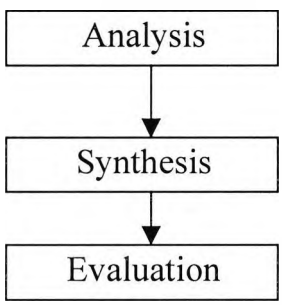


Figure 1-3 Design according to Jones[63]

One of the earliest prescriptive models is described by Jones[63], Figure 1-3. He suggests a structure based upon Analysis, Synthesis, and Evaluation:

- Analysis:- Listing of all design requirements and the reduction of these to a complete set of logically related performance criteria and constraints.
- Synthesis:- Finding a number of possible solutions for each performance criteria within the given constraints, eventually generating complete designs that fulfil the performance criteria.
- Evaluation:- Each design is compared and evaluated as to how well the performance and constraint criteria is met. This can include manufacturing, assembly and costs.

Although similar to the descriptive design process of French (French[71]), the emphasis is on producing a number of solutions by analysing logically the design brief to produce a performance specification, synthesising several alternative solutions by building up sub-solutions and then evaluating the solutions rationally against each other and the brief. As mentioned above, this is not the procedure that is followed in many cases.

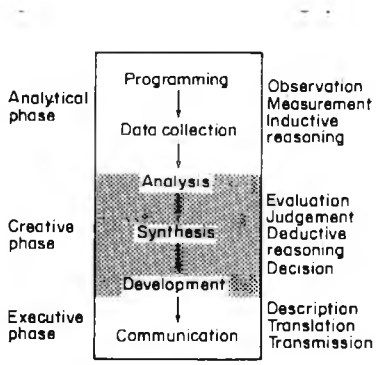


Figure 1-4 Design According to Archer[84]

Archer[84], according to Cross[89], produced a more sophisticated model,figure 1-4, which takes into account external interactions with the design process, such as the designers experience, information gathering and client input. Archer identifies six types of activity:-

- Programming:- Establish crucial issues and propose course of action.
- Data Collection:- Collect, classify and store data.
- Analysis:-Prepare performance specification, identify problems and sub-problems, and reappraise initial program and estimates.

- Synthesis:- Prepare outline design proposals.
- Development:-Develop prototype design and validate design.
- Communication:- Prepare manufacturing documentation.

This process is simplified into three broad phases, analytical, creative and executive. By comparing figure 1-3 to figure 1-4, it can be seen that the 3 phase model of Archer (Archer[84]) is very similar to the model proposed by Jones (Jones[63]). Many such models of the prescriptive design exist, however, the vast majority of them tend to be very complex, breaking the design process down into a hierarchical flow diagram of tasks. Within these models, the prescribed functionality of the design process is lost, due to the level of detail.

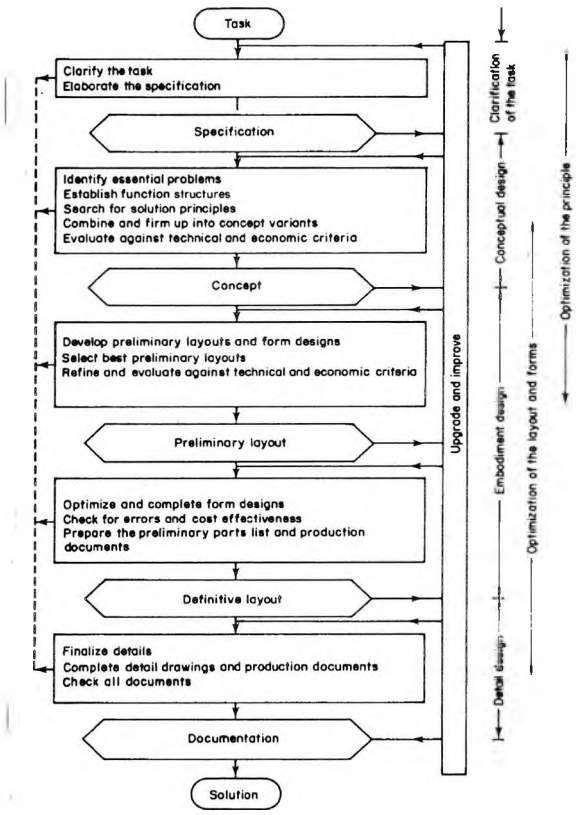


Figure 1-5 Pahl & Beitz's Design Model (Pahl & BeitZ[84])

Pahl and Beitz[84] (see figure 1-5) produced a detailed prescriptive model of the design process that sets out the various tasks and stages required, in a very comprehensive manner. The process takes the French model (French[71]) and expands upon the general format of task clarification, conceptual design, embodiment design and detail

design. This is done by including tools that can be used within the design process, such as function structures (Hubka[82]). This moves the French model from the descriptive domain into the prescriptive.

An extension to the work of Pahl and Beitz are the models produced by Verein Deutscher Ingenieure (VDI), Germany's professional engineering society. They produced a number of guidelines for design, including VDI 2221 "Systematic Approach to the design of Technical Systems and Products". The model produced by VDI is very methodical, with the design process broken down into a number of general working stages see figure 1-6., this makes the process very generic and applicable to many areas of engineering design, (Cross[89]).

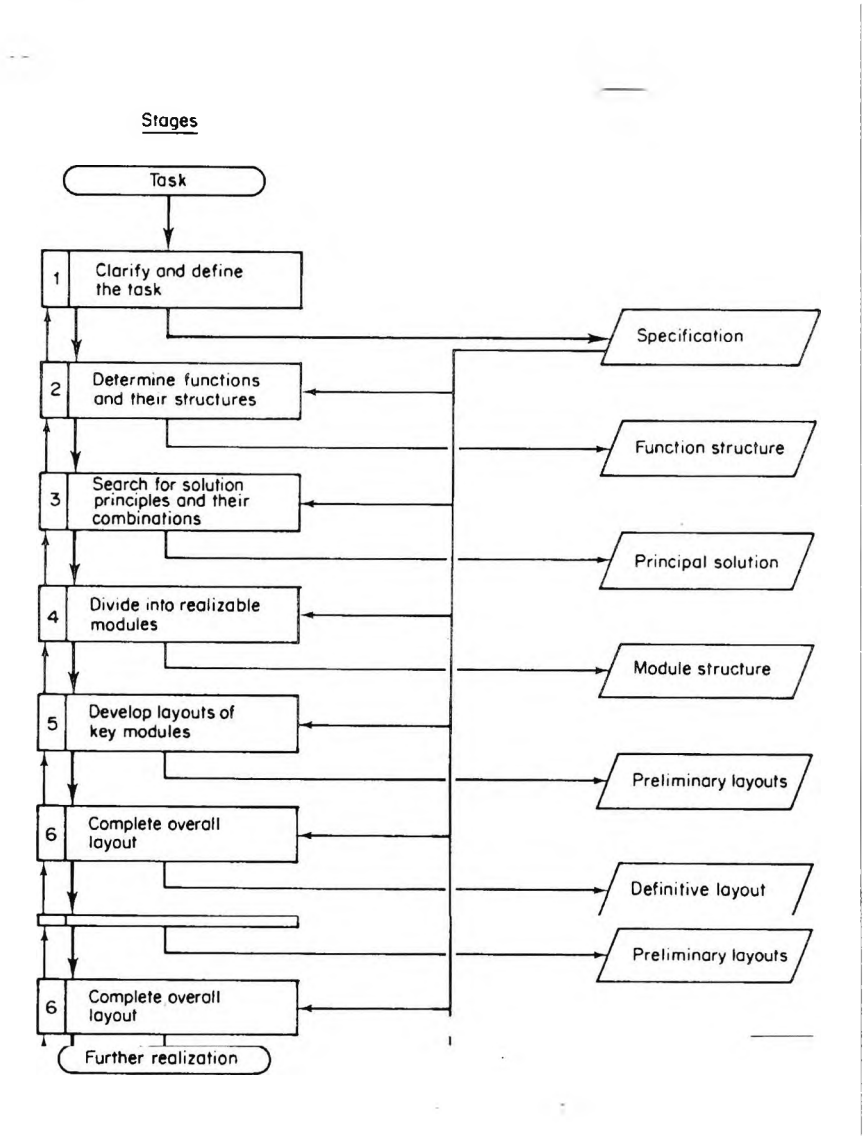


Figure 1-6 Design According to VDI 2221

1.1.1.4 Design Methods and Methodologies

A Design Methodology, according to the American Society of Mechanical Engineers, is a "collection of procedures, tools and techniques, which the designer may use in applying design theory to the process of design."(ASME[86]). The models of Pahl and Beitz and VDI are methodologies that prescribe a systematic process covering the design lifecycle from the initial design brief to manufacture. They specify tools (design methods) which can be used during the design process. These tools are necessary as they give the designer weapons which they can use to attack a design problem more efficiently.

Jones[81] identifies the existence of four groups of methods:-

1) Exploring design situations (information gathering techniques).
2) Searching for ideas (concept generation techniques).
3) Exploring problem structure (techniques which look at the interaction between and analysis of functions and components).
4) Evaluation (techniques which are used to compare solution alternatives).

Cross[89] simplifies this into creative and rational methods. The former appertain to methods of searching for ideas. On the whole, these methods tend to be less structured and more intuitive, their objective, to widen the solution space and generate alternatives. Rational methods are much more systematic and relate to the other three groups identified by Jones's (Jones[81]). The purpose of rational methods is to help the designer understand:-

- How different design solutions work.
- The pro's and con's of each solution.
- The interactions between components

Other benefits are:-

- Solutions are evaluated logically
- The designer is freed of the mental task of retaining information (i.e. externalises thinking).
- Design methods facilitate team-working.

Further benefits, identified by French [71], are:-

- Increased insight into problems and speed of acquiring insight.
- Diversification of approach to problems.

- Reduced mental step sizes required in the design process.
- Prompting of inventive steps, and reduced chances of overlooking them.
- Generation of design philosophies (synthesising principles, design rationales) for the particular problem.
- Improved record keeping of the design process and its progress. This increases knowledge retrieval within the company and should enable future designs to be developed more efficiently.

1.1.1.5 Concurrent Design Methods

Despite the development of complex design models, on the lines of those shown above, these are not capable of dealing with the design of a modern, complex products. This argument is stated in various ways by Cross[89] and Ullman[92]. To summarise, modern design procedures or methods are needed for the following reasons:-

- Increased complexity of modern design.
- Take into account technological development of computerised design tools, i.e. Finite Element Analysis, 3D product modelling, 3D kinematic and dynamic analysis tools etc.
- Need for products that have not existed before (i.g. solar powered vehicles).
- The requirement to develop team working practices because of the size of projects and the technological range of subsystems.
- To reduce the high risks and costs associated with modern design.
- To improve manufacturing and design efficiency.
- Demands for increased functionality and customisation of products.
- Design for a particular feature or function, i.e. design for cost, manufacture, safety, parameters etc..
- Escape from the traditional "over the wall" approach to engineering design., figure 1-7(Evbuomwan[95]). The design process cannot progress to the next stage before the current stage is complete, hence the term "over the wall".

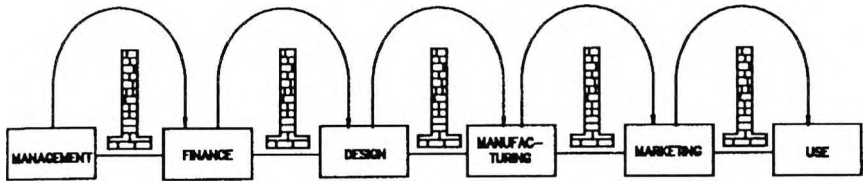


Figure 1-7 Over the Wall Approach to Engineering (Evbuomwan[95])

The initial attempts to improve the efficiency of the design process lead to the development of concurrent engineering. In concurrent engineering, the stages involved in the design process are started as soon as possible, so that processes run in parallel or concurrently (see figure 1-8).

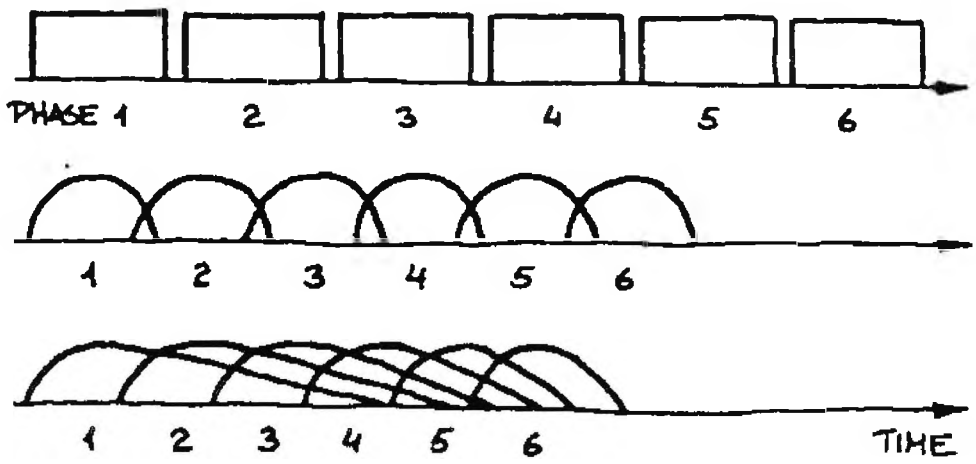


Figure 1-8 Comparison of time scales for Traditional engineering and Concurrent engineering (Burr[90a])

This is possible because it is not necessary, for example, to complete the design stage before the analysis stage. Instead initial analysis can be carried out while the design is progressing. Similarly once the design has been roughed out and basic analysis completed it may be possible to start the manufacturing planning. This approach reduces the length of time involved in the product design life-cycle dramatically (see figure 1-8).

The idea of concurrent engineering has been extended to look at the entire design in a holistic way. This has lead to methods which take into account the requirements of all aspects of the design life-cycle, marketing, design, analysis, production engineering,

manufacture. Two methodologies which reflect this approach are Andreasan's Integrated Product Development (Andreasan[87]), and Design Function Deployment (Sivalogonathan et.al [95]).

1.1.1.5.1 Integrated Product Development

Integrated Product Development (IPD) according to Hein[90], from Andreasan[87], "Embraces strategies and methodology for product development , based on a synthesis of the considerations related to the Market, the Product and the Production." This idea is represented in figure 1-9, taken from Andreasan[87], it shows the concurrent nature of IPD being represented as three functions, based upon marketing, design and production which are carried out concurrently, as the stages in figure 1-9 suggest.

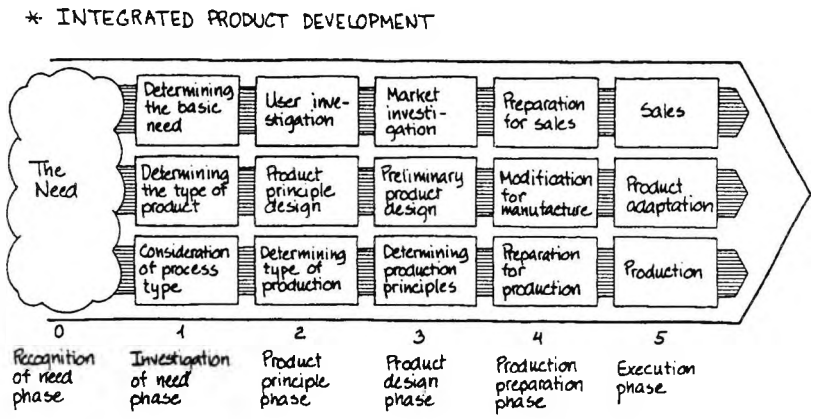


Figure 1-9 The Building Blocks of Integrated Product Development (Andreasan[87])

1.1.1.5.2 Design Function deployment

Design Function Deployment (DFD) is a concurrent engineering matrix based design methodology, developed at City University, Engineering Design Centre (Evbuomwan[95]). The method is based upon the Quality Function Deployment (QFD), Akao[90], 'House of Quality" and consists of five common stages, if used throughout the complete design life-cycle.

1) Translation of customer requirements, specifications and constraints into design functions.
2) Generation of Conceptual designs, and alternatives, from the design functions.

3) Detailed or embodiment design, from the selected concept.
4) Analysis of appropriate manufacturing technologies for each process.
5) Application of Material Resources Planning (MRP 1) and Manufacturing Requirements Planning (MRP II) data.

The integration of each stage is shown in figure 1-10 .

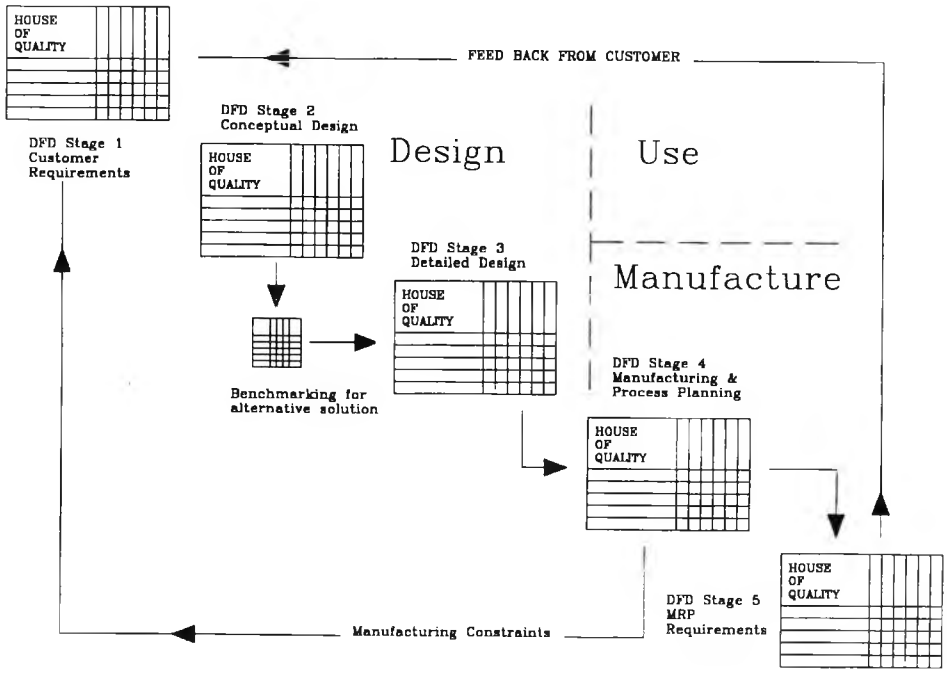


Figure 1-10 Stages in the Design Function Deployment, design life cycle

Stage 1. Customer Requirements to design functions

Customer requirements are the specifications to which the product has to comply, to be successful. However, in this instance the term customer requirements includes the needs from internal departments such as marketing, manufacture, packaging and would cover constraints such as legal constraints, safety issues, or even the maximum size of the product. Customer requirements can be vague, non- technical or non-specific, there can be further divided into explicit and implicit requirements. An explicit requirement could be , weight less than 1 Kg, and an implicit requirement could be compliance with

BSI safety procedures, i.e. a requirement which the specifier may not be aware of but the designer is. These requirements are WHAT the design must satisfy. Once these have been established, the designer can compile a list of design functions, e.g materials, sub-assemblies, basic manufacturing techniques etc. These are precise technical ideas which can be measured or specified i.e. They give an indication of HOW the customer requirements can be satisfied.

A top level DFD chart is shown in (figure 1-11). It consists of the WHATS on the left hand side and the HOW's across the top. The relevance of each WHAT to a HOW is analysed by rating the relationships between them. By implementing technical difficulty ratings for each HOW and importance ratings for each WHAT it is possible to identify the most important features in the design and the major cost drivers.

Stage 2. Concept generation

The HOW's from the previous stage are placed down the left hand side of a new matrix and become the WHAT's for the next stage. Each WHAT is analysed using creativity techniques so that at least two solutions to each design function is produced. These solutions are benchmarked against the original customer requirements so that there is a viable design which meets all the customer requirements.

Stage 3. Detail/Embodiment design

This phase involves design calculations, detailed materials selection, detailed component selection and detailed drawings. The conceptual design derived from stage 2 may need to be revised as a result of these calculations.

Stage 4. Manufacturing

The manufacturing process for each component can now be decided. This may also define make or buy policy. The detailed designs from stage 3 may be altered to fit in with the manufacturing process requirements, i.e. certain wall thicknesses may not be possible for a casting or a flow moulding.

Stage 5. MRP requirements

This provides a record of the detailed component structure, bills of materials, and production layout/schedule.

It must be noted that not many case studies exist in the public domain which detail the DFD process after stage 3, hence the lack of detail in the above.

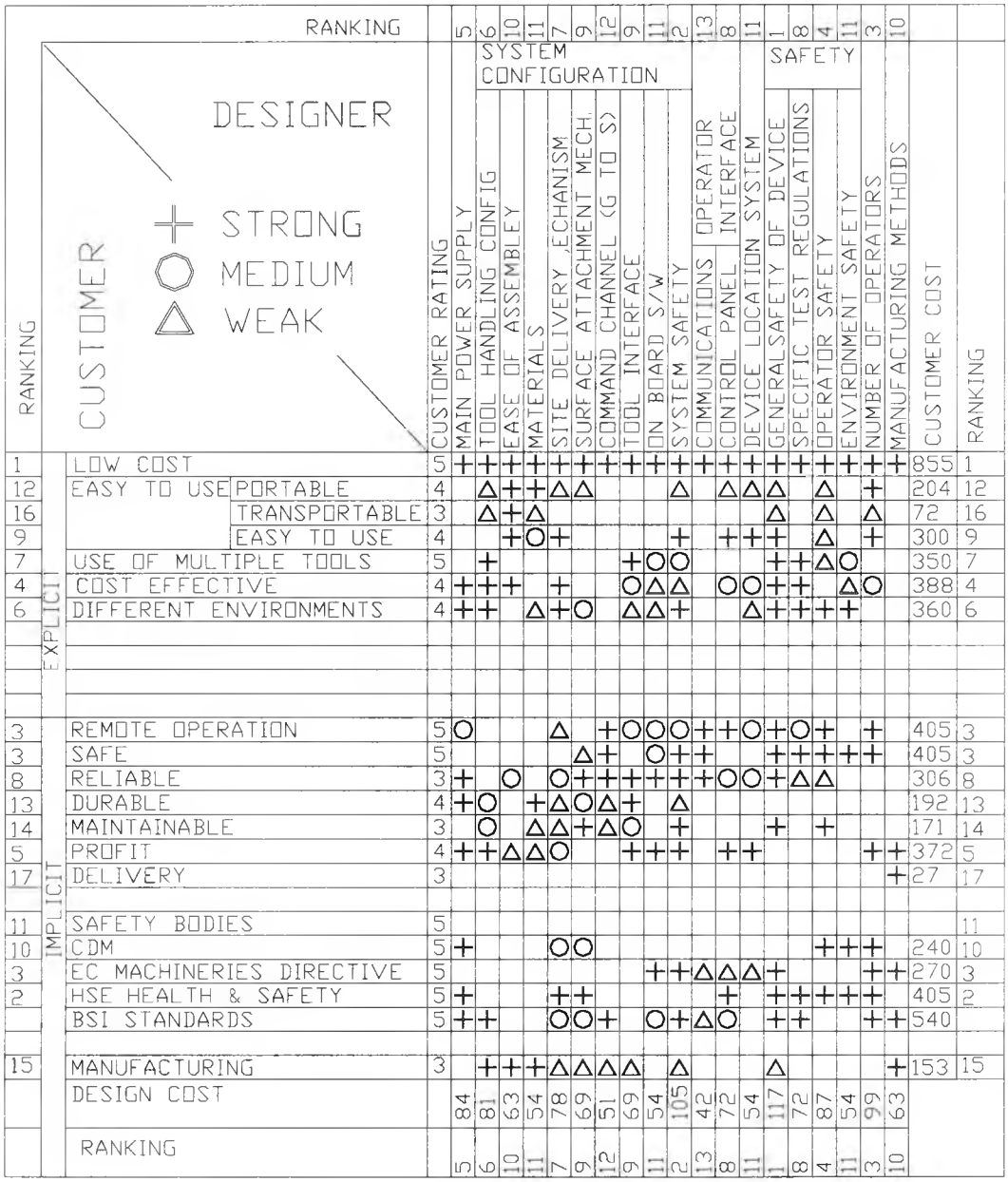


Figure 1-11 Top Level DFD chart for Automated Gamma Radiography Robot

1.1.1 Review of current mechatronics design methods

Mechatronics as described is defined in Buur[90] as a "a functional and spatial integration of mechanical and electrical components with information technology". This describes a device with electromechanical features combined with a degree of computer control. Bearing this definition in mind, examples of mechatronics range from common household products such as video recorders, microwaves or washing machines, through to engine management systems, modern central heating controllers, upto industrial

applications such as mass production conveyor lines, robots and CNC machines. Even a computer could be considered a form of mechatronic product.

Despite the large variation in the form that mechatronics can take, the main differences between all the examples stated above is the degree to which a specific discipline is applied to the product. For example, a computer is a mechatronic product however it mainly consists of an integration of electronic and information technology components, the only mechanical components being the disc drives and the key board. Another example is a robot, in which the integration is well balanced between three domains with the emphasis on the mechanical and the information technology domain. Finally a washing machine is selected, its main elements exist in the mechanical and electronics domain. All three of these examples can be seen on a mechatronic domain graph, figure 1-12, adapted from Frolich[82].

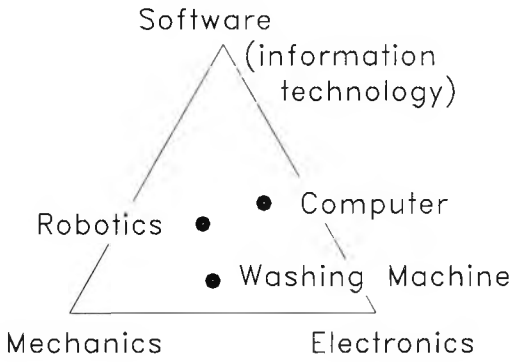


Figure 1-12 The position of products within the mechatronics domain

Because of the integration of the three disciplines, a number of problems exist with design for mechatronics, which differentiate it from product or industrial design. These differences are outlined below, after which three methods for mechatronics design are reviewed and the problems with them identified.

1.1.1.1 Why a specific method is required for Mechatronics design ?

Bradshaw[90] outlined some problems of mechatronics. One of the primary reasons given is the failure of software and electronics to be regarded as an integral part of the design. This is because it is often the physical realisation of the product that firstly captures the imagination and attracts attention. The result is that the interactions between mechanical systems, electronic devices and software are often neglected with software and electronics considered as bolt on extras. This leads to a call for an

approach to the mechatronics design problem in which each aspect of the product is compatible with overall development.

A further problem is stated in Bradley[93], which relates to the problems of design representation between the different disciplines. Mechanical design is considered as a 3 dimensional, spatial problem, whilst electronics design is described as 2 dimensional and software, non-dimensional.

Ullman[92] and Buur[90] identify the differences between the design process for each of the disciplines as being a problem for mechatronics design, see table 1-1, derived from their incompatibility.

	Mechanical	Electrical	Software
Type of objects produced	vary in shape, form, functional & technological complexity	Standard components	Standard structures of text strings
Types problems solved	Selection configuration parametric and original	Primarily selection and configuration but also parametric	Selection and Configuration
Form-Function relationship	Overlapping	standard combinations of components have specific functions	Structural form defines function
Decomposition potential	Often Strongly coupled	Along circuit and component boundaries	Into subroutines, procedures and objects
Representational form	Textual Analytical Graphical (2 D, 3D shaded sketches	Textual Analytical Graphical (2 D)	Textual but mainly 2D graphical (hierachial flow diagrams, petri nets, JSD etc.)
Design methods	Generally based on functional decomposition,	Some available (silicon compilation)	Many methods available, use dependant upon program function

Table 1-1 The incompatibility of mechatronic design domains (Ullman[92])

The number of possible concept solutions which need to be evaluated is increased dramatically if a mechatronics approach is taken (Bradley[91]). This is because the number of ways to realise a particular function is at least multiplied by the number of disciplines which can be used.

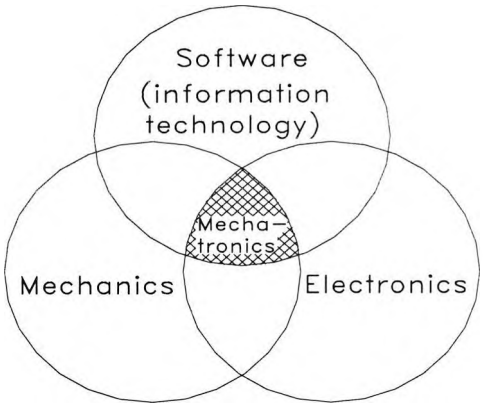


Figure 1-13 Domain Interaction spheres for mechatronics

On account of the multidisciplinary nature of mechatronics, it is necessary to have a team approach to the design problem (Salminem[92]) or the person carrying out the design has to be a multi-disciplinarian (Buur[92]). This is the only way in which all the possible concept solutions can be identified and evaluated. Figure 1-13, identifies the interaction spheres between the various disciplines or teams of experts. Salminen[92] also identifies the need for a solution that allows accurate recording keeping of the design process , due to the fact that no one person can be expected to contain and evaluate all the information involved in the design unaided. Figure 1-13 shows how the various expert groups should co-operate with each other in the conventional design process. This can also be interpreted as a form of concurrent engineering, as the design process between the disciplines is continuing simultaneously with communication across boundaries.

In summary, it is clear that a number of differences exist between product or industrial design and mechatronics design. This mainly concerns the scope of the technology available to realise a product, the need for experts in a number of fields and the incompatibility of the various disciplines.

1.1.1.2 The laws and axioms of Buur

One of the principal works in the field of mechatronics is the Ph.D. thesis of Jacob Buur, who in 1990 (Buur 90a) developed a series of theoretical guidelines and axioms

for mechatronics design. This work proposes the use of hierarchical functional analysis, based upon Hubka's law of Vertical Causality(Hubka[82]), which states that.

> "There is a casual relationship between the (purpose) functions of the system and the alternative means of realising it. Once the means for a function on one level is found it can be decomposed into sub-functions on a lower hierarchical level, each of which in turn requires its choice of means"

Buur also based his ideas upon Andreasan's Theory of Domains (Burr[90], Andreasan [89]) which covers concepts such as, transformation domain (the system is represented by a black box which transforms energy to achieve the function), process functions, (the means by which the transformation of energy is carried out), and organs(groups of components used to achieve the purpose) and parts (single components which make up the organ), to describe the mechatronic system and aid technology allocation. A great deal of effort is put into the development of the theory for the state modelling of mechatronics. For this, he used traditional, Finite State Machine, transition diagrams

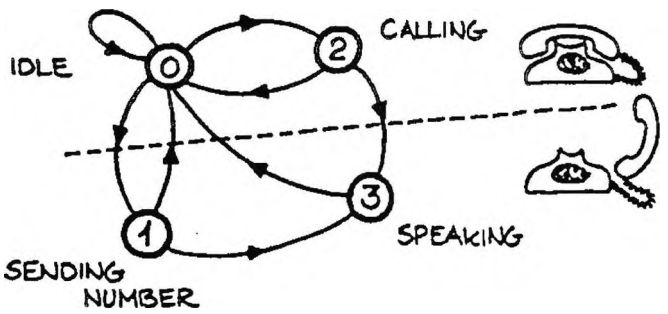


Figure 1-14 Finite State Machine Model of a telephone (Buur[90])

with text or state tables describing the logical operation of the system, Figure 1-14. Although the work is thorough in its examination of the needs and the theory behind mechatronics design ,it is very theoretical in outlook. The only examples are based upon simple systems. The need for design specification and design evaluation methods are recognised, but ignored. This is on the basis that they are independent of technologies and therefore not specific to mechatronics. The functional analysis approach is easy to use, but no true structure is in place to force variant solutions. Finally, the interpretation of the, Finite State Machine, transition diagrams for systems

modelling is extremely complex for multi state systems. This disadvantage is admitted by Buur in his conclusions.

1.1.1.3 The Finnish Approach, 'Metamethodics'

Around the same time as Buur's work, a three year Finish research program in mechatronics was being undertaken (Salminem[89,90,92]). This program developed a system called Metamethodics, based upon Quality Function Deployment (QFD, King [87], Akao[90]) as the design requirements capture and market comparison method, and the German standard, VDI 2221 (VDI 86) as the conceptual design vehicle. To carry out system modelling, a method known as System Analysis/System Design (SA/SD) see figure 1-15, developed by Ward and Mellor[85] for software design, was combined with VDI 2221. This method results in a more sophisticated form of State Transition Diagram (STD), see figure 1-16, to model the logical operation of multi-state components.

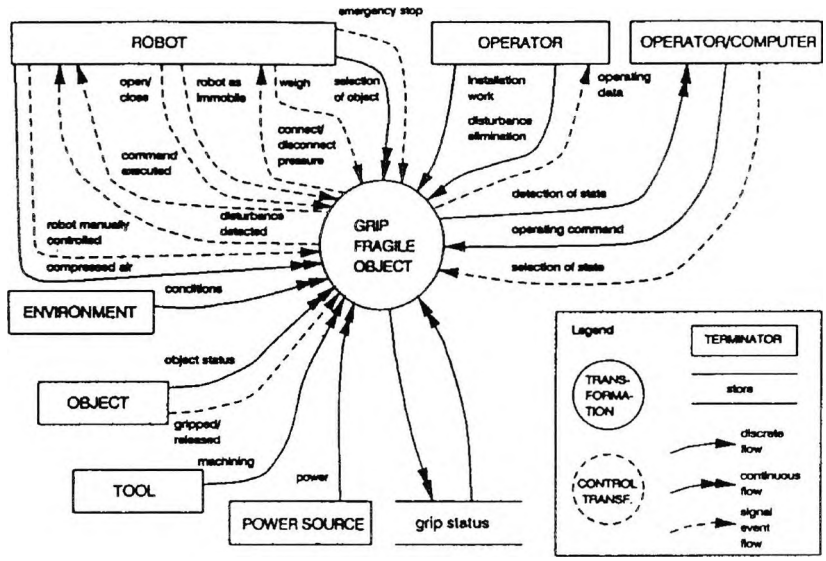


Figure 1-15 SA/SD, context diagram for a Robot Gripper (Salminem[90])

The author feels that this work was a step forward from Buur's work. This is because it offers a more rounded approach to the problem of mechatronics design. QFD works well as a requirements capture method, as it identifies areas, in a structured manner, where product improvement is required. Unfortunately, the application of QFD in this instance is very basic. It only used as a comparison between existing products in the market place and although ideal for generating improvements in existing products, it is

not very useful for spawning original designs or concepts (Evbuomwan et. al 94). The area of conceptual design and functional analysis is carried out according to VDI 2221. Problems with this approach (detailed in Salminem, Verho 92) are :-

- It is awkward for building up overall solutions.
- It has no provision for innovative analysis of the whole.
- There is no model for logical functionality.
- STD's, although, a lot more understandable than the state transition diagrams used by Buur, can still be very complex when used to describe multi-state systems.
- No geometric description of the product can be produced from the application of SA/SD.

It is also pointed out that SA/SD can be used for safety and reliability analysis, though this application is not described in detail.

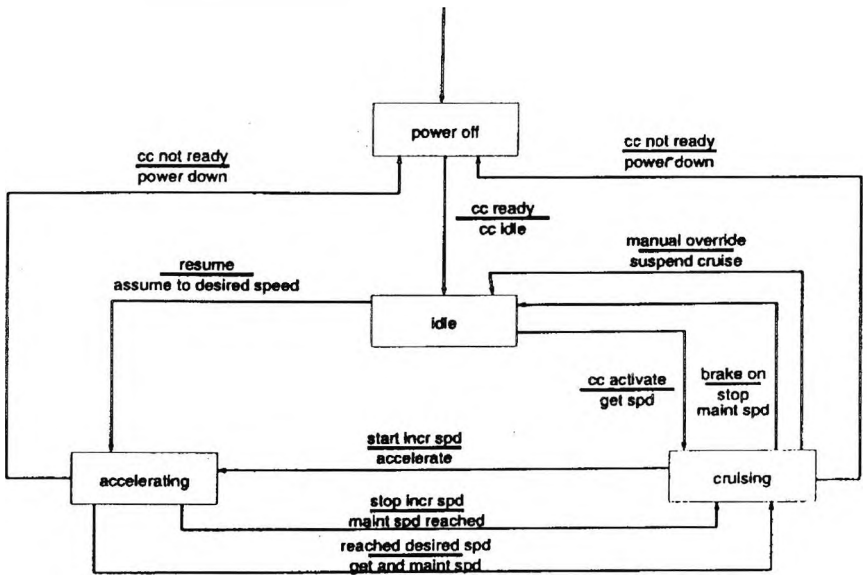


Figure 1-16 SA/SD State Transition Diagram, for Robot Gripper, (Salminem[90])

1.1.1.4 Design Function Deployment (DFD) and Mechatronics

Although Buur's thesis is flawed in some ways, it does layout the fundamental theoretical requirements for a system that can be used for mechatronics design. These are, a conceptual design based on functional analysis and a system modelling stage. This model is developed further by Salminem and Verho, to include an initial

requirements capture stage. From his point of view, any system for the design of mechatronics should incorporate these three stages into the design life cycle. It was decided to carry out a pilot study to test the usefulness of certain tools.

The DFD methodology, as specified in Edney[93] , was selected as the vehicle for requirements capture and design. As no system modelling technique was associated with DFD, it was decided to attempt to use an approach called Activity, Channel, Pool diagrams (ACP), Mort(90) , which it was hoped would be an improvement upon state transition diagrams and SA/SD. Paper based, Failure Modes and Effects Analysis (FMEA), and Fault Tree Analysis (FTA) were carried out as apart of the design life-cycle as indicated in figure 1-17.

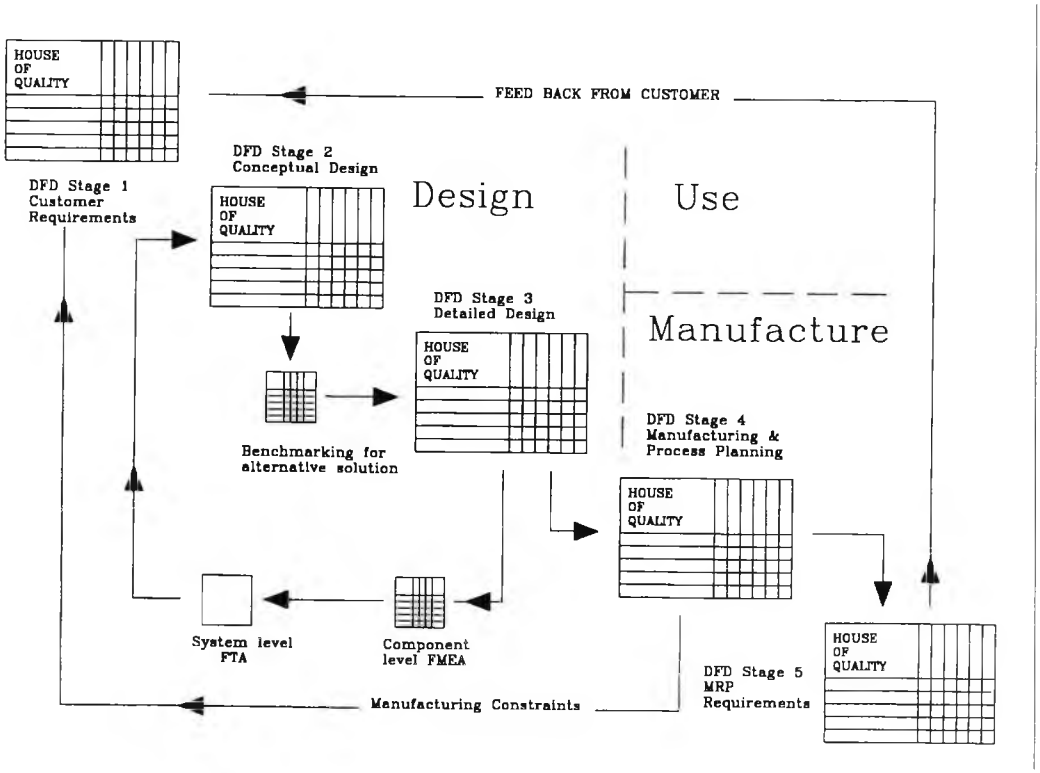


Figure 1-17 The Integration of FMEA and FTA with DFD

These methods were applied to the design of a robot for remote gamma radiography. This work is detailed in Chamberlain[95] and summarised in Chamberlain[96], to which the author made a large contribution. The work proved to be successful but the author found certain limitations in the application of the DFD methodology. Namely, that in its present state it was awkward to use, this was due to the size of the matrices produced when applying the rules laid out in Edney [93]. It was later found that the

application of the DFD methodology was in fact very similar to an extension of Quality Function Deployment called Enhanced QFD (EQFD).

ACP was found to be very useful as it linked with the correlation chain in the roof of the DFD matrix, thus allowing basic systems analysis to take place at a high level, see figure 1-18. Detailed ACP diagrams were produced to aid in the detail design of the hardware, software and safety analysis of the system, although time consuming, this was very successful. ACP diagrams were transferred to software and used to check the software and hardware design. This showed up some serious faults in the design which were corrected as a result of generating the model, Bleakley[97].

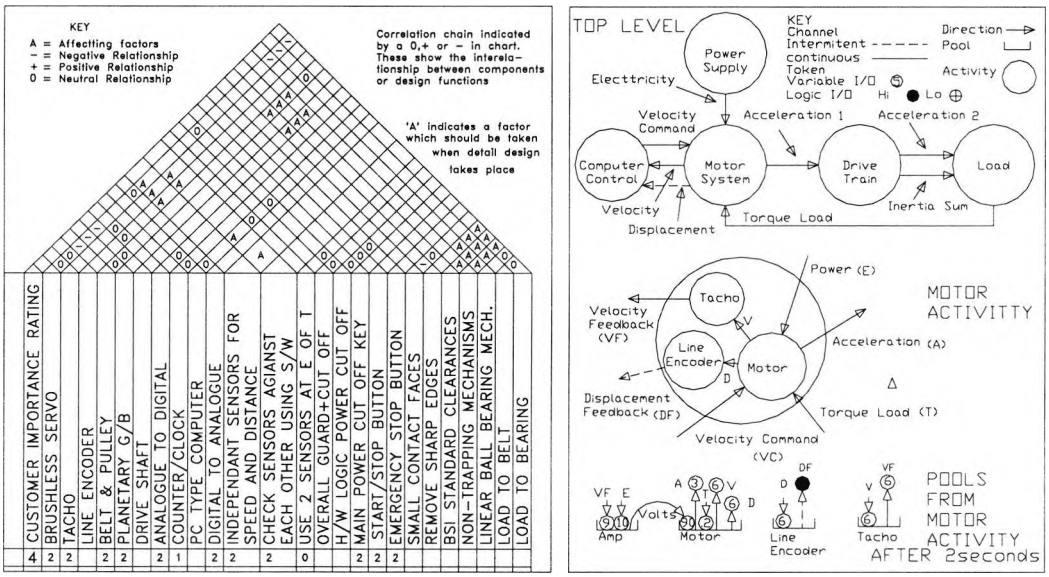


Figure 1-18 The interaction of DFD and ACP through the use of the correlation matrix (Bleakley[97])

FMEA and FTA could be implemented very easily within the design strategy of DFD/EQFD. It was found that the ACP system model could be used to generate Fault Trees, this is because the ACP diagrams show the interaction of components at every level. FMEA lent itself very easily to the matrix methods of DFD/EQFD, as it a matrix based ranking system.

1.1.2 Review Summary and conclusions

A summary of the advantages and disadvantages of each method can be seen in Table table 1-2.

	Basic Description	Method Scope	Design tools used	Advantages	Disadvantages
Theoretical Guidelines and axioms for mechatronics design	Textual description of theories appertaining to the design of mechatronic systems	Conceptual design function analysis Systems modelling	Theory of domains, The law of vertical causality, Function means trees, Morphological Charts traditional State transition diagrams	Breaks down the design into manageable areas Provides a good record of the design process Takes into account the interdisciplinary aspect of mechatronics	Textual descriptions of theories are highly complex. Modelling of multi-state, hierarchial systems. becomes very awkward No account taken of customer requirements or function evaluation.
Meta-methodics	Methodological approach to design based upon the application of a number of well established design tools.	Customer requirement analysis. Functional design and evaluation. Systems modelling. Risk Analysis.	Quality Function Deployment (QFD) VDI 2221 VDI 2222 Ward-Mellor State Transition Diagrams Failure Modes Effects Analysis	Covers the entire scope of the conceptual/ embodiement design cycle. Takes into account customer requirements Employs risk analysis techniques	VDI standards do not take into account the interdisciplinary nature of mechatronics design No real use could be made of the model created from the STD's Application of risk analysis to the design problems faced is not clear.
Design Function Deployment	Concurrent design methodology based upon QFD. Revolves around five stages which cover the design process from initial customer requirements through to production.	(Initial 3 stages only) Customer requirement analysis. Functional design and evaluation. Risk analysis.	QFD Heirarchical, Function analysis. Weighted objectives method. Morphological charts FMEA & FTA	Covers the design cycle Takes into account customer requirements Useful for original design Takes into account the interdisciplinary nature of mechatronics design. Employs risk analysis techniques.	Method in its standard form is awkward to use (from results of pilot study) No modelling facility is associated with the method Although mentioned no thought to the application of FTA and FMEA was made until the pilot study.

Table 1-2 Summary Comparison Of Design Methods For Mechatronics

Analysis of the table, combined with research carried out previously, and with the experience gained from the pilot study, it is felt that a design method for mechatronics should incorporate the following features and ideas:-

1. A Concurrent approach to mechatronics design should be taken. This is highlighted in Hein[90] , Salminem[92], Jebb[92], and Buur[92].
2. The methodology should incorporate a form of functional decomposition and analysis Salminem[92], Buur[90], and Jebb[92]
3. As there is very little or no application of proactive safety or risk analysis to mechatronics design, this should be integrated as one of the initial design stages.
4. Systems modelling is employed in all three methods, to a greater or lesser degree of success depending upon application. However, bearing in the mind the safety critical nature of many mechatronic products and its potential usefulness, the author feels it is a requirement for any mechatronics design method.
5. Computers were only applied in the DFD/ACP method, where it was seen that computer models were extremely useful. The author feels that computer based models are generally under-utilised for design, (this in agreement with observations made by Wang[97]) and that they could provide considerable support to the methodology. Component selection and failure analysis are good candidates as they both have large domain areas which need to be investigated.

1.2 Research Hypotheses, Objectives and Outcomes

Based upon the conclusions of the initial literature review, section 1.1, and work previously completed by the author (documented in Chamberlain[95]), it was felt that that this thesis should firstly, develop a pro-active safety and failure analysis approach to design within an design function methodology. And, secondly, look at applying computerised modelling techniques to the design process for component selection and failure analysis. DFD in the fully computerised form intended by the originators of the method (Evbuonwan[95], and Sivaloganathan[95]) is unlikely to be achieved in the near future. This thesis should add to knowledge about how computerised techniques can be integrated into DFD so that DFD, in its complete form, becomes more achievable.

1.2.1 Research Hypotheses

The intentions described above relate to the following hypotheses.

1) Integration of Failure Analysis and design methods:-The integration of risk and failure analysis with design function methods can lead to improvements in new mechatronic product safety and reliability.
2) Integration of computer models with the design process:-The integration of computer based models for component selection and failure analysis is feasible and beneficial to the design process.

1.2.2 Research objectives

The above hypotheses will be investigated to the following research objectives.

1) Develop a function based design methodology, which employs pro-active safety analysis features. The methodology will also be compatible with computer modelling tools for component selection and failure analysis.
2) Carry out a case study based upon the design and failure analysis of two axes of the gamma radiography robot.
3) Evaluate the success of the new methodology by comparing it to the methodology originally used for the design of the product (The previous design method used is described in Chamberlain[95, 96]).

1.2.3 Scope of Research

Fulfilment of the above objectives will lead to the following research deliverables:

1) Selection and definition of the problem:- The problem has to be large enough to be significant but small enough to be completed in a constrained time period. Because of the nature of the thesis it has to be a mechatronic product with a high degree of safety critical features. It was decided to apply the methodology to the design and failure analysis of two axes of the gamma radiography robot.
2) Review of function based design methods:- An examination of the QFD family of methods principally EQFD,EEQFD,RFCA. More general design theory such as the laws of Vertical causality, function means trees, theory of domains, and morphological analysis will be examined.

3) Review of paper-based systems modelling techniques and examine how to computerise them:- The techniques to be examined fall broadly into 3 categories graphical, mathematical, textual. Each will be assessed for its ease of use. Various methods of implementing them will also be examined such as Object Orientated Programming (OOP), rule bases and graphical modelling languages.

4) Review of pre- and post-active failure analysis tools:- The principal methods used for failure analysis are to be CLASH, HAZOP, FMEA, FTA. Various computer based tools are also to be reviewed.

5) Evaluate, select and integrate the design tools to be employed:- The type of problem selected has a bearing upon the tools chosen. However, it is intended that the methodology should be made up of common design tools which can be easily applied to a number of different cases.

6) Carry out a case study into the application of the various tools for the design of mechatronics. This involves the following stages:-

- Proactive safety analysis of the product
- Determination of customer requirements
- Functional analysis
- Determination and selection of viable concepts
- Generation of schematic computer model to determine system components
- Analysis of results of the use of computer based tools to select components
- Failure analysis of components
- Generation of detailed computer model for failure analysis.
- Analysis of results from computerised failure model

7) Comparison of the design tools used against the method used in the pilot study

1.3 Novelty of work

There are two main points of novelty to this work:-

1) The integration of new computerised tools into EQFD.
2) Design tools, not previously operated together, integrated into the QFD family of design methods.

These main points, and the benefits of their application to design problems, are discussed in sections 1.3.1 and 1.3.2

1.3.1 The integration of new computerised tools into EQFD.

At present only a few computerised tools have been integrated into any QFD, these are:-

- QFD Scope for Windows by IQD, USA, California
- The application of Excel to QFD for the determination of function targets, EXCELQFD (Atherton[98]).
- The application of Bond Graphs for the selection of control variables using Taguchi's design of experiments (Jebb[92])
- The application of Bond graph modelling to parametric design of components, using Robust Engineering Design methods (Atherton[98])
- Work on developing computerised DFD carried out at City University EDC, which was unsuccessful due to the overall complexity of the project.

The author's work integrates computer models into the EQFD design structure and uses them to carry out component selection and failure analysis. The benefits of this envisaged innovation are threefold:-

1) The efficiency of the design process is improved by employing a computer to carry out searches of component databases.
2) All viable combinations of component failure modes which lead to system failure are systematically tested and evaluated.
3) Adding to the knowledge already in place about the application of computer-based tools to the QFD family of methods, paving the way, eventually to a realisation of computerised DFD.

1.3.2 Integration of combined design tools in QFD methods

The application areas for these tools are safety analysis and systems modelling. Each area is discussed separately.

The main safety/failure analysis tools integrated or linked into the QFD family of methods will be FMEA and FTA, (Sontow(93)), Salminem and Verho(92), and Sivaloganathan(95)):-

- Sontow and Clausing(93) employed a form of pro-active FMEA as part of a case study into the design of a photocopier sub-assembly. In this instance the designers of the sub-assembly had a vast experience of possible faults that may occur, without this prior knowledge, pro-active FMEA could not been carried out.
- Salminem and Verho(92) performed FMEA and FTA as part of their work into the design of a robot gripper. However, the method was not integrated into the QFD section of the design methodology and the description of its operation was rather vague and unclear.
- DFD has been linked with FMEA and FTA previously, Sivaloganathan(95), but with no indication of how to integrate the methods within DFD. A true pro-active risk/safety analysis approach has never been attempted as part of the QFD method.

In this instance CLASH and HAZOP are to be employed to define possible safety hazards and enter the solution of these hazards as customer requirements. FMEA will also be integrated properly into EQFD in a manner similar to that proposed by Sontow and Clausing, but with greater effectiveness due to the simpler nature of the proposed methodology.

Although Ward-Mellor diagrams were used by Salminem and Verho for modelling a robot gripper, they were never integrated into QFD as part of the design process and no real use could be found for them. The Author's work will employ ACP diagrams, (Mort 1990)), a simple, graphical and textual notation to model the system on paper.

There is a direct relationship between the roof of the EQFD house of quality and ACP diagrams. From these simple models, intuitive fault analysis can be carried out. They also provide the basis for the computer model as they layout the general structure of the components and the flows of energy and signals.

The two main advantages of employing these methods are considered to be:-

1) Reduced iteration in the design process by designing safety and quality into the product at the beginning of the design life-cycle. Safety in the design of many products is often added as an afterthought to the design process. This is unacceptable as the safety of the user should be the main concern of the manufacturer/designer. It is also normal practice to carry out several iterations to improve the safety of a product. If these iterations are being carried out when the product design has nearly been completed, the knock on effect of changes can be very expensive.

2) By adopting ACP diagrams or any modelling technique as part of the design process, it is possible to generate a deeper understanding of the relationships, which exist between components. This is especially useful if intuitive fault analysis is being carried out as an understanding of the propagation of faults within a system is required to generate a comprehensive fault tree.

1.4 Scope of work

The objective of this section is to describe the limitations of the research, of which, there are two main areas:-

1) Research into Design tools & methods, leading to the development of a design methodology.
2) Application of the proposed methodology to a case study.

A number of stages are involved in each area, as described in the next section.

1.4.1 Research into design tools and methods

Research into design tools and methods will be focused upon the following areas.

- Review of design tools used for mechatronics design:- Principally tools relating to functional methods, these will include general design tools and a review of the QFD family of methods.
- Review of systems modelling techniques:- This will cover the main tools used for systems modelling, and the selection of an appropriate methodology. A

review, and evaluation of software methods to represent the modelling technique will be carried out.

- Review of failure and risk analysis tools:- A review and evaluation of paper and software based failure analysis tools will be carried out.

This will lead to the development of an integrated concurrent engineering methodology for safety led design of mechatronics systems. Despite selecting a concurrent engineering methodology, concurrent engineering, in its true sense, will not be carried out due to manpower limitations.

1.4.2 Application of the proposed methodology to a case study.

The design methodology will be limited to the following processes:-

- Determination of customer requirements from safety based tools.
- Development and evaluation of a number of solution concepts.
- Development of a top-level paper based system model for the proposed solution.
- Development and evaluation of a software model for embodiment design.
- Development and evaluation of a software model for failure analysis.

The design process will only be carried out up to the point whereby all single point component failures are removed from the fault tree. This is because a significant increase in the reliability and safety of a system is achieved by having a fault tree with no significant single point failures.

1.5 Thesis structure

The thesis is divided into four sections and ten chapters.

The first section consist of chapter one in this, the problem is introduced and the objectives of the thesis defined. The second section comprises three chapters of literature review and covers the areas of design methods (chapter 2), systems and software modelling (chapter 3), and chapter 4, fault analysis techniques. The third section is the case study and consists of five chapters . In chapter 5 the design methodology is defined, with examples detailing the use of all the tools employed. Chapter 6 describes the problem in detail, justify the

system specification and previous work. Chapter 7 takes the reader through the design process from the initial safety led design up to the end of the stage of concept selection. Chapter 8 describes the generation of the computer based model and how to use it to select components. The final chapter in this section, chapter 9, takes the reader through the process of computerised fault analysis. The fourth section consists of one chapter (chapter 10), it summarises the work, evaluates the results of the case study, in comparison to the method used previously, and makes conclusions and recommendations for future work.

Four appendices are supplied with the thesis. Appendix A, consists of a Glossary covering commonly used terms which used throughout the thesis. Appendix B displays condensed contents of the result files produced by the VISSIM models for component selection and failure analysis. Appendix C is a table which details the contents of three supplementary computer discs, which contain all the major VISSIM models, described in this thesis and the means of viewing them. This approach was taken as many of the models are extremely large and it would take over 200 hundred pages to display a single model. Appendix D is a list of papers and reports that the author has made major contributions to.

2 Review of design methods and tools

2.1 Introduction

Chapter 1 gave an introduction to the main methods and tools used for mechatronics design. This chapter takes this work a step further, providing the reader with a more detailed description of the various tools and methods used within Enhanced Quality Function Deployment and Function Based Design. The final objective being the development of a function based design method for mechatronics. To reach this objective two sub-objectives need to be met.

The first objective is to describe the common design tools and methods, which have been described in the literature, for mechatronics design and the Quality Function Deployment group of methods. The design tools are described first, this is because an understanding of these is prerequisite to understanding many of the ideas applied in the QFD group of methods.

The second objective is to highlight the advantages and disadvantages of each method or tool with the aim of identifying which should be included in the new design methodology.

This chapter provides a contribution to the Ph.D. as it provides a justification for the design methodology, which is used later in the work, for the design of the Gamma Radiography robot.

2.2 Function based design tools

Function Based Design, an established technique (detailed in French[71], Pahl and Beitz[84], Hubka[82], and Roozenburg[95], among others) is based upon the ideas of functional decomposition. This is a process whereby the functionality of a product is considered and hierarchically broken down into sub functions. For example, the sub-functions of an automatic washing machine are, water filling, soap powder addition, drum rotation, and water emptying. These can also be broken down into sub functions, for example drum rotation could be decomposed into, clockwise rotation, anticlockwise rotation or spin. This form of decomposition leads to a hierachial layering of product operation. This layering can be reproduced in a diagram know as the functional structure (see figure 2-1, below).

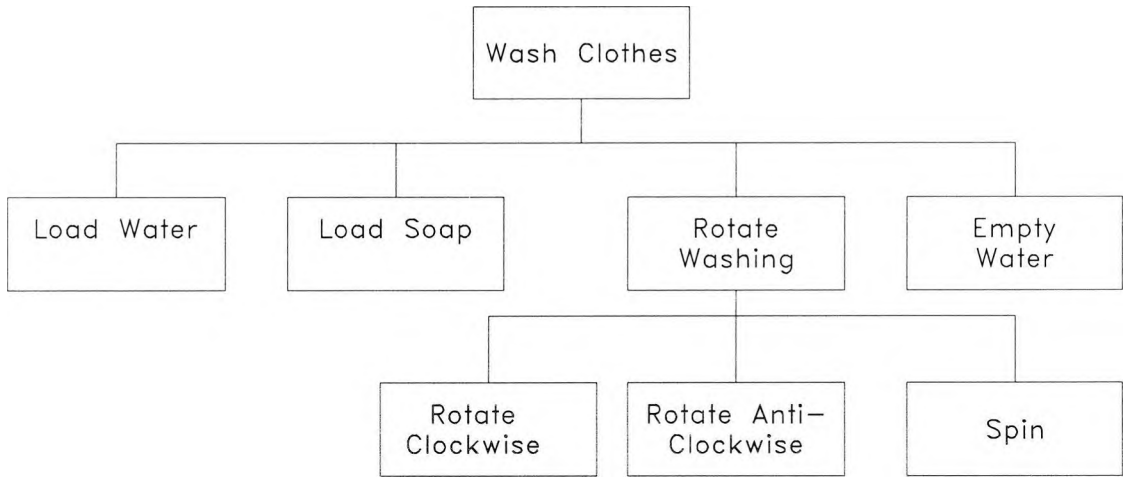


Figure 2-1 Function structure of a washing machine

This is particularly important with regard to mechatronics as it allows a modular approach to be taken, which enables the divisions between mechanical hardware, electronics and software. It also acts as an aid to project planning, because it allows teams of designers to concentrate on particular areas of the design, thus allowing some degree of concurrent engineering at the design stage.

Also, note that the functions are described in a neutral solution format. No indication is given as to how the function is to be achieved, for example, the function, load water, could be realised by a pump, connection under mains pressure or a separate water tank supply. The purpose of this is to ensure that the designer is not constrained by preconceived ideas when the search for solutions to functions begins.

2.2.1 Function trees, the law of vertical causality and function coupling

As stated in the previous section, the product functionality can be represented by a tree, which can be expanded to show the means of realising the various functions, providing the law of vertical causality is obeyed (Hubka[82]): "There is a causal relationship between the (purpose) functions of the system and the alternative means of realising it. Once the means for a function on one level is found it can be decomposed into sub-functions on a lower hierachial level, each of which in turn requires its choice of means".

The essence of this statement is that one cannot break a function down into sub functions until, a means of achieving the overall functionality of the function is selected. This is where 'transformation functions' and 'purpose functions' are appropriate.

A transformation function, is the neutral solution description of a function, for instance. the process functions relating to automatic tea making (Buur[90]) are, tea process with tea extract, normal tea process, tea process with perfusion. This is represented diagramatically by the use of a Function Means Tree.

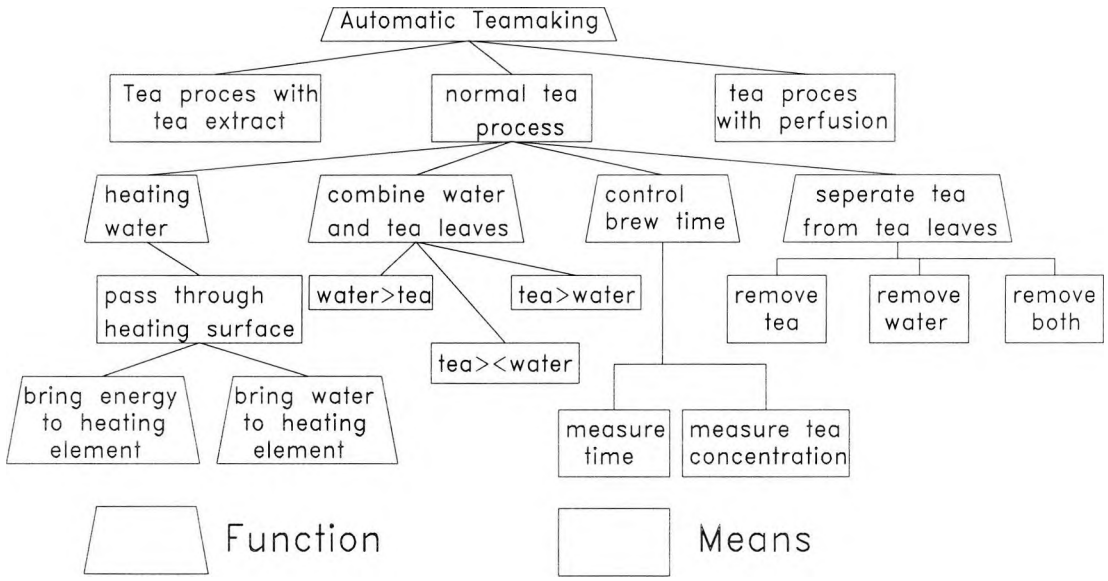


Figure 2-2 Function Means Tree for an automatic tea maker (Tjalve[1979])

From figure 2-2, it is seen that the law of vertical causality is enforced, because a function cannot be broken down into its sub function before a solution concept is selected. Function Means Trees are also a good example of forcing divergence and convergence in the design process (Roozenburg[95]). This is necessary because, if 3 or more solutions are discovered for each function and these are further developed into 3 or 4 further sub-functions from each possible function, the number of concepts to be considered gets very large. By selecting an appropriate solution, this increase in variant combinations of ideas is reduced and the function tree converges at each function stage. It only diverges again when looking for solutions to the function at a particular level.

Although not shown here, it is possible that certain purpose functions or solutions will contribute to the attainment of more than one function. This is called function coupling, the software equivalent is termed cohesion. The degree of function coupling which can occur

in mechanical design, is an important aspect as it increases the complexity of the design process. If high function coupling exists, it must be noted, as any change in a solution to a number of functions, means that numerous other changes may need to be made . To this end, Suh (90) develops two axioms which simplify the design process, these being:-

1) Maintain the independence of the functional requirements (ie generate solutions based on independent subsystems).
2) Minimise the information content.

In Joseph (96) these axioms are simplified to the unoriginal but very sound advice of a) do not try to get one component to do too many things; and b) keep it simple. Breaking down the dependencies of the functions makes the mechanical design process a lot more consistent with the decomposition processes associated with software and electronics design. This is because the degree of coupling between functions, in these disciplines is very low. This acts as an aid to technology allocation, as the various functions can, possibly be realised in either of the three disciplines used in mechatronics.

2.2.2 Theory of domains

The Theory Of Domains (Andreasan[80], Burr[90])is based upon four abstract areas:-

1) Transformation domain:- Solution neutral description of the function
2) Process domain:- Top level means of realisation of the transformation domain.
3) Organ domain:- Subsystem realisation of the process domain.
4) Parts domain:- Breakdown of the Organ domain into components.

These domains are mapped in two dimensions, going horizontally from the abstract to the concrete, and vertically from the vague to the detailed. The domain areas are not sequential and do not provide a structure to the design process, but they can be used to indicate status in the design process, see figure 2-3.

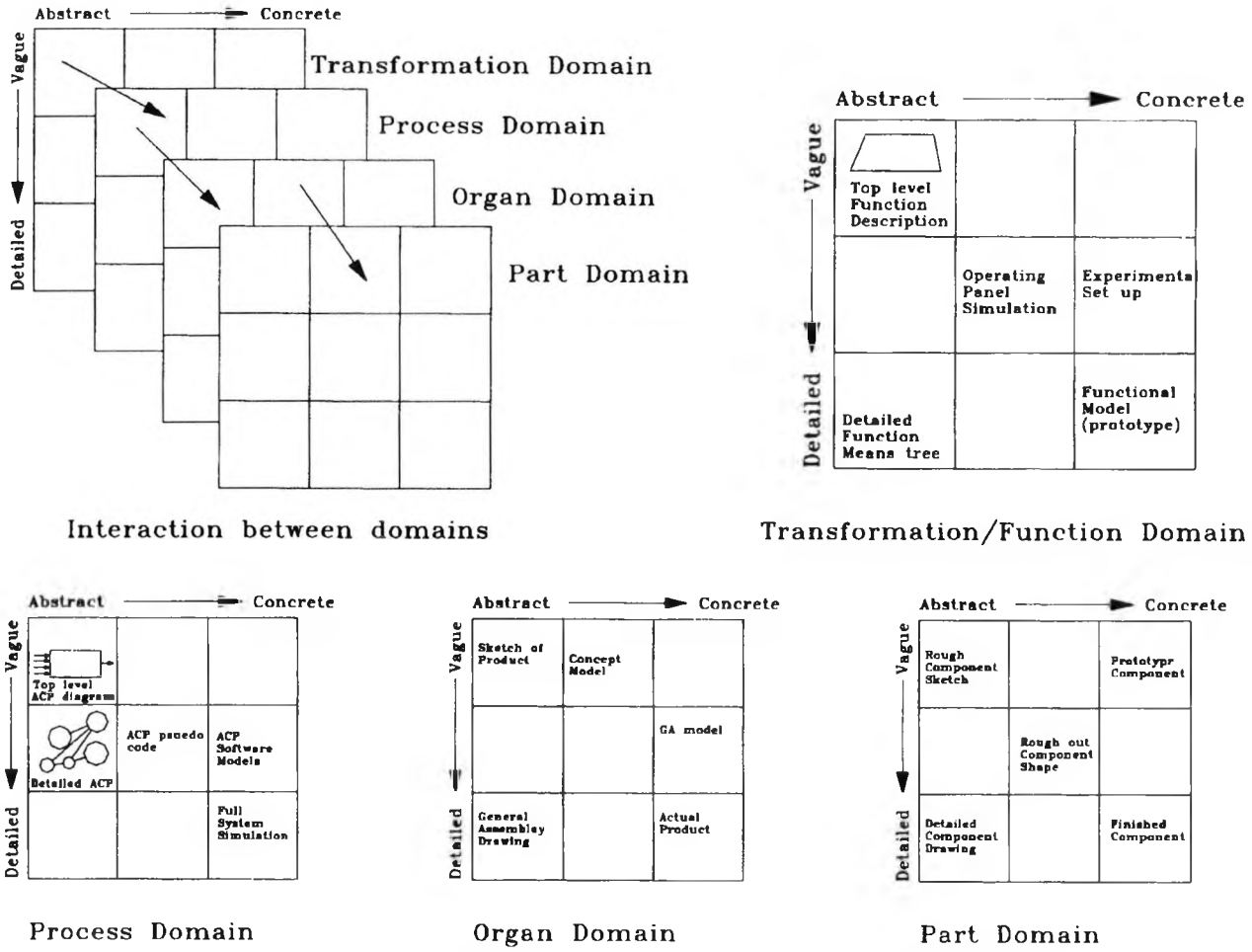


Figure 2-3 The Theory of Domains

2.2.3 Morphological charts

Morphological charts are one of the earliest design tools, Zwicky[48]. They operate at the organ and parts domain level and cover the means of realising a process function, which is described as a series of sub-functions, entered along one axis. The means of realising the various sub-functions are set along the other axis. By examining the chart it is possible to determine several complete conceptual solutions for the process function and to identify groups of functions which act in conflict with each other.

Feature	Means				
Support	Wheels	Track	Air cushion	Slides	Pedipulators
Propulsion	Driven wheels	Air thrust	Moving cable	Linear induction	
Power	Electric	Petrol	Diesel	Bottled gas	Steam
Transmission	Gears and shafts	Belts	Chains	Hydraulic	Flexible cable
Steering	Turning wheels	Air thrust	Rails		
Stopping	Brakes	Reverse thrust	Ratchet		
Lifting	Hydraulic ram	Rack and pinion	Screw	Chain or rope hoist	
Operator	Seated at front	Seated at rear	Standing	Walking	Remote control

Figure 2-4 Tabular, Morphological Chart, for a forklift truck (Cross[1989])

Figure 2-4 indicates a combination of solutions which achieve the functionality of a forklift truck. In this instance, the indicated solution for the forklift truck is that it will be remotely controlled, with an electric motor, the electricity delivered by rails, a belt will be used to transmit power and drive wheels, to stop the fork lift brakes will used, and finally the lift for forks will be screw driven.

2.2.4 Benchmarking techniques

One of the areas not addressed by Burr in his thesis (Buur[90]) was concept evaluation or benchmarking. This is a very important area as it enables the designer to select the 'best solution' to a problem in a logical manner. Many techniques exist, but the two

most common, which are examined here, are Pugh's Concept Selection (Pugh[81],[91]) and Cross's Weighted Objective Method (Cross[89]).

2.2.4.1 Pugh's concept selection

This selection technique employs a large visual chart, containing the available concepts as column headings and the selection criteria as row headings. One column is set as the concept benchmark. The other concepts are then evaluated against the benchmark with regard to the selection criteria and marked "+", if better than the concept, and a "-" if worse than the benchmark, or an "S" if equal. These evaluations are then summed up and used to rank the various concepts (see figure 2-5).

Concepts / Criteria						
A	+	-	+	-	+	
B	-	S	+	+	S	DATUM
C	+	-	+	+	-	
D	S	S	+	-	-	
E	-	+	-	S	S	
∑ -	2	2	1	2	2	
∑ S	1	2	-	1	2	
∑ +	2	1	4	2	1	
Ranking	2	3	1	2	3	2

Figure 2-5 Pugh's Concept Selection table (Pugh[1981])

2.2.4.2 Weighted objectives method

Cross[89] developed the weighted objectives method to take into account the differences between the importance of the various selection criteria. The chart used is very similar to that used for Pughs concept selection the only differences are that, no benchmark concept is set, and a weighting column is introduced alongside the selection criteria. In the example, shown in figure 2-6, three cars are compared under the headings of , fuel consumption, the cost of spares, maintenance and comfort. A weighting is associated with each requirement, depending upon its importance. The parameter to be measured is placed in the third column. Each of the vehicles are then evaluated under a further three columns, the first showing the magnitude of each function, the second a score (between 1 and 10) and the third a value, which is the product of the score and the weighting. The values for each vehicle are summed to give an overall utility value. In this instance Car B is the best option as it has the highest score.

Objective	Weight	Parameter	Car A			Car B			Car C		
			Magnitude	Score	Value	Magnitude	Score	Value	Magnitude	Score	Value
Low fuel consumption	0.5	Miles per gallon	33	2	1.0	40	4	2.0	36	3	1.5
Low cost of spare parts	0.2	Cost of 5 typical parts	£18	7	1.4	£22	5	1.0	£28	2	0.4
Easy to maintain	0.1	Simplicity of servicing	Very simple	5	0.5	Com-plicated	2	0.2	Average	3	0.3
High comfort	0.2	Comfort rating	Poor	2	0.4	Very good	5	1.0	Good	4	0.8
Overall utility value					3.3			4.2			3.0

Figure 2-6 Weighted Objectives Chart (Cross[1989])

The author observed that Cross's method is a more refined way of concept evaluation and selection. This is because there is a fine numerical evaluation of each selection criteria. The evaluation using Pugh's method is very coarse as no indication is given of how well a particular selection is met apart from +,S,-. With Cross, in this example, there are ten divisions. Cross's method can also be varied so that negative values can be taken into account.

2.3 Introduction to QFD family of methods

The basic QFD methodology was developed during the late sixties and early seventies by Dr.Yoshi Akao at Mitsubushi's, Kobe Shipyards, Japan. Many works exist on the

subject, the definitive work being Akao[90]. Other important contributors to the field include Hauser [88] , Clausing[89] and Pugh (Clausing[91]). Their work has resulted in a number of methods and tools which have been associated with QFD and led to a variety of enhancements and extensions to the basic methodology.

2.3.1 Quality Function Deployment (QFD)

The idea behind QFD, is that, "quality as perceived by the customer represents a major criteria for the entire product development process", Sontow[93]. This is started by taking into account customer requirements, i.e. factors considered important by the customer, such as the position of the exposure button on a camera. This is also referred to as "the voice of the customer", because the functions are described in the language of the customer. Various products within the same market sector are then compared and benchmarked against each other to determine the good and bad points of each design, and highlight areas of improvement. This process is carried out on a product planning chart, known as "The House of Quality", See figure 2-7. The functions of each part of the "House of Quality" (HOQ), and some of the terms common to the methodology, will now be defined.

2.3.2 QFD House of Quality

The House of Quality consist of a number of rooms, each of these rooms having a specific function and related to each other very closely

Room 1: In this section, the customer requirements are entered into the grid. These are the WHATS, implying what is wanted. It is important that they are entered in customer language, for example, the product must be "light to carry". This is to enable a qualitative assessment to be made, by the customer.

Room 2: In this section the customer requirements are translated into more specific technical requirements, for instance, the customer requirement that the product must be "light to carry", might be translated into, "the product weight must be between 200-300 grams". The entry's in this section are known variously as "corporate expectations", "substitute quality characteristics" or "engineering characteristics". In simpler terms they are the HOWS which realise the WHATS of room 1.

Room 3: The block matrix at the heart of the "House of Quality" is used to asses the strength of the relationships between WHATS and HOWS, detailing how well the

corporate expectations are met by the customer requirements, strong, medium, weak or negligible. For each corporate expectation there should be at least one customer requirement. If no relationship exists, then either a column or a row will be empty and it shows that market research or engineering did not address all the necessary features or technical aspects of the product.

Rooms 4 and 5: Both of these rooms are used for competitive benchmarking of the product against competitors, in the areas of customer requirements and technical specification. In room 4, the customers perception of existing products is displayed graphically , while room 5 contains benchmarking that is based on technical tests.

Both evaluations are compared to determine target values for the corporate expectations that satisfy the customer, by beating the competitors design or by being at least equal to the competitors values.

Room 6: The roof of the house of quality, is used to show the interaction between corporate expectations, detailing either conflict or synergy. This allows the product development team to identify problems in the product specification early in the design process.

Room 7: This room details the importance of a particular customer expectation. It contains the sum of the values associated with the strengths of relationships, between customer requirements and corporate expectations multiplied by the relative importance put on a corporate expectation by a customer.

Room 8: The targets for the corporate expectations, derived from the details in rooms 4 and 5, are entered into this section.

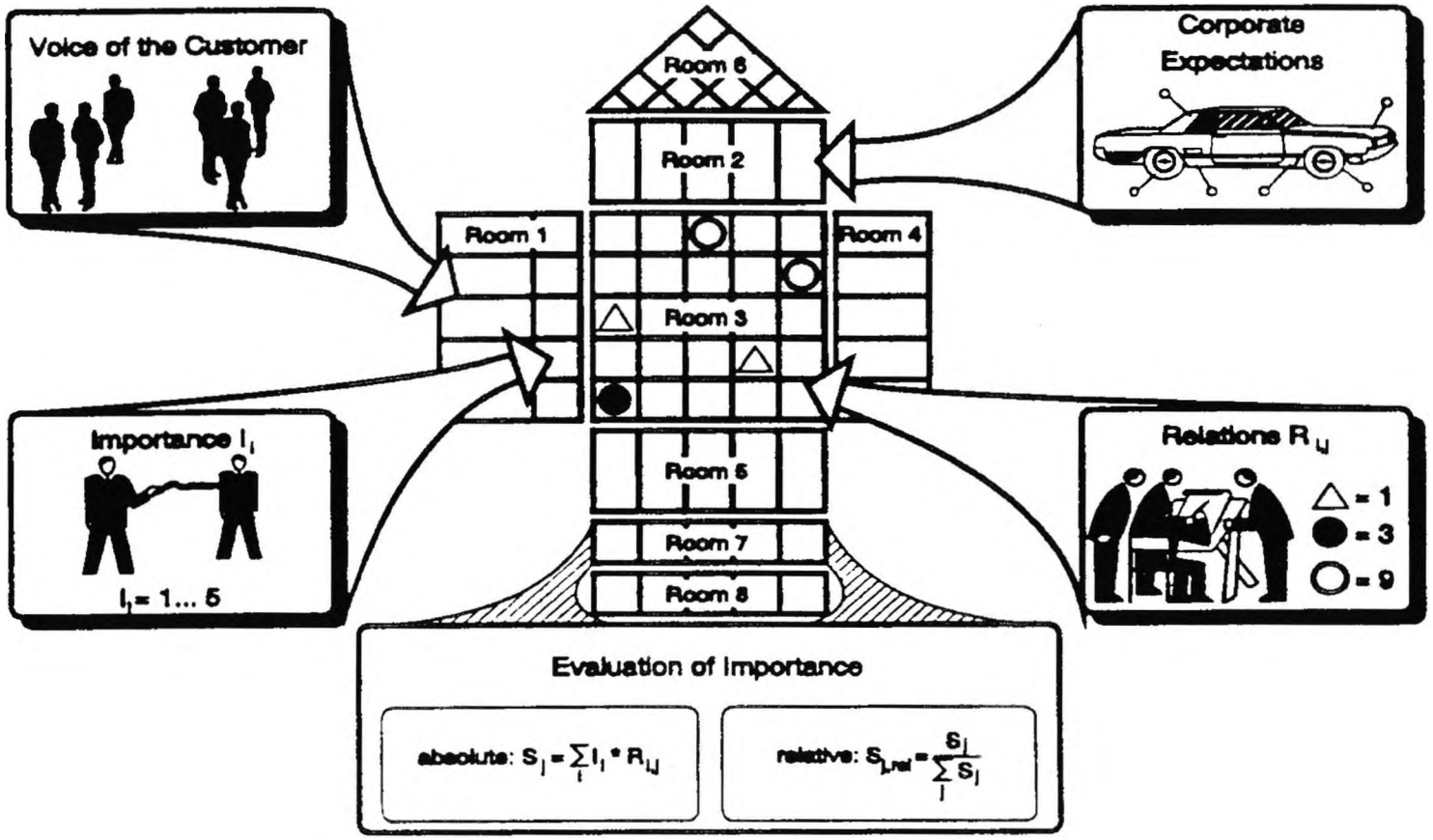


Figure 2-7 The QFD House of Quality (Sontow[1993])

2.3.3 *Macro procedure for basic QFD*

The above activities are now carried forward to a further 3 stages, which cover design, process planning and productions operations planning. At the end of each stage the corporate expectations relating to that stage become the customer requirements for the following stage.

For instance the result of the product planning stage should be a realisable product specification with technical targets set by the customer requirements from room 8. This product specification is then translated into the customer requirements for the design stage. Room 2 is now used to detail design functions can be used to realise the customer requirements. The outcome of stage 2 is the part design of the various components. This is used to guide the design of the process plan. Finally the production operations that are required in order to manufacture the piece parts are determined.

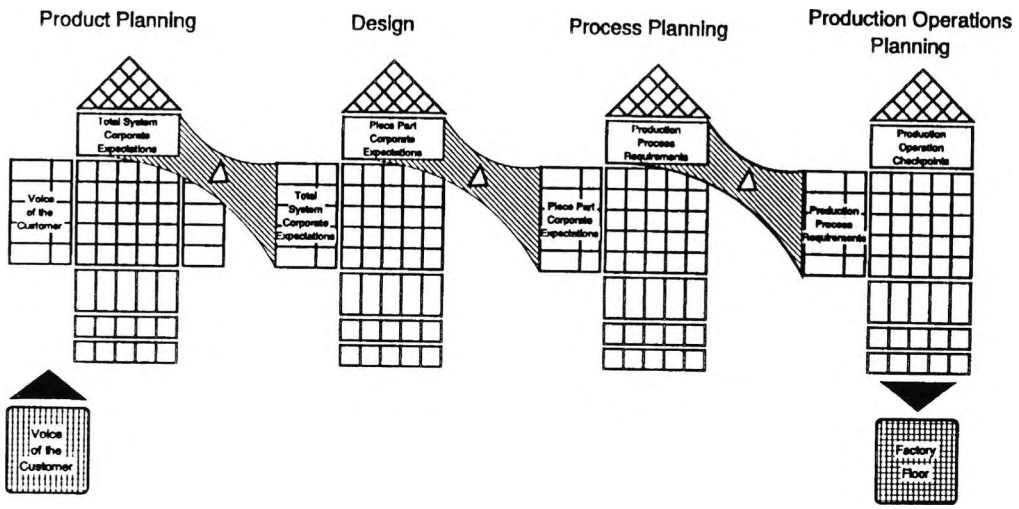


Figure 2-8 The QFD process (Sontow[1993])

The basic form of QFD has been shown to exhibit the following main advantages:-

1) The communication between all the functions of a company involved in product design and manufacture process is improved (Sontow[93]).
2) The use of customer requirements to guide product development leads to an improved product (Sontow[93]).
3) House of quality charts can be used as data-bases that record design decisions which can be used for subsequent product redesigns, (Clausing[92]).

The application of basic QFD can lead to a substantial improvement in the design and manufacture process, however, the process has several weaknesses.

1) There is no systematic way in which piece part, process and production operation parameters are selected as they determined by engineering knowledge rather than logical benchmarking (Sontow[93]).
2) The use of target values derived from customer requirements and the comparison of competitive products can lead to unfeasible design specifications (Pugh[91]).
3) Deriving piece part design from initial product specifications is nearly impossible because the nature of most products requires many parts and there interactions to be considered. There is more than one way to achieve the same result. In this version there is no concept evaluation procedure in place to examine numerous products (Bleakley[98]).
4) Basic quality function deployment is only good for products that have already evolved and are in competition with other products of a similar ilk.

2.3.4 Enhanced Quality Function Deployment (EQFD)

It is apparent from the above disadvantages that to apply QFD properly improvements were necessary. QFD was modified during the early nineties by Clausing and Pugh (Clausing[91] & [94]). These deal with the integration of Pugh's Concept Selection (Pugh [81]&[91]) and the development of multi-level deployment charts, known as deployment through the levels, with QFD.

Deployment through the levels refers to the hierachial breakdown of the design phase into 'Total System Design', 'Subsystem Design' and 'Piece Part Design'. This author observed that these can be directly related to the ideas of the, Function Domain, Organ Domain, and Part Domain, advocated by Andreasan [81]. The only difference being that Andreasan works in two dimensions, so that he can map the product development process. The ability to break the product down into subsections and parts allows process planning and production planning also to be broken down into sub-stages.

The meta-level chart used during this process is two-dimensional. It shows the shift from the voice of the customer to the factory floor in the horizontal dimension, whilst the design development progresses from the total system level to the parts level. The charts used are essentially the same as those used for Basic QFD. Figure 2-9 shows the layout of the charts used for this process.

The use of Enhanced QFD enables concurrent engineering to be carried out as it allows the product development team to start with process planning before the design phase is completely finished. (Sontow[92]).

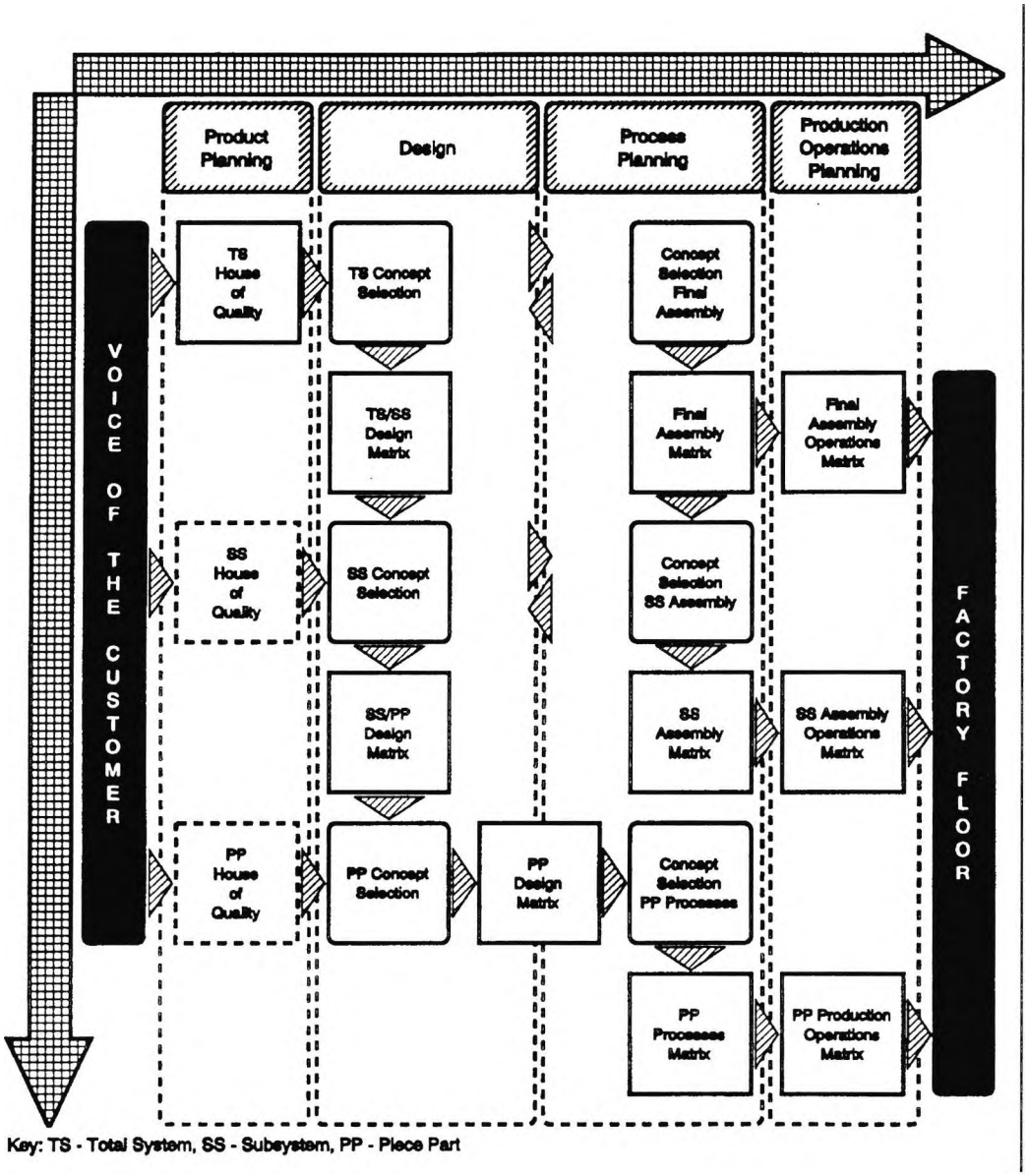


Figure 2-9 EQFD process diagram (Sontow(93))

Pugh's concept selection is used as a benchmarking tool to evaluate different concepts against the system specification targets, derived from the original QFD analysis.

These additions gave EQFD a number of advantages over QFD. These advantages are principally due to the fact that EQFD supports the development of complex products through the use of the subsystem QFD charts. Another advantage comes through the use of Pugh's concept selection. This is because it supports the development of

conceptually dynamic products , i.e. products which change rapidly due to, competition, technological advances and trends in styling. Finally, because the process has the ability to ape the characteristics of concurrent engineering shorter product development times can be realised.

Despite these advantages there are problems with EQFD, this is due to the lack of guidance, for the user in, firstly, how to split the 'customer requirements' of each subsystem from the design stage QFD chart, 'corporate expectations'. And secondly the partitioning of the ' Subsystem expectations' or design functions, for each subsystem. This is because they rely on engineering judgement and experience with little, or no, formal analysis of the subsystems.

2.3.5 Requirements and Failure Causes, Analysis (RFCA).

RFCA is detailed in Sontow[93], it is an extension of EQFD which includes a number of secondary design tools such as Fault Tree Analysis (FTA), Failure Modes Effects (FMEA), the Amplification Sub-Tree (Pandey 92), and Taguchi's System of Quality Engineering (Taguchi [90]).

Amplification Sub-Trees are a method used to define or extend a function from its verb-noun relationship. The function is broken down into a series of sub-functions (amplifications) which describe the sequence of sub-functions required to achieve this and the customer requirements.

In figure 2-10, taken from Sontow[93], the function 'make a copy' is divided into:-

1) Customer requirements which affect the quality of the function from the customers point of view (in the top box).
2) Sub functions (sub amplifications) which are used to produce the function (lower box).

These sub functions and customer requirements are used to identify the functions which need to be addressed during the design and development of a particular part.

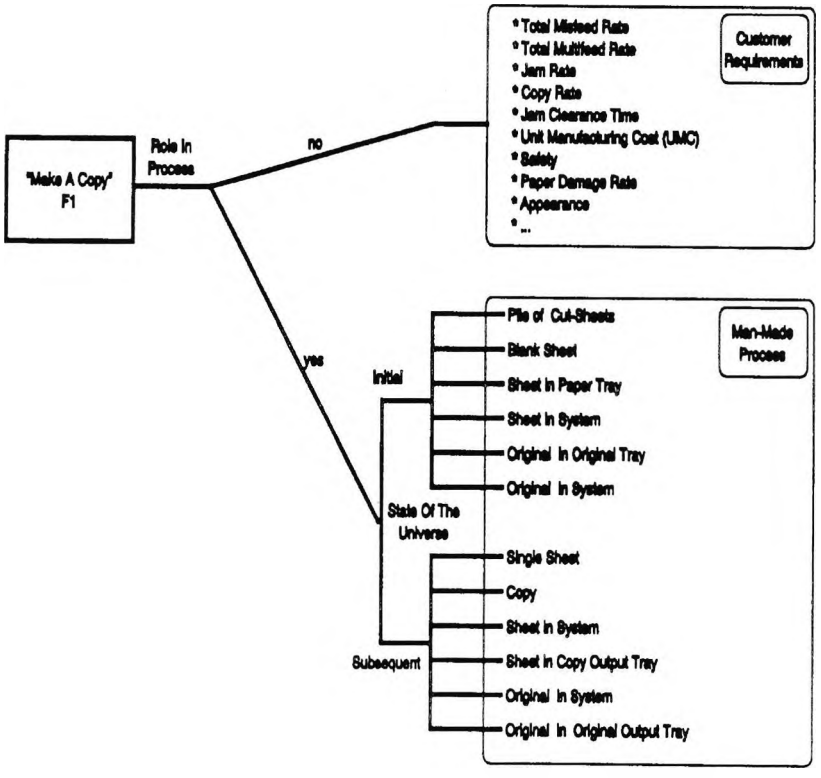


Figure 2-10 Amplification Sub-Tree (Pandey[1992])

Taguchi's System of Quality Engineering, serves "to reduce economically the variations of a product's ideal performance in the customer environment, making it robust" (Phadke[89]). It measures the loss of quality in a product which occurs due to its deviation the product's target performance characteristics. This can be related to financial loss, either to the company, by surpassing the performance target, or to the customer and indirectly to the company, by a product not meeting its target (see figure 2-11).

The object of the system is to identify and minimise the effect of "noises" on the product. These come in a variety of guises, generally falling into three categories.

1) Variations in the conditions of use.
2) Product variations
3) Product deterioration

In its simplest terms, noises are identified and the product analysed using experiments to determine the effect of variation of these noises on the functionality of the product, subsystem or part.

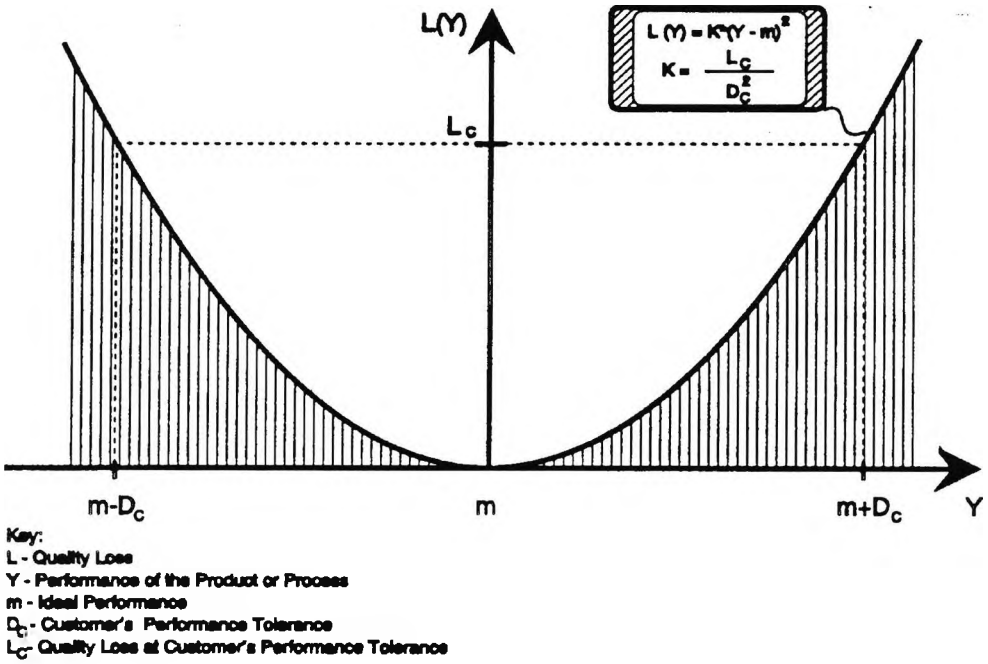


Figure 2-11 Quality Loss Function (Phadke[89])

FMEA and FTA are tools used to identify and analyse the effects of failure and how failure propagates through a system. A full description is given in sections 4.3.1 and 4.3.2.

The objectives of RFCA are to provide:-

1) Systematic determination of the products ideal performance, guided by customer requirements.
2) Systematic determination and avoidance of events that cause the deviation from the products ideal performance, as early as possible during product development.
3) Systematic analysis of the relationship between causes and effects of undesired events
4) Evaluation of design requirements and undesired events in order to focus on critical aspects during quality planning.

The first phase in RFCA is product planning, which is used to determine the customer requirements. These are structured with regard to the system level and then finally translated into design requirements for the product.

The design phase follows. In this stage, various concepts are developed for the product components, based upon the component functions. These concepts are evaluated with respect to the causes and effects of failure and attainment of the customer requirements for a particular function.

Phase 3 examines the process planning of the component, taking into account the design requirements from phase 2. An analysis of the various noises which affect manufacturing quality is also carried out. Once this has been completed optimal assembly and production processes are selected.

Phase 4 is carried out concurrently with phase 3. It is used to carry out operations planning for critical assembly, production and quality assurance. Quality assurance techniques such as Statistical Process Control are employed to keep the noise factors, identified in phase 3, within specified tolerances.

2.3.6 Design Function Deployment (DFD)

A basic description of Design Function Deployment based on, Edney[93] was given in section 1.1.1.5.2. This was the basic methodology employed during an extensive case study, carried out by the author in conjunction with Chamberlain and Edney, (appendix A), the results of which are summarised in section 1.1.2.4. It is now apparent from the work covered here that this interpretation shows many similarities with QFD. Further work detailed in Sivaloganathan[95], Evuobuomwan[95] put the work into a wider context, by including a number of computer based tools which are used at various stages during the four phases of product development.

The basic process of DFD can be seen in figure 2-12, which shows the various stages in the product and production development process. The processess shown are top level stages, they are supported by a number of existing, paper and computer based tools covering:-

1) Design Tools:- such as, functional analysis, morphological charts, concept and selection tools, etc.
2) Computer based design tools:- 2 and 3 D Computer Aided Drawing, Finite Element Analysis, Solid Modelling, Circuit Simulation tools (SPICE), etc.

3) Tools for Design for X.
4) Computer Aided Process Planning, Scheduling, Statistical Process Control etc.
5) Quality analysis tools, such FMEA, FTA, and Robust Engineering Design techniques.

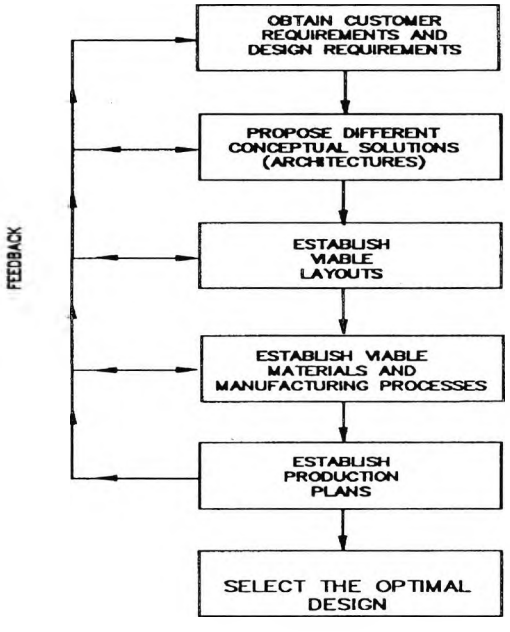


Figure 2-12 DFD design process (Evuobuomwan(95))

Beneath this level exist a further level of databases and rulebases/expert systems, which theoretically give the engineer access to a vast mine of information and knowledge about how to use it. All these features are indicated in figure 2-13.

The overall intention of DFD, described in Evuobuomwan's[95] Ph.D. thesis, was to integrate the entire product design and development process into one computer package, consisting of a number of integrated modules. Although the modules and their functionality are described, no indication of how these modules fit into the matrix based framework of DFD is shown, or implemented. Further work was carried out by the Engineering Design Centre, at City University, into the computerisation of DFD but this proved to be too large a task, and the project was unsuccessful.

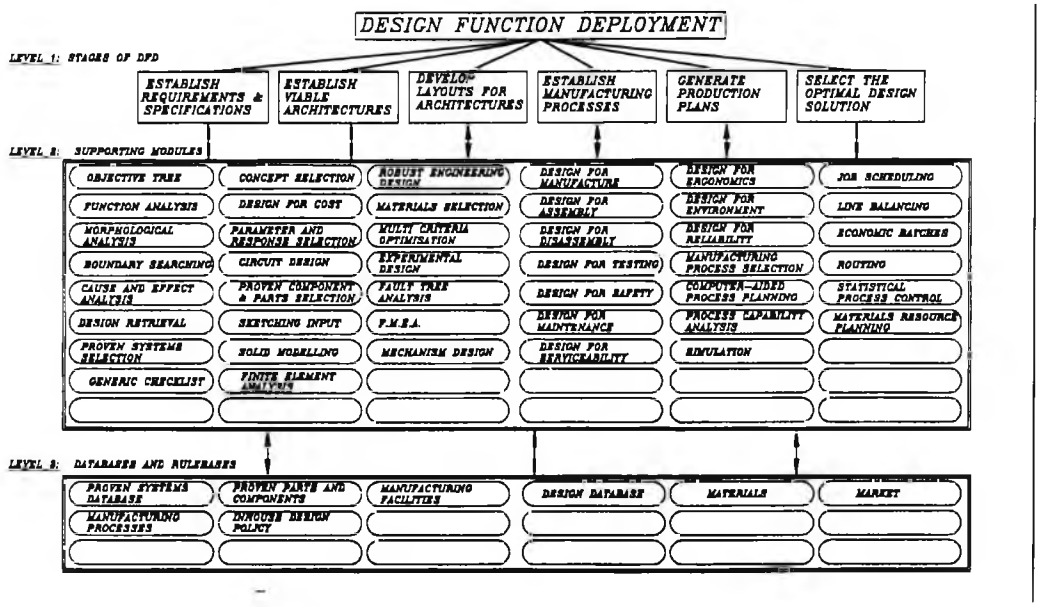


Figure 2-13 DFD Three layer model of design tools (Sivaloganathan(95))

2.4 Summary of Review and conclusions

As already stated in section 2.3.1 basic QFD had a number of drawbacks relating to its use:-

1) No systematic approach to piece part, process and production operation parameters are selected.
2) Deriving piece part design from initial product specifications is nearly impossible, due to the numerous interactions between parts.
3) Basic quality function deployment is only good for products that have already evolved and are in competition with other products of a similar ilk.

To counter these problems, Clausing and Pugh improved basic QFD by applying the principles of Total Design (Pugh[91]) by splitting up the top level chart into separate charts for each sub system and part. Concept evaluation was achieved by the use of Pugh's Concept selection. This method was called Enhanced Quality Function deployment, or EQFD.

EQFD evolved into Reliability and Failure Causes Deployment, otherwise known as RFCA, Sontow[93]. The advantage of this development is that the effects of production noises on product quality could be assessed, using Robust Engineering Design techniques, along with the reliability of the product, which could be evaluated

using FMEA and FTA. The use of some computer tools to help with this analysis was proposed, but no useful details were given.

Design Function Deployment (Evuobonwam[95]), took the ideas of EQFD and linked them with a number of already existing, paper and computer based design tools. The emphasis of DFD seems to be in the design development stage as it allows the user greater freedom to develop concepts in a structured manner.

It was felt that basic QFD was not a suitable design method to employ for the design of the gamma radiography robot, this was because of the shortcomings indicated above. EQFD however, did answer many of the problems of basic QFD, but in the form described in Sontow[93], it was still not suitable due to the lack of integration with a fault or reliability analysis tool. RFCA answered this problem, but due to the highly complex nature of the method, as described in Sontow[93], and other problems, observed by the author, listed below, it was not felt suitable for this project.

Authors observations on RFCA:-

1) It does not carry out a truly proactive approach to safety.
2) It requires that the design team has detailed knowledge of previous products in the line, in terms of failure issues and of how to solve them This is apparent from the description of the methodology, as applied to the design of parts of a Xerox photocopier in Sontow[93].
3) There seems to be very little scope for the development of new concepts, the examples given mainly dealing with incremental design.
4) The process is very heavily paper based, with numerous charts being created for the design of even a simple component.

DFD also integrated a number of design tools into a methodology, however, to successfully implement computerised DFD it needed to be developed and a large number of software tools needed to be integrated with it. This project was only partially completed because although the vision was in place, the scope of the project was too large. Finally, no examples exist about how to integrate the various tools into the DFD framework, whether computer or paper based (Bleakley[97]).

The majority of the design tools referred to at the beginning of this chapter do have parallels within the QFD family of methods. EQFD uses Pugh's concept selection whilst DFD suggests Morphological Charts and Cross's Weighted Objectives method. RFCA uses Amplification Sub-Trees, whilst DFD employs Function Trees. None of the

present group of QFD methods use the theory of domains or obey the theory of vertical causality, which the author feels is a major error. This is because the theory of domains and the law of vertical causality does allow the design process to be controlled and mapped properly. This was a found to be a problem when the application study, section 1.1.2.4, Chamberlain(95), was carried out, as the House of Quality charts that were produced were very large and tedious to analyse. They were so awkward to employ that the use of them was halted half way through the project.

Because of the above problems with DFD and RFCA, the author decided to base the new design method on EQFD. This appeared to be reasonably robust, and allowed for a modular design strategy, which is important for mechatronics where technology allocation of functions is a big issue. The authors early work on the application study, detailed in Chamberlain(98), highlighted some of the problems of implementing EQFD. It is felt that the application of The Theory Of Domains and the Law Of Vertical Causality would allow the process to be controlled, so that smaller, more focussed charts could be produced and the design mapped properly. Morphological charts, Function Trees, and Cross's Weighted Objectives, provide logical tools to realise functions and select appropriate solutions to those functions, because of this, they too will be included in the design methodology.

3 Systems modelling techniques

3.1 Introduction

A systems approach to the design of mechatronics has been advocated in the literature many times, Buur[90], Seward[96], Salminem[90],[92], Heilala[92], Bradshaw[90], Fischer[90], French[90], Chamberlain[95] and Cop[92]. This is normally achieved using a functional analysis approach and, in many cases, this approach has involved the creation of some kind of paper based systems model. Buur[90], as mentioned previously, employed function diagrams with Finite State Machines. Edney[96] used a variation on ACP diagrams(Mort[90]). Salminem[90],[92] and Heilala[92] employed SA/SD(Ward[85]). Kecskemethy[93] developed an object oriented tool set called Mobile for the modelling of mechatronic systems. Gawthorp[92] employed Bond Graphs, an energy based representation system for the modelling of mechatronics. This work was used as the basis for the Scheme Builder project, Bradley[93], which developed the bond graph as a computerised energy modelling system for mechatronics design.

From this brief review, it is clear that the concept of modelling mechatronic systems as an aid to the design process is not new, the main reason for the modelling being to asses the ability of a design to meet its specification. It is also clear that there are a large number of methods which have been used to represent the models.

Early approaches to mechatronic models have either concentrated on the system functionality as a series of state charts, (Heilala[92]), or an energy model (Gawthorp[92]). Very little thought has been given to modelling the control aspect of the mechatronic system, which is embodied by the computer and the control software and integrating all three views together. It is only recently that an integrated approach to modelling of the entire system has been taken (Bradley[96], Bleakley[97], Shakeri[97]). Bradley [96], in private conversation with the author, proposed the use of Timed Petri-Nets for the state modelling of systems behaviour in the Scheme Builder project, thus linking together the logical and behavioural views. This is a view repeated in Shakeri[97] who identified the logical and behavioural model, and proposed linking them with the theory of domains (Andreasan[80]). Bleakley[97] shows three views of a

computer controlled DC motor model based upon a system model, a logical model and a behavioural model.

The integration of these views are as important as the functional view because factors, such as, software logic, safety checks, sampling times and settling lags, and how they are implemented, have a significant impact on the overall functionality of the system. This leads to a three model view of the system:-

1) Top-level, partitionable system model, which shows the overall structure of the system and sub-functions figure 3-2. This view should cover physical system elements shown in figure 3-1.
2) Software or logical, behaviour model which describes the functionality of the digital and software elements within the system model table 3-1.
3) Energy flow behavioural model which describes the continuos behaviour of electro-mechanical, hydraulic or pneumatic elements within the system table 3-1.

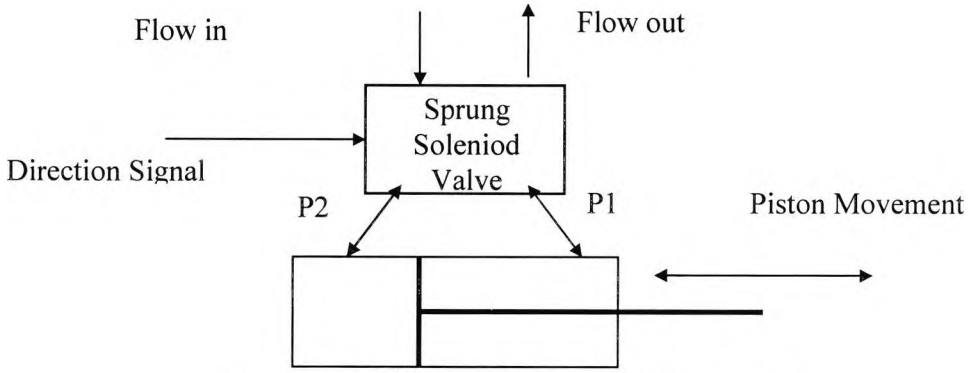


Figure 3-1 Schematic of Hydraulic ram system

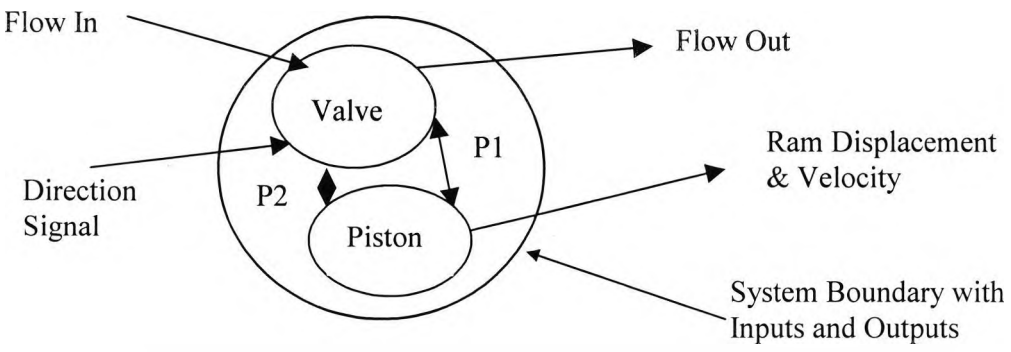


Figure 3-2 Top and second level system diagram of Hydraulic ram system

Logical model (state based)
If direction signal = 0, P1 = +ve Flow, (ram retracts)
If direction signal = 1, P2 = +ve Flow, (ram extends)
Behavioural model (equation based upon continuos real world behaviour)
Ram speed of extension α Flow rate and ram volume

Table 3-1 Logical and Behavioural model of Hydraulic Ram

Figure 3-2, is a top-level system drawing, which defines the systems boundary and encapsulates the inputs and outputs of a hydraulic ram. Table 3-1, details the rams behaviour, note how the control information is state based, whilst the actual ram behaviour is treated as continuos.

In the schematic diagram of a hydraulic ram system (figure 3-1) component details are given, such as port addresses, and components, partitioning of the system into the ram and the solenoid. This information can be hidden in the top level system diagram. as the user does not require it, however, this can be added at a lower level of system abstraction, figure 3-2. When this occurs the model changes from an essential model (barest amount of information required to describe a systems behaviour), to an implementation model (details means of system realisation).

Apart from the Scheme-builder project (Bracewell[95]), a computerised design tool for conceptual design and modelling, very little work has been published on the computerisation of these models and their utilisation. The objective of this chapter is to review the various systems modelling techniques that have been used, select appropriate tools and then find a computerisation method to implement them.

3.2 Software tools for system design

The majority of the tools to be examined are derived from software modelling techniques for real time systems. This seems to be the approach suggested by the literature(Bradley[93], Salminem[92], Heilala[92], Buur[90]). The reason for this choice is that a mechatronic system is by its very nature a real time system, and numerous tools have been developed in this field, the majority of them being graphical in character.

Graphical models are very useful because they can convey vast amounts of information concisely and compactly, a view taken by Ward[85] and Yourdon[89] who emphasise the old adage " a picture is worth a thousand words". They also allow the system to be

represented on a number of different levels in a hierarchical manner, thus allowing modelling along the same lines as a systems, function means tree, as described in section 2.2.1.

3.3 System Models

This section is devoted to methods that can be used for the system level description, of the product under development. It is broken down into two main groups, these being function based models derived from structured analysis techniques, and Object Orientated OO) methods. Both types of model are graphical in nature.

3.3.1 Function based modelling

Function based models are developed from the top-level functionality of the system, by setting the system boundary for the overall systems and defining all the inputs and outputs involved with the system. Various subsystems are identified and the interactions between them. This is the first level of hierarchy, each subsystem is broken down further, until the lowest level is reached. At this time, system behaviour is defined and the system rebuilt up from the bottom. This approach is called Top-Down, Bottom Up. Three forms of functionally based methods will be examined, Activity, Channel Pool diagrams (ACP) /MASCOT, Jackson[75], the Structured Analysis Design Technique (SADT), Ross[77], family of methods, and Jackson Structured Programming (JSP) (Cameron[89]).

3.3.1.1 ACP diagrams and MASCOT

Activity Channel pool diagrams were first introduced during the mid 70's. They were developed from Philips diagrams (Philips[67]), which were one of the first attempts to apply a modular approach to the design and integration of complex programs. In this instance, the application was the design of computerised radar systems.

ACP diagrams were the tool used to document the design methodology associated with MASCOT (Modular Approach to Software Construction, Operation and Test). The methodology, which was developed at RSRE Malvern, originally came from the requirement to develop complex, concurrent programs, running under real-time constraints for military applications (Jackson[75]). The basic notation is very simple, consisting of four symbols, see figure 3-3, Activities, Channels and Pools.

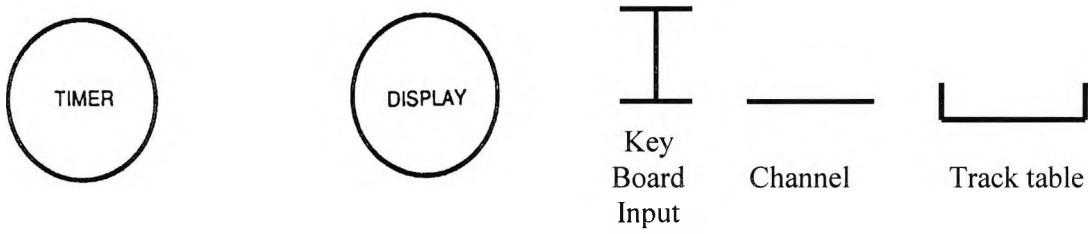


Figure 3-3 Key for ACP diagrams

ACTIVITIES contain the run-time activity of a process and are represented by a circle, they communicate with each other and the outside world through shared data areas called IDAs (Intercommunication Data Area). Two types of IDA exist. A CHANNEL carries a flow of data, either to or from the outside world, i.e. input or output, Channels are identified by an I section connected to a line. The I section represents the input output medium, whilst the line represents the information flow. The other type of IDA is a POOL, which represents a container for static data, i.e. a file or array. A top level description of each function or component name is noted by each representation of that object or process.

A simple example of an ACP diagram is shown below, see figure 3-4, it symbolises a computer controller used to control the rate of a chemical reaction. The Interface IDAs, Keyboard, Sensor, Printer and Valve interact with the outside world. Information is passed between the various system elements via Channels, A Pool is placed between the keyboard and the control Activities to indicate a file or lookup table. The Control Activity takes information in via the Temperature Sensor, and Keyboard, information is outputted to the Valve, Keyboard and Printer. The behaviour of the control algorithm is buried within the Activity, Control, effectively encapsulating and hiding the system behaviour, a technique employed in Object Orientated Programming.

In the diagram, no indication is given about how to implement the system behaviour. It is a basic, top level description of the system. The version illustrated here, relates to MASCOT 2 (MASCOT[80]). Problems were found this version, principally in the lack of the methods ability to support hierarchy, Bennet[90]. Improvements were made to MASCOT 2, and MASCOT 3 was developed MASCOT[87], Bennet[90], MASCOT3[86].

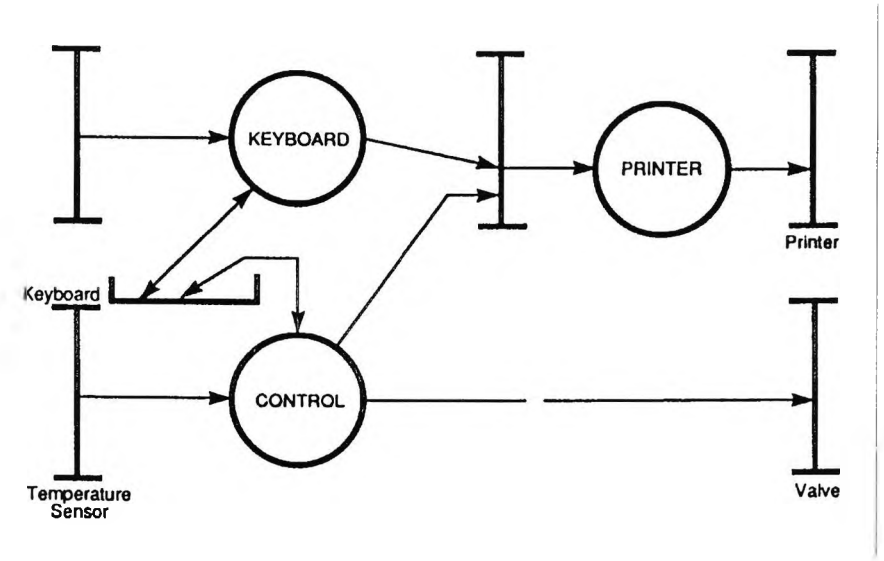


Figure 3-4 ACP model of computer controlled chemical plant (Bennet (90))

MASCOT 3 provided a means of representing hierarchy and passing data between levels of hierarchy. This was achieved by implementing interfaces for the data flows at the POOLS and ACTIVITIES, enabling the data to be labelled formally. Further extensions to the MASCOT 3, methodology produced an Object Orientated Version called MOON, (Hull[91]), which also employed Jackson Structured Programming, (see section 3.3.1.3) MASCOT has also been associated with Modula 2 (Budgen[85]), Timed-Petri-Nets, (Hull[94]), Ada, (Jackson[86]), Occam, Pascal and Coral 66, (Knowles[90]).

3.3.1.2 Structured Analysis and Design Technique (SADT)

SADT (Ross[77]) was developed around the same time as ACP diagrams, however, it developed a lot more rapidly than ACP, probably due to its coverage in the academic press, (Yourdon[79], and De Marco[79]).

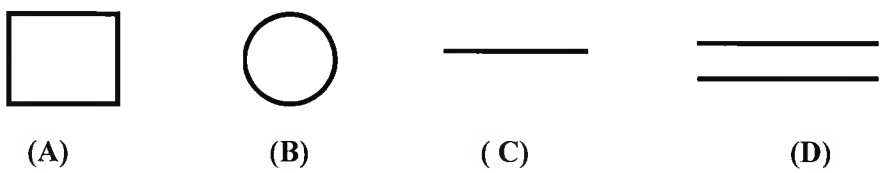


Figure 3-5 SADT & Data Flow Diagram symbology

The basic notation consists of four symbols (See figure 3-5), an external agent (a), function which processes data (b), signal or data flow (c) and a permanent data repository (d). Very similar to ACP diagrams, however SADT supported hierarchical decomposition from the very beginning.

The initial stage of program design is the definition of the system boundary, inputs and outputs. This is known as the context diagram. Further functional decomposition takes place at the next level down, providing the basic system structure. Functions are approached in a black box manner, which transforms inputs to outputs. Data is transferred between functions by arcs, which connect the functions. These diagrams are know as Data Flow Diagrams or DFD's (not to be confused with Design Function Deployment). Data dictionaries are used to define the various data stores and the data which is being transferred. The data dictionary can be used as a template for describing the program functions for writing code.

Extensions to the basic notation were quickly developed for real-time systems, which led to DARTS (Design Approach for Real-Time Systems, Gomma[84],[86]), and SDRTS, (Structured Design For Real Time Systems, Ward[85]). This involved identifying and synchronising the various processes, which were real-time. Ward[85], [86] extended the notation further, by applying a control structure and using STD's to define process behaviour. Hatley[87] added control bars, which were used to represent event occurrences.

An example of the application of SADT, according to the Hartley notation, is given in figure 3-6. Figure 3-6a shows the context diagram for a cruise control system, both data (solid lines) and control (dashed lines) flows shown, external objects denoted by square boxes. Figure 3-6b expands the context diagram into its sub-processes, showing the control and data flows and the control bar, CNT_1, the behaviour of this control bar would be encapsulated in a State Transition Diagram, (for a description of State Transition Diagrams, see section 3.4.1).

Because of the popularity of this series of techniques, a number of CASE (Computer Aided Software Engineering) tools exist, such as Excelerator, Team work, and Software Through Pictures. These tools provide a computerised sketch pad by which diagrams can be created and verified.

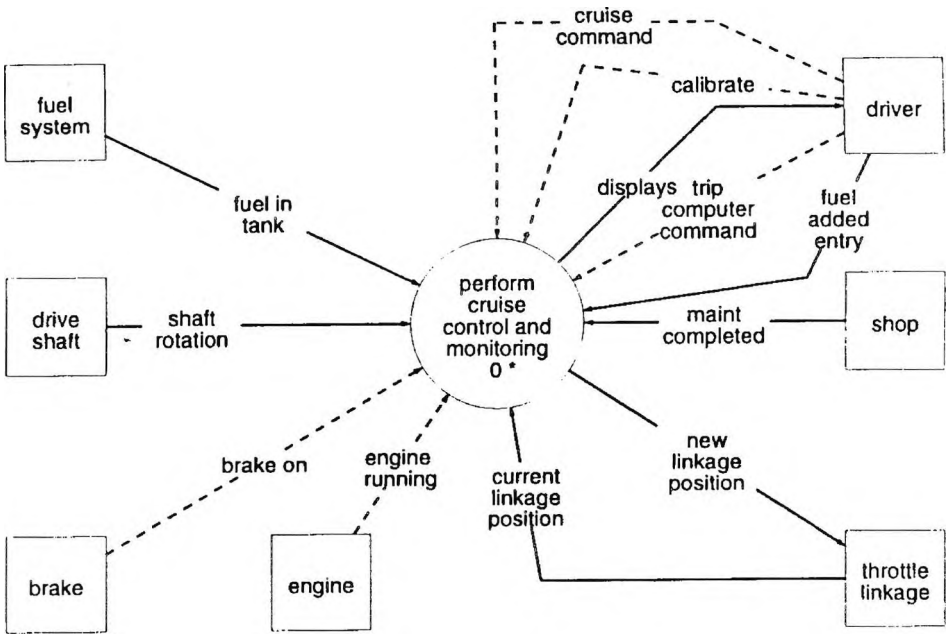


Figure 3-6A Cruise Control System, Context Diagram (Laplante(93))

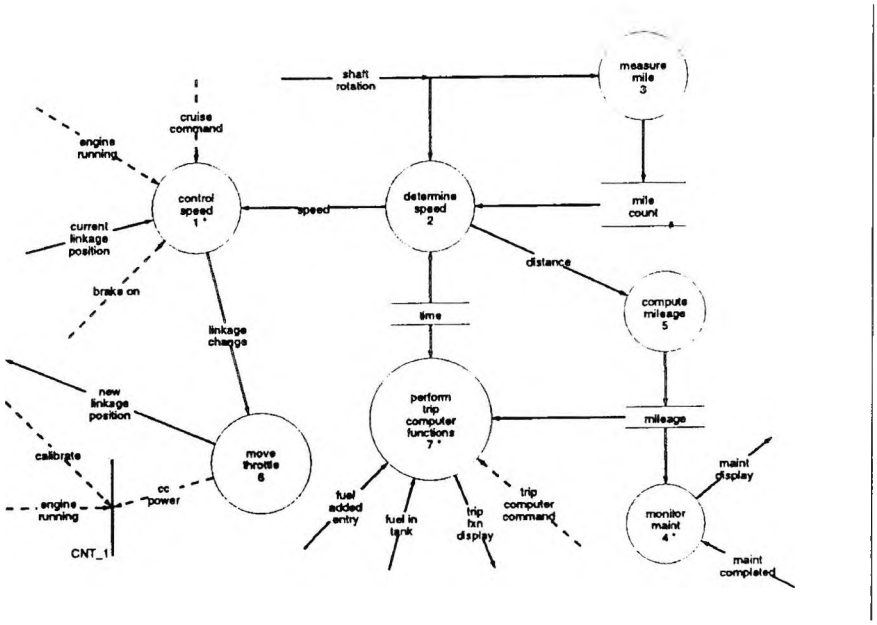


Figure 3-6B Sub Level, Data Flow Diagram of Cruise Control System (Laplante(93))

3.3.1.3 Jackson Structured Programming

Jackson Structured Programming (JSP), Jackson[75], Ingavaldsson[79], Cameron[86], was developed during the mid seventies. It provides a structured approach based upon the entity life cycle of the process to be produced. It is represented as a structure chart which shows the process life-cycle, an example of an elevator control program given in figure 3-7.

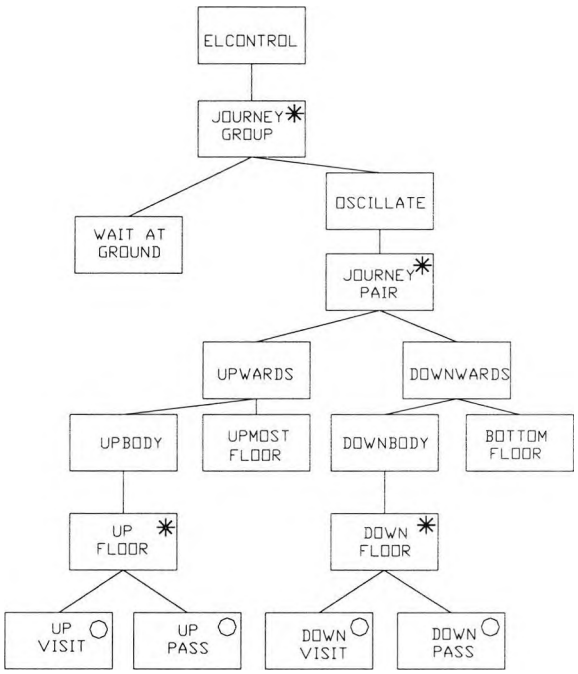


Figure 3-7 JSD software model of Elevator Controller (Laplante(93))

The top-level description is given as Elevator Control, the next stage, to define if a journey should be taken or whether the lift should move. If the lift is to move, the direction a journey should take is decided, for example. Operations or processes are represented by square boxes, which are further described by decomposition. At each level, operations are executed from left to right, iterative executions marked by a * and selection by O.

JSP is very logical and easy to use, however it does not support concurrent operations or timing and there is no representation system for components or energy flows.

3.3.2 Object Orientated methods

In the early nineties, a number of Object Orientated methods started to appear. Among the best known are, HOOD, (Hierachical Object Orientated Design, Robinson[92], Coad-Yourdon(Yourdon[91],[94]), OMT (Object Modelling Technique, Rumbaugh[91]), Martin/Odell (Martin[91]), BOOD (Booch Object Orientated Design, Booch[91]), Objectory ,Jacobsen[92], ROOM, (Realtime Object Orientated Methodolgy, Selic[92]).

The approach taken by each method is similar, however there are many differences in notation and apparent advantages and disadvantages with the various methods. For example, OMT is strong in analysis, BOOD is good for design, whilst Objectory is useful for behaviour analysis, (Quatrani[98]). The basic approach is to decompose the system into objects representing the structural aspects of the system under consideration. Object relationships are defined through the use of diagrams, which show data flow and messages passing between objects. System behaviour is encapsulated in the implementation of the objects, the user has no knowledge of how behaviour is effected, just the functionality provided and the types of inputs and outputs expected.

The problem with object orientated methods is the confusion generated by having so many methods available to evaluate, they are subtly similar, but different enough to have distinct advantages over each other in certain situations. It is only recently with the emergence of UML (Unified Modelling Language, detailed in Quatrani[98]), which brings together the best of Objectory(Jacobsen[92]), BOOD (Booch[91]), and OMT, (Rumbaugh[91]), that some form of standardisation has started to occur. UML was developed too late to be considered as a systems modelling technique for this project.

Song[94], whilst describing an approach to compare software design tools/methods, stated that

> "Trying to evaluate the different methods by examining user experiences is very difficult, this is because existing comparisons are often based upon the experiences of practitioners and the intuitive understanding of the authors. Consequently, they tend to be subjective and affected by the application domain."

The method developed by Song[94] is easy to understand, however a problem arises with the number of software design methods one would need to apply the comparison method too, and the amount of time required to learn the methods themselves. To carry out this task, with regard to comparing object orientated tools, is beyond the scope of this project.

3.4 Behavioural Model: Logical View

Behavioural models are a low level representation of system behaviour (Ward[85]). They relate to small, modular, sections of subsystem functionality. For the behaviour model, a number of tools will be examined, ranging from the very abstract, such as finite state machines, Z, and petri-nets, to the more common flow charts and Psuedo-Code. The techniques to be examined are a mixture of graphical and textual methods.

3.4.1 Finite state models

The basic theory of Finite State Machines (FSM) dates back to the 1950's, with original work by Mealy[55] and Moore[56] . FSM's are abstract representations of the state a process can be in. The representation notation is very simple, but the level of abstraction is very high. The basic notation of a Moore FSM consists of a finite set of states depicted as labelled circles, one of which is the initial state (marked with an arrow) and one or more terminal states (marked with "halos"). An alphabet of symbols is given, and a transition function, depicted by labelled arcs, showing how, given a current state and symbol, a new state is reached. Figure 3-8, below, taken from (Laplante[93]) shows the FSM for an even parity checker.

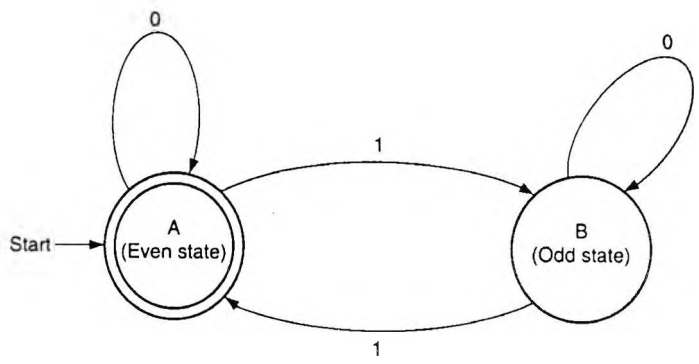


Figure 3-8 Finite State Machine representation of an Even Parity Checker

The state machine is in its 'even' state if there is an even number of ones in a byte. i.e.. the bit pattern 10110110 would generate a 1 as it is even parity. This process is used as a common error capturing procedure for serial bit communications.

Bucci[95], states that the classical models of FSM's are unsuitable to represent the functional and structural aspects of a system. This is because they cannot represent an infinite number of states and model time dependency. Both of these functions are required of a system to model the continuous behaviour of a mechatronic system. It is also stated that the classical FSM model is unsuitable when a large number of states are present. This is a difficulty which Buur[90] also found when using this method to model system behaviour.

To get around the problems of classical FSM systems, several extensions were made to the notation. This eventually evolved into the State Transition Diagrams (STD's) as used by De-Marco[78], Ward-Mellor[85] and Yourdon[89]. They can define inputs, outputs, cause of the change in state and the change in state. The approach taken is a lot less abstract than classical FSM's. An example of a simple STD for a lamp with three light settings (taken from Ward[85]), is shown in figure 3-9.

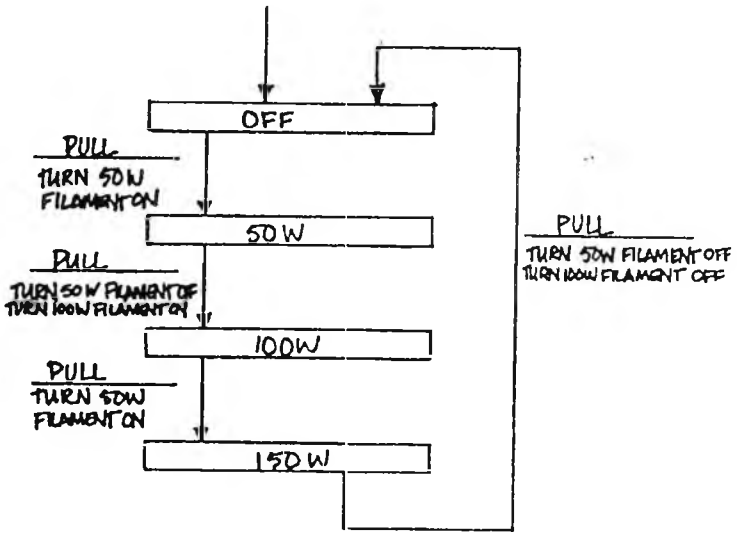


Figure 3-9 State Transition Diagram of a variable light lamp (Ward(85))

Each possible state of the light is given in the box, the action to change the state is written above the line and the result shown below the line. They are generally easier to understand than FSM, but complicated diagrams, can still be incomprehensible unless detailed knowledge of the functionality is given. This approach was employed in the

method advocated by Salminem[92] and Heilala[92], although they could not find much justification for the use of these models.

Another extension to FSM's are Harel statecharts (Harel[87],[88]). In this extension, it is possible to show integration between states and substates, and how they interact. They are more flexible than STD's and a computer tool exists called STATEMATE, (Harel[90]), from which C code can be automatically derived from the diagrams. An example of a Harel statechart is shown in figure 3-10. taken from Laplante[93].

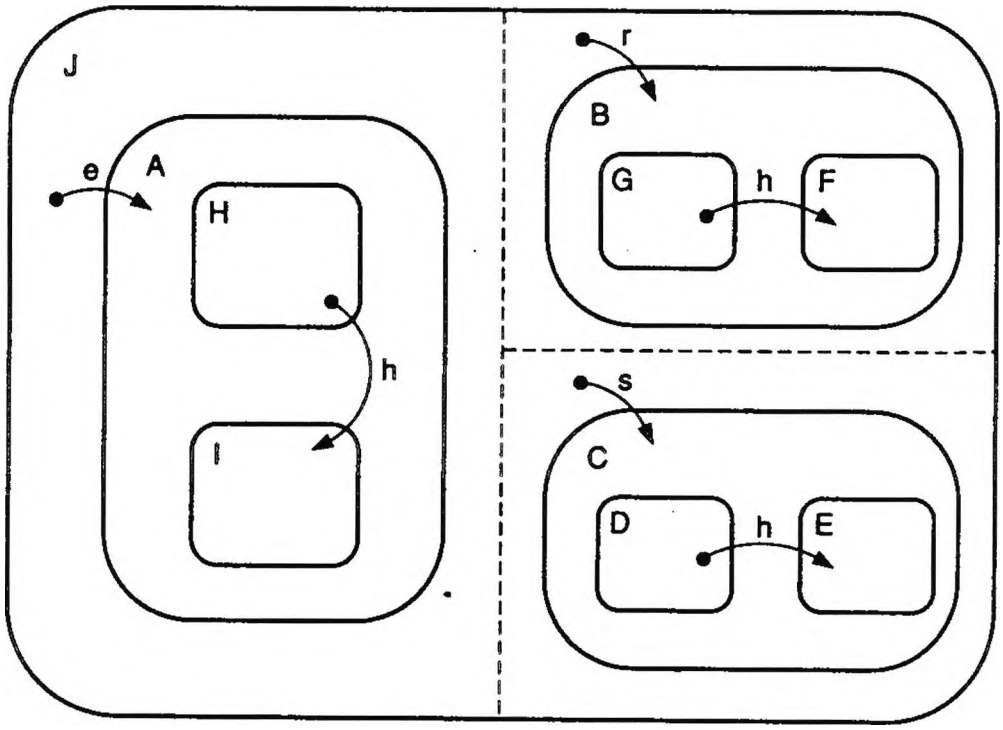


Figure 3-10 Example of a Harel State Chart(Laplante[93])

3.4.2 Petri nets

Petri-Nets were first introduced by Carl Petri in 1962, Petri[62]. His work examined the communication between asynchronous components of computer systems, and it was further developed by Holt[70], who described how Petri-Nets could be applied to the modelling and analysis of systems containing, concurrently operating components. This idea fits in well with mechatronics, because control events and component behaviour are normally occurring concurrently in a highly asynchronous manner. Petri-

nets and various extensions to petri-nets could be used to model and analyse these interactions.

Petri-nets, which consist of a series of circles called "Places", are used to represent data stores or processes. Lines are used to represent transitions or operations. Transitions are fired when a process is finished or a data store is filled, signified when a "token" is placed in the circle (a black dot), not shown in figure 3-11. The transition can only be fired when all the Places that contribute to the firing of that transition contain a token. An example, taken from Peterson[81], is shown in figure 3-11.

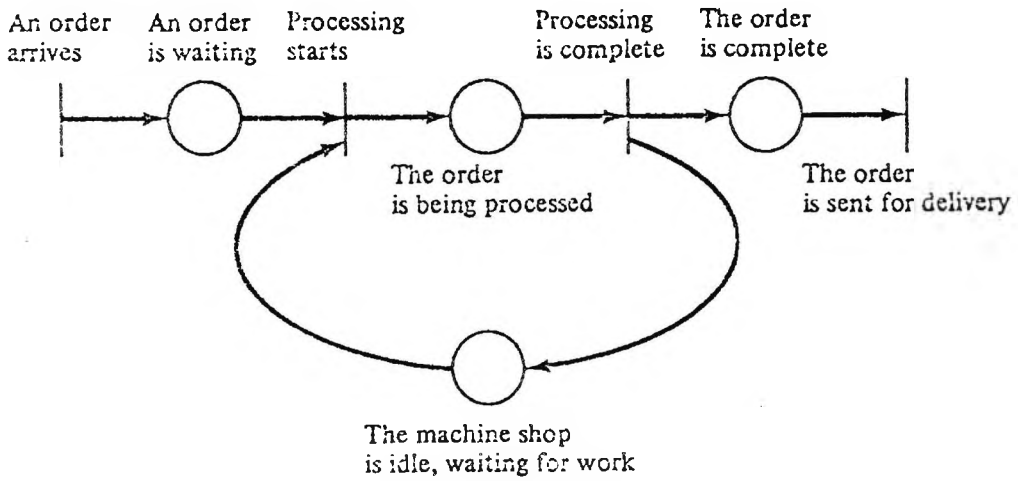


Figure 3-11 Petri-Net model of machine shoporder processing (Petersen[81])

Petri nets have an advantage over FSM's, which are a subset of Petri-nets Murata[89], because they show the preconditions required to fire a particular process. However, they do suffer from the fact that they are still extremely abstract and unless complex extensions employed suffer from limitations of expressiveness (Bucci[95]). Several diagrams are also necessary to fully represent each state of the system as tokens pass from one transition to another and the system state changes.

3.4.3 flow charts

Flowcharting is one of the simplest and earliest techniques developed for representing system behaviour. It shows the progression of behaviour along predefined paths, different behaviour enacted if certain conditions are met. Flowcharts are very easy to read, they support hierachial sub-functions and they can also model systems with infinite states by the use of comparator functions. However, they do not support

concurrency , timing details or interrupt functions very easily. If using flowcharts, there is a tendency to develop an implementation model as opposed to a behavioural model. Laplante[93], quoting Parnas, states that they should only be used for small programs (under 5000 lines). A simple flow chart is shown in figure 3-12.

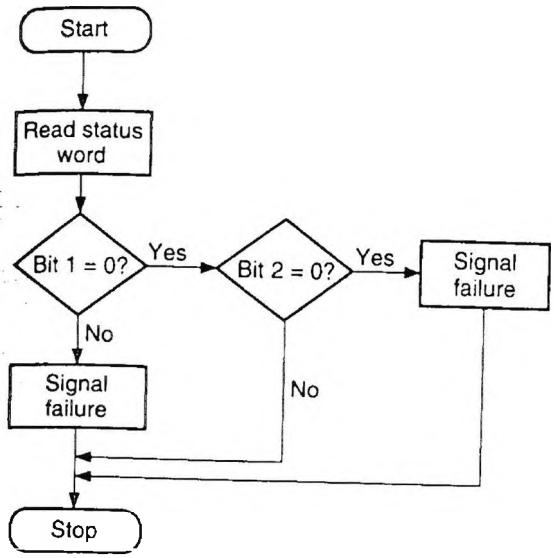


Figure 3-12 Example of a simple Flowchart (Laplante[93])

3.4.4 Textual Behavioural Models

All of the methods examined so far are graphically based, however, there is a series of tools which are primarily textually based, these cover Program Design Languages (PDL's), Psuedo Code and Formal Methods.

Program Design Languages differ slightly from Psuedo-Code. The latter is a an informal high-level description language for which a compiler does not exists. PDL's are a more formal, high-level language which can be translated directly into program code, either by use of a compiler or simple translation. Both methods are used to specify system requirements in a manner that is very close to actually writing code(Laplante[93]).

The chief advantages of Pseudo-Code and PDL's are that they are more abstract than writing specifications in English, as they are closer to a programming language. Both methods can deal with concurrency, and formal program-proving techniques more

easily than unstructured methods, such as, pictures or English, see figure 3-13 .

```
begin
        Accept the card
        Display "enter your code"
        If code not record
                begin
                        Display " Sorry"
                        Exit/Eject card
                end
        else
                begin
                        etc.
                end
end
```

Figure 3-13 Example of Pseudo Code (from Laplante[93])

The main disadvantage is that even though PDL's/Psuedo Code are both forms of simplified programming language, they still have a set of semantics in which the user must be fluent. It is also possible that because the specification is written in a form very similar to the final code there is a possibility that the behavioural model will define the implementation model.

Formal methods, techniques such as the Calculus of Communicating Systems (CCS) ,Milner[80], Communicating Sequential Processes (CSP), Hoare[78],[85], Z, Abrial[82], Temporal Logic, Ostroff[89], are just a few of the methods used to mathematically specify user requirements for real time systems. Petri-Nets are also considered to be a formal method, this because they are supported mathematically by a variation on set theory called, bag theory (Peterson[81]) an explanation of which is beyond the scope of this thesis.

The approach taken by each method is different, the main commonality is the use of a strict, very rich, mathematical symbology which is used to represent the system requirements (see figure 3-14 below for an example of Z). This symbology is used to

generate a "Formal Specification" against which the software can be mathematically proved. They can be used for high or low level program descriptions.

AddvalidPair==

{Vocab, Vocab':set of (Native, Foreign);
n?:Native;f?:Foreign!: Message |
Vocab ∈ WellFormedVocabs
∧ n? ∈ OrthoNative ∧ f? ∈ OthoForeign
∧ Vocab' = Vocab ∪ {(n?,f?)}
∧ rep!= Ok}

Key	
∈	Membership
==	Equals
=	Equivalence
∪	Set Union
\	Set Union
∧	And
:	Subset definition
?	Mapping variable

Figure 3-14 Section of Z code defining an operation (Potter[91])

The main advantages of these methods are that the program requirements are rigidly defined and can be verified using the supporting mathematical tools. The disadvantages are the complexity of the mathematics required to represent and verify the model and the very high level of program abstraction which these methods produce.

3.5 Behavioural Modelling: Energy based view

The majority of mechatronic systems employ some degree of energy flow between system elements. This energy flow needs to be analysed to determine which components are capable of allowing the system to match its specification and to examine how the system behaves under different conditions in the time domain. This analysis can either be very simple, as is the case of selecting a gear box to give a certain output speed or very complicated when, for instance, the acceleration of the output is very important along with the output velocity and correct torque characteristics.

Two techniques are commonly used for this form of simulation based analysis. A traditional approach based upon modern control theory (Ogata[70]), and the Bond Graph approach (Paynter[61], Karnopp[75]).

3.5.1 Control Approach

The control approach to systems modelling is based upon breaking the system down into its various subsystems and components. These are then joined together in a block diagram, using connecting links between the blocks to indicate energy flows. Each block within the diagram contains the differential equations, in the Laplace domain, which describe that particular components behaviour. Figures 3-15 and 3-16 show the schematic and associated block diagram representation for a motor speed control system.

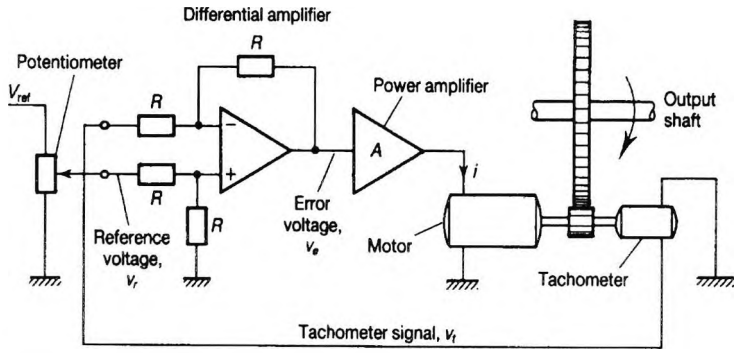


Figure 3-15 Schematic of a motor speed controller (Golten[92])

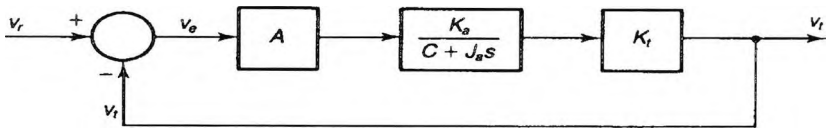


Figure 3-16 Block diagram in the S domain for the motor speed controller (Golten[92])

The closed loop transfer function is derived in the Laplace domain and an input signal applied. Partial fraction expansion is then carried out and the response equation transformed back to the time domain to determine the system response in the time domain.

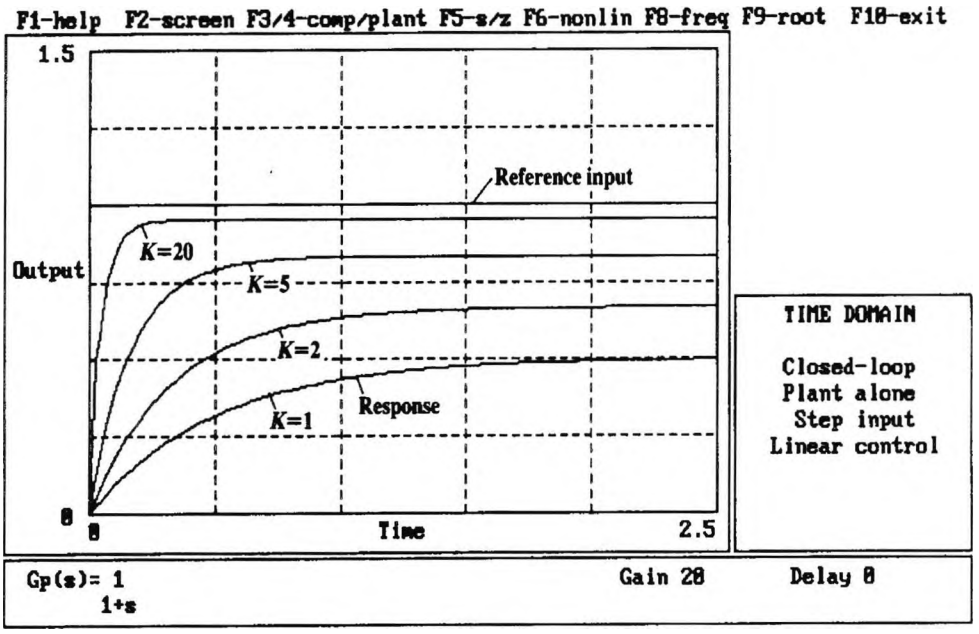


Figure 3-17 Time response output of a 1st order control system to a step input

Many programs exist, such as CODAS(Golten[91]), VISSIM, and SIMULINK, which allow system modelling in the Laplace domain and give the system response in the time domain. This is achieved without the tedious work of carrying out partial fraction expansion, and transformation of the resultant formula back into the time domain to get the system response. Figure 3-17, shows the time response output, produced by CODAS, of a typical 1st order control system, to a step input and varying values of gain (K).

Jebb[90] proposed the use of block diagrams for the simulation of a motor control system, the objective being, to link the simulation with Robust Engineering Design to evaluate the effect of system parameter noise within DFD. Viitanen[92], whilst carrying out a survey of simulation methods for mechatronics, mentions block diagrams as a possible approach to motor and electronics simulation. Farsi[94], employed the block diagram technique to model a PID controller for a window cleaning robot.

3.5.2 Bond Graph Approach

Bond graphs were originally introduced by Paynter[61], with further work carried out by Karnopp[75], Rosenberg[87]. These are based on the development of a series of state equations derived from translating a system diagram into a number of power bonds, which show generalised directions of effort and flow. A common set of elements are used to model a variety of energy domains, a feature which is attractive to mechatronics modelling. A simple example is now given, this is based upon the application of the method to an air pump, (taken from Atherton[98]), a schematic of which is shown in figure 3-18.

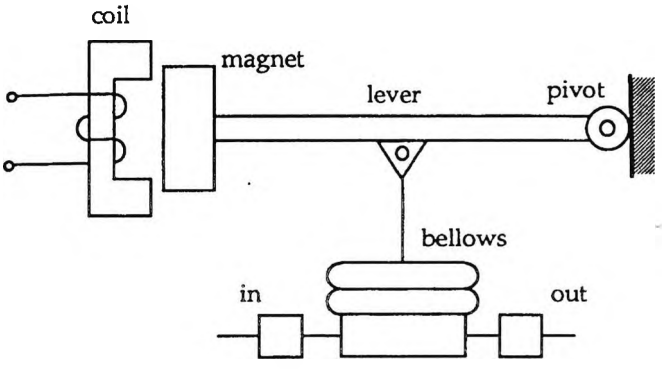


Figure 3-18 Schematic of an Air Pump (Atherton[98])

The first task is to follow the effort and flows through the system, component by component. For this the components are translated into system elements such as Transformers, Gyrators and Flow or Effort sources, each of these having a predefined relationship with effort and flow. Power bonds (see figure 3-19) are used to transmit the efforts and flows between elements. Signal Bonds are used to define control actions to the elements (not shown in figures 3-19 and 3-20) , and are considered to have a very low power such that they are considered not to affect the power of any system element.

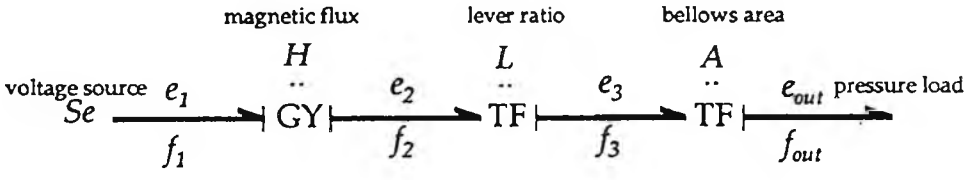


Figure 3-19 Bond Graph model of an Air Pump (Atherton[98])

The magnet, shown in figure 3-18, is represented as a Gyrator (power conservation device), the lever and bellows as Transformers (effort/flow magnifier) and Input and Output relationships defined. These relationships are shown in figure 3-19.

The system can be made more complex by incorporating losses into the system such as, inductance, electrical resistance, capacitance, compliance, for the elements in the system which they affect. This gives the following bond graph model, (figure 3-20).

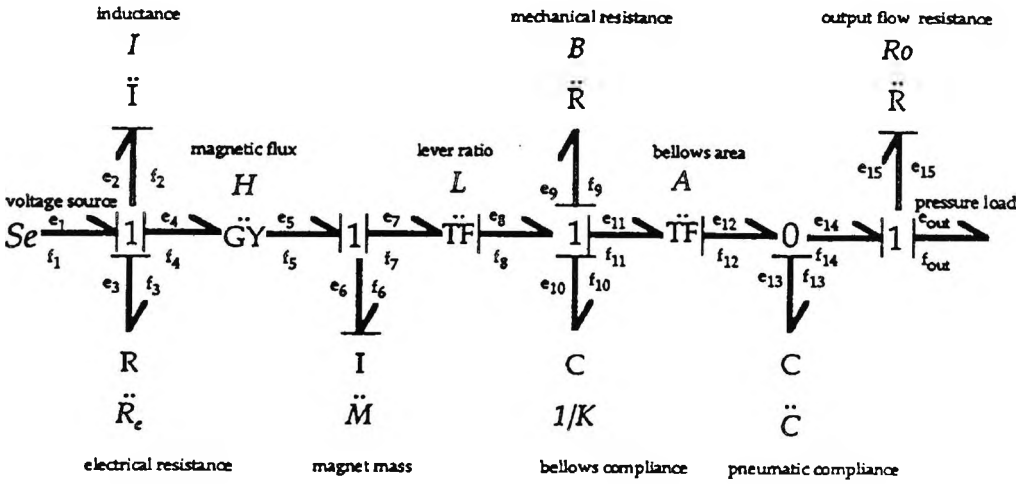


Figure 3-20 Bond Graph model of an Air Pump, incorporating system losses(Atherton[98])

Programs exist (20-Sim, University of Twente, Holland), which allow the designer to generate the model with the mathematical model generated automatically, Atherton[98]. Ulrich[89] applied a bond graph approach to describe the systems in a mechanical design, giving examples of their application to a number of electro-mechanical systems. Gawthorp[92], has given a detailed example of the application of Bond Graphs to modelling a robot controller system. The process is based around the derivation of differential equations from the bond graph, these equations are then represented as linearised state equations, which are solved using MATLAB software. Bradley[91],[93a], has proposed the use of bond graphs for the representation of mechatronic systems, whilst no examples are given, this indicates how Bond Graphs could be applied to a computer aided mechatronics design package. Yan[97] develops a bond graph model of a DC motor, the effort and flow bonds are separated into two signal lines and converted into a conventional block diagram, this is turned into a computer model using, a dynamic, graphical, modelling tool.

3.6 Summary of modelling techniques

From the variety of tools and methods reviewed, it is clear that no single method is applicable to the entire problem area. This was also concluded by Shakeri[97] when looking for methods to model mechatronic system behaviour and Zhang[96], whilst modelling a mobile robot for trench excavation (LUCIE, Bradley[92]). Further problems are encountered due to the positive way in which nearly all the methods were presented in the literature, thus making it much harder to differentiate between them. Song[94], suggested that this is due to the biased view of the various authors in promoting their favourite tools. Yourdon[89], advocates an approach to systems modelling suggesting that:-

> "You should use any model which works for you and the situation you are in. Different users require different modelling tools either because of past experience or they find certain tools confusing and intimidating."

He continues in this vein, proposing that good modelling tools have very simple notation, with very few rules, symbols and new words for the user to learn.

Bearing this advice in mind, and the need for a method which supports hierarchical functional decomposition , (to fit with the design tools selected) and this, authors previous experience it was decide to use a form of ACP diagram for the systems design tool. The structured analysis techniques of Ward[85] and Yourdon[89] are not far removed from ACP diagrams as employed in MASCOT 3, the only real differences being in the symbology and notation. Object Orientated methods were not selected due to the sheer variety of methods available and the lack of clear evidence as to which to pick, although the general ideas of Object Orientation seem to fit in with ACP diagrams. Jackson Structured Programming was not felt suitable for the general systems model because it cannot be combined with the functional decomposition approach taken with the design generation.

To model the behaviour of the hardware logic and software at sub-system level it was decide to employ pseudo-code. Pseudo code provides a behavioural and implementation model which can be used to match the final system to the original requirements, without having to carry out any complex mathematical proofs. Formal methods, petri-nets, and FSM are too abstract and, although expressive enough to deal with the complex decision structure required of the logical model, require very different ways of

thinking, to employ. State Transition Diagrams and Harel Statecharts also have their place and can be very useful, but can be incomprehensible if they get too large. Flowcharts can be used for modelling small subsystems, but they cannot deal with concurrency.

The methods researched as candidates for the hardware behavioural model or, energy modelling tool, were Bond Graphs, as mentioned above, or a conventional block diagram approach (Ogata[70]). The two advantages that the Bond Graph approach offers is that, firstly the effort and flow variables are kept together because they employ, multi-input, multi-output state equations to describe the system behaviour, which can then be solved simultaneously. Secondly, there are topographical in layout, mirroring the topology of the physical system. The main disadvantages are that computerised modelling tools for bond graphs are not expressive enough when dealing with the logical view of the system and they lack the ability to model components, such as a motor, in any great detail. The block diagram approach keeps the topographical layout of the system intact and it can model component behaviour in a detailed manner. However, it does requires a greater understanding of the signals generated from a system and how they interact with each other, but overall the result is the same. This observation was made by Shakeri[97], who developed block diagrams from bond graphs. The block diagram approach was eventually selected because:-

1) The availability of suitable software modelling tools.
2) The ability of block diagrams to match the functional decomposition approach of ACP diagrams.
3) The majority of computerised tools, which follow a block diagram approach, allow for modelling of the logical model.
4) Allows partitioning of the failure behaviour from the perfect behaviour of system elements, when carrying out failure modelling.

However, as mentioned above, although the number and types of modelling methods associated with an OO approach are very confusing, the general ideas of OO modelling can easily be applied to ACP.

In conclusion it was decided to use a generalised OO approach, linked to ACP diagrams to represent the systems model. For the logical model, pseudo code was selected because it is very easy to read and compare system behaviour with the original

specification. A block diagram approach was eventually selected to represent the energy-based model due to the support offered by readily available computer simulation languages.

3.7 Review of methods for computerisation

A number of software tools were examined for the computerisation of the design tools proposed for the system model. Here, the requirement was for the capability of representing a hierarchical systems model, based on ACP diagrams. At the lowest level of system hierarchy, the tool selected should support a logical model that:-

- May not necessarily be textually based, but should allow comparison with a textual model.
- Support differential equations and logic which enable the modelling of systems dynamics and discrete state behaviour.

The tools that were reviewed covered, expert system shells, OO programming, graphical simulation tools and the Scheme Builder project, which is based on bond graphs.

3.7.1 Object orientated methods

Object Orientated methods were included because they are a powerful software development paradigm supported by common software languages, such as Smalltalk or C++. Despite this support, a number of problems exist with using an OO approach.

1) Detailed knowledge and experience of the software language to be used is required to model an OO system effectively, a process which could take a great deal of time.
2) Dynamic modelling of the system is complicated by the lack of support for pre-existing objects for modelling electromechanical system elements, such as, motors, screw drives, and gearboxes.
3) The system model requires the behaviour of all the system elements to be modelled concurrently, this form of OO programming being a skill in itself.

3.7.2 Expert System shells

A number of expert systems shells (Crystal, VP expert, Kappa, Knowledge shop) were reviewed to determine whether they could be used to create the system model. They are

all based upon production rule systems, which describe the system behaviour as a series of IF THEN statements. An important feature of experts systems is the ability to automatically determine failure probabilities based upon certainty factors that a particular rule will fire. This can be entered as the reliability of the system element that the rule is representing, this was the main reason that expert systems were examined. The systems were tested by trying to simulate a simple Emergency Stop, latched relay circuit, see figure 3-21. The circuit gives a latched output when the power is on, and the momentary on/off switch is pressed. The power is turned of when the momentary emergency stop switch is pressed or power is turned of externally to the system boundary.

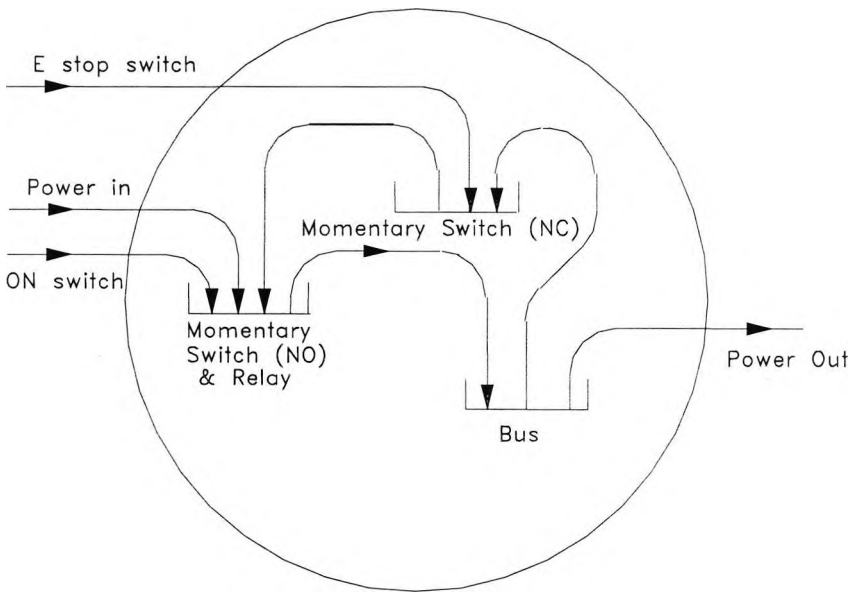


Figure 3-21 ACP model of On/ Emergency stop latched relay circuit

It was not possible to use the VP expert or Crystal software to model this simple circuit, as they could not model the feedback element of the latched relay. Kappa software proved to be extremely complicated to operate, despite being windows based. The Knowledge shop software was the most useful, it allowed the relay system to be modelled and created as C code. However, to make the system behave properly and generate accurate values for systems reliability required many alterations to be made to the underlying C-Code. The process proved to be fairly long winded and limited by the expressiveness of Knowledge Shop, by which it is not possible to model dynamic equations.

3.7.3 Graphical modeling tools

Two main graphical modelling tools were examined to model the system behaviour, VISSIM (from Visual Solutions Inc) and SIMULINK (Mathworks). Both operate on a hierarchical block diagram approach and contain toolboxes of common electrical and mechanical elements. Each element of the system can be represented as a module linked by energy, signal or physical links. They support Laplace transforms and can give time domain output in either a graphical form or as a file.

When using VISSIM, a model with identical behaviour to the Knowledge shop model (See figure 3-22) was rapidly produced, the model being created in a matter of minutes. This is because VISSIM is very intuitive and more readily understandable than Knowledge Shop for this form of behavioural modelling. Behaviour is represented as a series of logic gates and differential equations, that are activated concurrently for a specific time step..

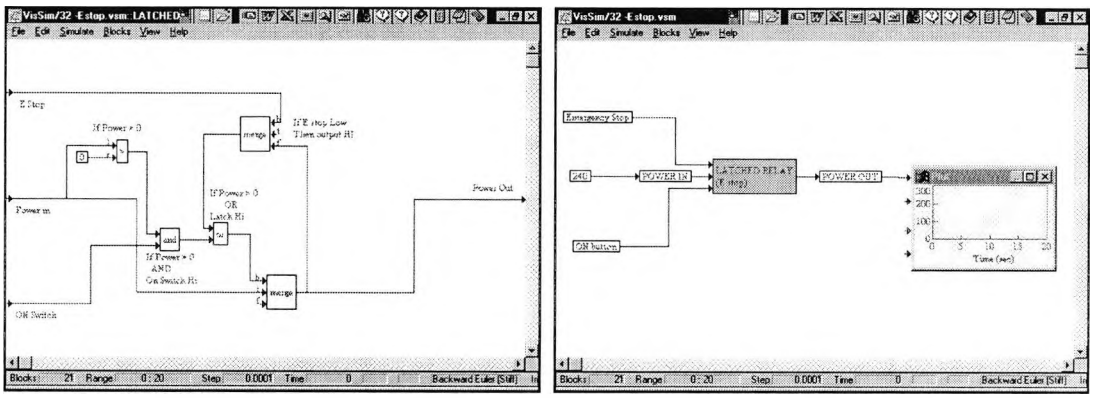


Figure 3-22 VISSIM models of the ACP diagram in figure 3-21

As mentioned previously, the functionality of SIMULINK is almost identical to VISSIM, the main difference is the user interface. SIMULINK employs a graphical user interface which tends to clutter the screen with toolboxes, whilst VISSIM provides a menu driven system, which leaves the computer screen uncluttered. It was also easier to link the system elements in VISSIM compared with SIMULINK. This author feels that VISSIM provides a better working environment than SIMULINK, a useful advantage considering the amount of time that would be spent using the package to be selected. Both software packages are commercially available.

3.7.4 Schemebuilder

The Scheme Builder project(French[90], Bracewell[92], Bradley[93a], Bracewell[95], Da Silva[97]) has to date made the greatest advancement in developing an all round design and modelling tool for product design. The original project specification is given in French[90], who reports that the project was originally envisaged to be a concept generation and evaluation tool. Bracewell[92] records the development of the project into a computerised design and simulation tool as does Bradley[93a], for mechatronic products. The detailed working and abilities of Schemebuilder are described in Bracewell[95]. The example used is the application of Schemebuilder to the design of a metered hypodermic syringe. Mechatronic elements such as a microprocessor are shown in the overall context diagrams, but no information is given of how the microprocessor behaviour is represented in the model. Bond graphs are used to model the dynamic behaviour of the system. Timed Petri-Nets were proposed by Bradley[96] to take into account the logical model and timing considerations required by a mechatronics model, but this was not implemented. Since then, the focus of Schemebuilder has turned to the modelling of hydraulic systems and the use of knowledge based systems (Da Silva[97) for the selection of Hydraulic components.

3.8 Summary for computer representation methods

The Schemebuilder project despite being very sophisticated is not suitable as a modelling medium, because, firstly it does not support the logical model, and secondly it is still a researched based tool which is unavailable to the author. Rule based expert systems could be used for system modelling as they support the logical model, but problems can be envisaged in generating the behavioural model. They also proved far more complicated to implement than computerised graphical models.

Computerised graphical models provide the best approach to modelling the entire system. They allow hierarchical functionality to be maintained along the lines of ACP diagrams. Behavioural and logical models can be implemented relatively easily and are concurrent in nature. Of the two choices available, SIMULINK and VISSIM, the author decided that the easiest to use is VISSIM. It provides the best graphical user interface and has the same functionality as the SIMULINK software.

Whilst the implementation of object orientated software using languages such as C++ for the model representation is laborious to implement, the general ideas embodied by

OO techniques can still be applied to a function-based model generated in VISSIM. This is achieved by developing a model for a system element, the behaviour of this is encapsulated in a compound block, with messages passed to the system element via the energy and signal links, to produce element behaviour. The true power of applying OO techniques to the model come when testing combinations of components with each other. If the system element is thought of as an object, it allows different values of constants to be passed to the object. For example, a system element such as a motor could be represented by its Laplace Transform encapsulated in a compound block, with inputs such as load and voltage applied to the block. Here, the outputs generated would relate to torque and angular velocity. By inputting different values of the motor constants i.e. Kt (torque constant), Ra (Resistance) and I (inductance) several different motors can be simulated and evaluated (Bleakley[97]). The object can also be reused by saving the compound block encapsulating the system element, as a new diagram.

The method and tools used for the computer system representation will be based on a simplified Object Orientated approach applied to VISSIM.

This chapter selected appropriate tools for the development of paper-based system models, various methods were then examined which enabled these tools to be computerised.

The next chapter reviews the various method available to carry out a systems fault and risk analysis, this review is extended to cover computerised tools for the fault modelling of systems. A method is then proposed, based upon ACP diagrams, and VISSIM models, for the computerised fault analysis of mechatronic systems.

4 Failure analysis: techniques and computer tools

4.1 Introduction

The main objective of this chapter is to investigate methods and tools used for safety and reliability analysis which could be used for the analysis of mechatronic systems. Based on the review, methods will be selected which could be applied to the other tools selected to date. This research is necessary because, although standards exist which give guidelines for the safe design of machinery, such as, BS EN 292 part 1 (BSI[91a]) and BS EN 954-1: (BSI[92]), little information about how to apply these guidelines is given, apart from a complex design flow diagram, see figure 4-1(from BS EN 292 part 1).

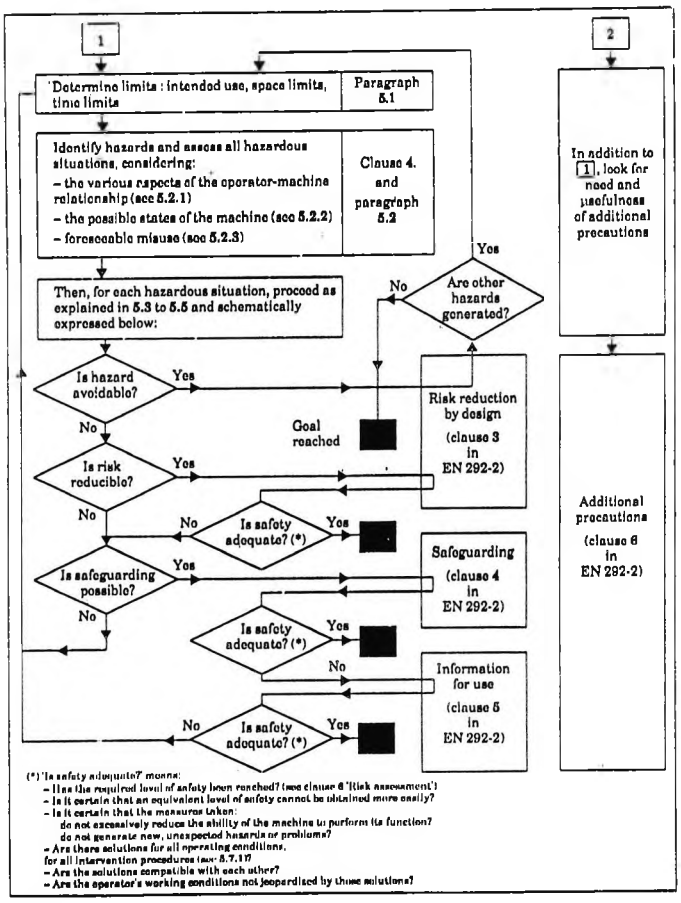


Figure 4-1 Design for safety according to BSI EN 292 part 1

Information relating to tools for failure and reliability analysis, such as Failure Modes and Effects Analysis(FMEA) and Fault Tree Analysis(FTA), (BSI[91b] & BSI[91c]).

The EC Machineries Directive (EC[93]) is slightly more understandable, giving guidelines, but with little or no information about how to implement them.

It is only in the guidelines relating to Programmable Electronic Systems (PES's) in safety related applications (HSE[87] & HSE[93]) that any detailed guidelines are given with examples. In this case, the design of a PES for a chemical plant is given. The method involves the use of a hazard analysis techniques and FTA. The final system uses a combination of redundant sensors and majority voting to determine system failure and implement plant shutdown. The basic design process is given in figure 4-2. Little or no information is given in relation to mechatronic systems. Wang[97] notes that there is very little information in the literature generally, relating to specific application of techniques, such as FMEA and FTA, to the design process and how they interrelate with it. The work of Edney[96] and Sontow[93] are the exception rather than the rule, with respect to this issue.

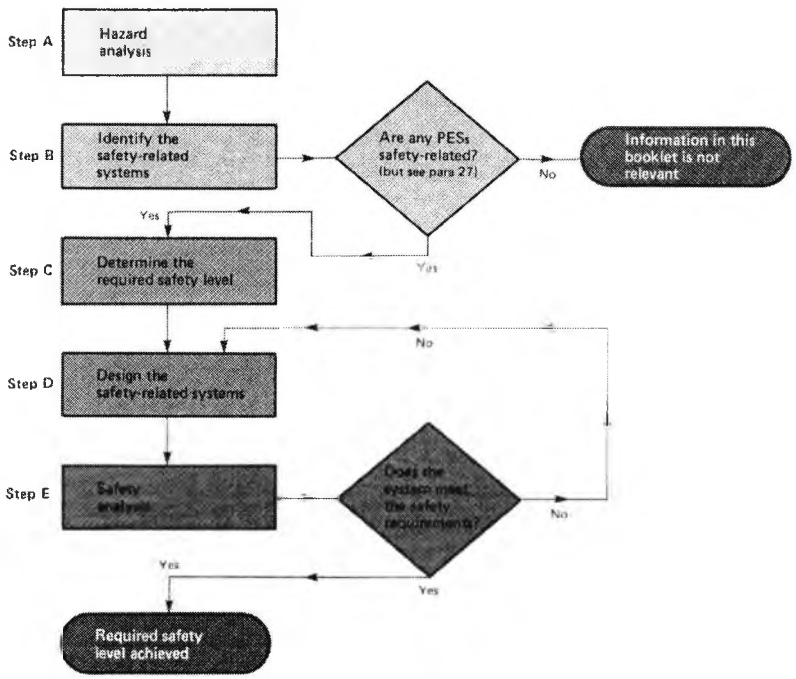


Figure 4-2 Design for safety according to The guide for PES's (HSE[93])

The chapter is divided into two sections, section 4.3 dealing with qualitative tools such as FMEA and FTA, whilst section 4.4 examines a number of computer based tools which have been used for failure modelling and analysis.

4.2 Introduction to Failure Analysis tools

Three areas of tools seem to exist in the literature relating to failure analysis tools. These are qualitative, quantitative (Wang[97] and computer based modelling tools, which integrate qualitative and quantitative tools to some extent..

Qualitative tools depend upon the ability of the design team to work systematically through various fault scenarios to determine the effects of failure propagation through a system, or work from a failure to determine how the failure was produced.

Quantitative tools are used to determine the system characteristics, such as failure rates, repair rates, reliability and mission times. They generally involve the development of a complex mathematical model which is analysed to determine the system characteristics(Wang[97]). These models are purely mathematical and are based upon standard calculation methods. Because this study is interested in designing out the failure modes, as opposed to analysing failure probabilities, for example, this area will not be discussed further.

Three types of computer based tool exist for failure analysis of complex systems. These are based upon, expert systems, computerised mathematical models employing state behaviour, and dynamic simulation. These tools are reviewed in section 4.4.

4.3 Qualitative Failure Analysis

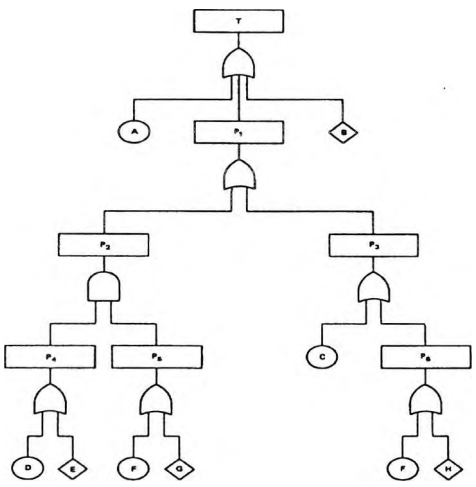
Qualitative methods are based upon describing a hazard in terms of its severity to the system or personnel (Wang[97]). The terms used range from catastrophic to negligible. The probability of hazard occurrence is also described, and in this instance the terms range from frequent to remote. Based upon the information produced by the analysis, measures can be taken to eliminate or control hazards. Qualitative methods include Fault Tree Analysis, Failure Modes Effects Criticality Analysis, Event Tree Analysis and Hazard and Operability studies.

Problems associated with qualitative methods are that their application is tedious, designers and analysts have a poor understanding of the tools and that knowledge tends to be restricted to small groups of experts, (Russomanno[92]). However, they are the most common failure analysis tools employed.

4.3.1 Fault Tree Analysis (FTA)

Fault Tree Analysis (BSI[91b]) is a well documented, commonly used, qualitative, top down, safety analysis tool. The process starts by identifying a top level fault and then determining the conditions that lead to this in a deductive manner. The conditions that lead to each fault and deduced sub fault are analysed hierarchically and represented in a tree structure. Faults which lead to a further fault are joined together by AND and OR logic gates. An AND gate is used when two or more sub-faults must be present to produce a fault and OR gates are used when either of two or more sub-faults are required to produce a fault.

The system can be analysed further, by determining minimum cut sets. These are the minimum series of events which lead to a system failure. The worst case scenario is a system which has a minimum cut set which consists of a large number of single-point failures. A single point failure is a failure which occurs in one component and causes the complete system to fail. When analysis of complex systems is being carried out, the fault tree can become very large due to the various combinations of failure modes. When this situation occurs it is necessary to reduce the fault tree to a Reduced Fault Tree (RFT) (Klaassen[89]). This enables the minimum cut sets to be determined a lot easier. Figure 4-3a shows a typical Fault Tree, and figure 4-3b shows its corresponding reduced fault tree.



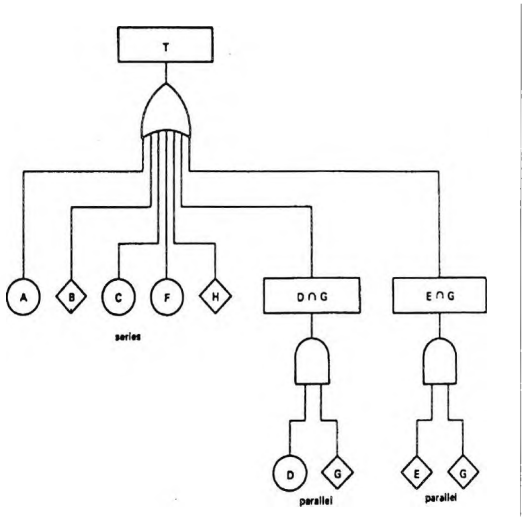


Figure 4-3a Typical Fault Tree **Figure 4-3b RFT of figure 4-3a**

FTA is a an ideal tool for the reliability analysis of complex systems (Seward[94]), it allows the designer to systematically analyse the events which lead to failure. It is

possible to associate reliability data with the failure modes and determine the overall system reliability (Sontow[93]). Two problems are associated with FTA. Firstly, it can only be applied in a reactive fashion as the design or general schematic needs to be in place to determine system or component failures. Secondly, there is no established method for identifying the major system failures (Seward[94]).

4.3.2 Failure Modes Effects Criticality Analysis (FMECA)

Failure Modes Effects Criticality Analysis or Failure Modes Effects Analysis,(FMEA, BSI[91a]) is a bottom up approach which examines each component and determines how that component can fail, thus leading to a failure mode. This process is carried out for each component or sub-system within the system. Once complete, each failure mode is analysed and evaluated according to the following criteria.

1) The likelihood of failure, 1= low, 10 high
2) The seriousness of failure, 1= not serious, 10 very serious
3) The detectability of failure and the ability of the system to take corrective action, 1= detectable, 10= not detectable.

The score for each of these criteria are multiplied together to give a criticality rating for a failure mode. A high score denotes a high risk failure mode and identifies an area of improvement. A low score denotes an area of low risk which could be improved but it is not essential. A section for an FMEA chart taken from Chamberlain[95], is shown in table 4-1.

FMEA is very useful for identifying areas of high risk. It is systematic and requires an in-depth understanding of the system and the components within it. It is not a proactive tool because, like FTA, it requires that the system design is nearly complete before the FMEA process can be carried out.

4.3.3 Event Tree Analysis (ETA)

ETA is another bottom up tool (Wang[97]), it is employed to examine the propagation of failure through a system in a systematic manner. It achieves this by theoretically setting an event, the effect of this event is then examined on the surrounding systems and subsystems. The propagation of these events are shown in a tree structure. This is shown in figure 4-4,(from Bentley[99]), which depicts an event tree associated with the

Fault	Rpn	Probability	Delectability	Severity	Solution
Communications failures	16	Low (1-2) Mechanical Damage	High (1) Lights on operator display disappear	Med-high (8) Emergency stop ,Auto or mechanical Rewind	Restrict access
Power supply failures	20	Low (1-2)	High (1) Self test diagnostics and watch dog software	High (10) Mechanical rewind in worst case	Independent psu's and battery backups where required
Faults in operator display	15	Lowish (3)	High (1) Self test on system start up	Medium (5) System will not function at all	Supply spare bulbs
Faults in dose rate meter	10	Very low (1)	High (1) Cross reference Encoders and hall effect sensors	High (10) Possible undetected exposure	Initiate contingency measures for source recovery
Source position encoder faults	25	Lowish (2-3) Spurious noise + Component failure	High (1) Cross reference Dose rate meter and hall effect sensors	High (10) Possible undetected exposure	Initiate contingency measures for source recovery
Faults in hall effect sensors	25	Lowish (2-3) spurious noise + Component failure	High (1) Cross reference Encoders and dose rate meter	High (10) Possible undetected exposure	Initiate contingency measures for source recovery

Table 4-1 Example of FMEA chart for the safe design of an remote Gamma radiography Control System

escape of an odourless poisonous gas in a chemical plant. In this example, the probability of the initiating event and all subsequent events are shown.

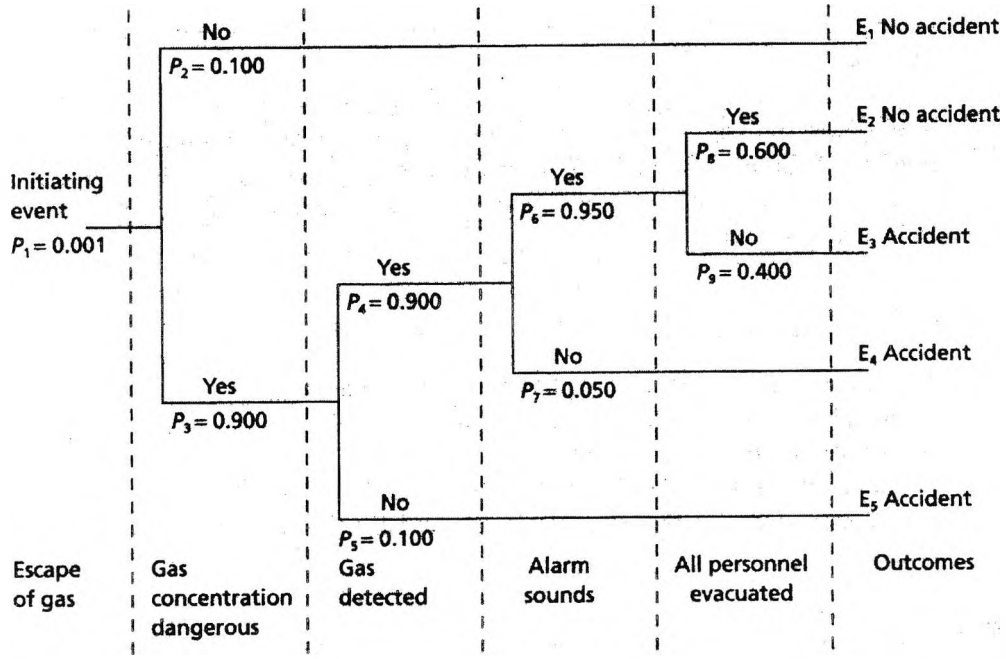


Figure 4-4 Event Tree Analysis for gas escape hazard (Bentley[99])

Event Tree Analysis provides a systematic and logical approach to identifying consequences and assessing the probability of occurrence of each possible resulting sequence caused by the initiating failure event.

4.3.4 HAZard and Operability Analysis (HAZOP)

HAZOP(CIA[77]) is a commonly used technique for the identification of hazards and operability problems associated with a process or system. It was originally developed for the chemical industry to identify the effects that changes in system parameters have on the system behaviour. It is usually carried out by a team of specialists who systematically question each aspect of every part of a system and its operation. This is achieved by applying a set of guide words, such as NO or NOT, MORE or LESS, AS WELL AS, too each component or parameter in the system or process. The effects of these deviations on the system are analysed to theoretically determine what effects may cause a hazardous situation. Once the deviations which produce the system failure are identified, the cause of a components departure from its normal state are identified and either eliminated or reduced.

Because of the nature of the process, the team of experts should consist of the system designers, engineers, managers and safety personnel. They should have access to , or detailed knowledge of , the system design and process operation (Adamantopoulus[96]) in order to carry out HAZOP successfully. Seward[94] notes that HAZOP is limited in its basic form in that it is more appropriate for use with existing technology. Because of the large number of interactions between different component deviations, it is also possible to make errors in the failure analysis.

4.3.5 Consequence Led Analysis of Safety Hazards (CLASH)

CLASH is a methodology proposed by Seward[94] in his paper "Hazard analysis techniques for mobile construction robots". It combines a number of the methods mentioned above, which have been adapted towards the design of robotic machinery.
The methodology is based upon developing a number of key words, such as, Crushing, Shearing, Trapping, Electrocution, and Arcing, which relate to mechanical, electrical or any other form of hazard. HAZOP is employed to identify the risks, using the above guide words. By examining the risks and using the FTA approach, it is possible to identify basic causes of the hazards. Once the failure causes have been identified, negation of them can be fed directly into the design process as customer requirements. FMECA is employed after several iterations of the above methods, to determine the criticality of the remaining failure modes. Further design may then be applied so that the probability of system failure becomes, As Low As Reasonably Possible (ALARP).

4.4 Computer based failure analysis tools

Due to the complexity of the interactions of components and subsystems, failure analysis of modern systems can be complicated and very involved. This makes qualitative failure analysis one of the most expensive tasks carried out by reliability engineers (Russomanno[92]). One of the ways in which the accuracy of the analysis could be improved, and cost reduced, is by applying computers to the problem. Other reasons for using computers, stated in Shaw[90], who examined the use of FMEA in the UK motor industry, are:-

1) Engineers have insufficient time for proper analysis
2) Analysts have a poor understanding of FMEA and designers are inadequately trained in FMEA.
3) FMEA knowledge tends to be restricted to small groups of specialists.

Wang[97], in his survey of safety analysis techniques, also calls for the further development of computer simulation techniques to be integrated with the standard failure analysis tools.

A number of failure tools, which employ some form of computer based model, have been produced. These can be split into three main areas, expert system based tools, computerised mathematical models, based on state behaviour, and dynamic simulation models. These tools are discussed in sections 4.4.1 to 4.4.3.

4.4.1 ES failure analysis models

One of the main reasons why expert systems were considered as a simulation technique was because a number of failure analysis tools detailed in the literature are expert system based. However, it must be noted that they are few and mostly very specific in their application area.

One of the earliest applications of expert systems to failure analysis was Boeings FMEA assistant (Kamhieh[89). The structure of the system to be examined is modelled as a semantic network, which links the system components, whose behaviour is represented in frames (simple objects which define the components behaviour and failure modes). The engineer can input new components into the system, using an intelligent editor which contains failure facts already derived from the component main class, this information can be added too by the engineer, if required. The engineer can investigate the effects of various failure modes and the failure propagation through the system using the system model. Failure mode selection and analysis of the results is carried out manually.

Failure Analysis and Diagnostic Expert System, (FADES) (Wood[90]), is a more sophisticated modelling tool used for reliability analysis of hydraulic and fluid systems. It is based upon a system model in which component behaviour is modelled in a qualitative coarse grained manner , e.g. flow rates are given a designation of high, low or medium instead of quantitative values. This decreases the analysis time and reflects

the inability of most expert systems to carry out complex differential calculations. The component behaviour is defined in production rules of the form , IF X is high AND Y is low THEN output is low. FADES (Wood[90]) automatically searches all failure modes and combinations of failure modes, from which it derives an extensive FTA that then has to be reduced manually to determine the minimum cut sets.

The Integrated Fault Management Environment (IFME), (Murdoch [92]) is an expert system based modelling environment in which component behaviour is represented as a logic based model. The component behaviour is entered as a simple procedural language or as a truth table. One component communicates with another via specific connections which exist between ports on two components. FMEA information is contained within a series of production rules which allow the engineer to examine different failure effects. The IFME was applied to the reliability analysis of an attitude and orbit control subsystem for satellite. It was found to be very useful in the design and development of the control system, enabling automated failure simulation to be carried out, the results of which were analysed by hand. Palumbo[94] takes a very similar approach, except in this instance the target system is aircraft hydraulics

Sneak Circuit Analysis (SCA) and FMEA were combined together by Savakoor [93]. SCA is a heuristic based approach for examining the latent conditions of electronic circuits, which may lead to unplanned or unexpected modes of operation. These rules are combined with rules for component behaviour based upon the system model. The system is then simulated to determine under what conditions system failure occurs. This information is then used to improve the design.

Brehm[95], describes START/ES, an expert system based modelling tool for the performance and reliability analysis of complex computer systems. The system is represented on a computer screen as a series of interacting modules which the designer can manipulate as required. When the design is complete, it is turned into a complex mathematical model which is then evaluated for reliability and performance. The expert system is used to diagnose reliability and performance problems, acting as an aid to the designer in selecting appropriate components.

The factor that links the majority of the tools discussed above is that they depend on a coarse, rule based approach to defining the system behaviour. Murdoch[92] points out that this makes it difficult to model dynamic processes, ruling out the accurate treatment of faults and propagation of effects which involve dynamic variables. He

suggests that it is necessary that a logic model and a behavioural model both exist to carry out accurate fault analysis. This is in agreement with the authors earlier observations, see chapter 2, on the requirement for both types of behavioural model.

4.4.2 Formal Methods for failure analysis

Some of the first approaches to carry out failure analysis of complex system relied upon generating an abstract model which represented the system mathematically. This could then be analysed to determine whether certain failure states could be reached. The modelling approaches taken, include formal methods (Moser[90], Bowen[93]) and petri net analysis (Levenson[87], Console[92]). These tools are generally applied to safety critical systems, such as fault tolerant software for aircraft flight control, (Moser[90]), railway signal systems(Levenson[87], Atkinson[91]) and nuclear power plants(Bowen[93]).

When using formal methods, the process relies on the development of a detailed behavioural model of the system, written in a formal specification language, such as Z, CSP, or CCS. The system can then be anlaysed mathematically to see if any conflicts arise between the behaviour of the system elements to produce a fault. Levensosn[87] and Console[92] agree that the major difficulty in applying formal methods to the problem, is that the model is so complex that it is impossible to analyse successfully. Console[92] goes further, stating that to generate a model which is capable of being analysed, it must be so simple as to make the analysis worthless. There is a conflict between complexity and accuracy.

Petri-Net modelling was applied to the analysis of a railway signalling system (Levensosn[87]). It was found that human behaviour could also be represented in the model as well as fault based behaviour of hardware components and software. This is because petri-nets are based upon the modelling of states and the events that lead to a state being reached. In this instance, Timed Petri-Nets (specialised form of petri-net which can take into account time information) were used, because the effect of time on the behaviour of real time embedded systems is very important(Levenson[87]). Once the model has been created, it can be analysed using Bag theory (Peterson[81]). The approach was found to be successful and was applied to the failure analysis of a car petrol engine (Console[92]).

The mathematical modelling of systems in this way has proved to be very complex and because of the state based nature of the languages and techniques used, it is impossible to model dynamic variables. This is why it is mainly applied to verifying software for safety critical situations, where hardware such as sensors or transducers are just in ON or OFF states and it is the behaviour of the software which is being examined.

4.4.3 Dynamic, computer based modelling

Despite the fact, mentioned in section 4.4.1, that it is necessary that a logical model and a behavioural model both exist to carry out accurate fault analysis, (Murdoch[92]), dynamic model based fault analysis of systems is not common. Graham[95] accredits this to the high calculation overhead which is generated by this form of modelling. However, it can be argued that with the power of present day computers and software, the high computation overhead has reduced dramatically and such methods become viable.

One of the earliest applications of dynamic modelling for computerised failure analysis was Lehtela's Electronic Assistant (Lehtela[90]). An analogue circuit was built using a SPICE (electronic simulation and modelling language) type language in which the components are specified as nodes and joined together using links. Component behaviour is specified in a library file. The circuit is then verified to ensure that it works correctly under perfect conditions, when the model is correct the voltages and currents at each node are logged. Component failure behaviour can then be input by the user, to simulate failure. Deviation from the original voltage and current settings indicate a system failure, these are logged by the user, who can redesign the circuit. This early tool speeds up the analysis of complex circuits, but does not carry out a comprehensive search of the failure space, which is left to the user. On account of this fault, it is very easy to miss potential failure modes.

FIABEX (Russamano[92], Gachet[92]), was an attempt to provide an intelligent knowledge based system for the automation of FMEA. It achieved this, with mixed results, by the aid of a dynamic model which represented the systems functional and dysfunctional behaviour. By combining an expert system with the dynamic model, it was possible to induce faults in a logical and systematic manner and assess the system failure quantitatively. Gachet[92] employed FIABEX to analyse the Mars Rover's localisation unit, for which FMECA, FTA and HAZOP analysis were carried out.

FLAME II (Hunt[95], Pugh[96], Price[97]), is an electronic circuit simulator based upon a dynamic model and a system model. Again, the process is very similar to that described already. Failure Modes are induced in the circuit at the dynamic level and the effects are seen at the system level. No indication is given of whether FTA or the search of the failure space is generated automatically, although the results reported are promising.

Dynamic simulation modelling for safety was also carried out by Zhang[96], whilst examining the position of sensors for a mobile robot excavator developed at Lancaster University. The object of the simulation is to develop a model which simulated the position and sensing area of various transducers. Items could then be placed in the danger zone to determine whether or not they would be noticed by the transducers. The model produced was a continuous one. This form of model was chosen, despite being time consuming and expensive, because it was felt that discrete models do not take into account the truly dynamic situation found with robotics. For example, robots are continually changing their behaviour due to the input from sensors and the sequential actions programmed into them. They can only react to hazards or faults if a continuous state of sensor awareness is maintained. Due to the nature of all computer systems this continuous state of awareness is not possible, requiring high system sampling rates that need to be processed, However, modelling in discrete time steps of say 0.01 seconds, is nearly continuous and acceptable.

Izquierdo-Rocha[94] discuss the development of a simulation model for a nuclear power plant. This model is different from those discussed previously because it also involves the integration of human behaviour with the dynamic model. The object of the exercise was to determine not only the failure modes and effects of system failure on the system but also how errors attributable to humans affect the system. A similar case study to this is described in Luvic[98]. In this instance, a software model of the possible human error associated with the failure of a level crossing barrier is analysed. This is achieved by identify hazards, such as a broken down car on the crossing, and carrying out FTA and Cause Consequence Analysis (CCA). These failures are mapped into a Microsoft Access database, running ACCA (Advanced Cause Consequence Analysis software) and quantitatively analysed for reliability and probability.

4.5 conclusions and summary

It is seen that a number of fault analysis tools are available to the engineer. They are very rarely effective by themselves and a combined approach is recommended in much of the literature. (Seward[94], HSE[93], BSI[92]). Very few of the tools examined are proactive. The only technique reviewed which implies any true degree of proactive design for safety is CLASH, proposed by Seward[94], a combinational tool that integrates, CCA, FMECA and FTA into the design process. It is felt that these tools could be integrated into an EQFD design structure.

The computer based fault analysis tools reviewed show a consistency of approach. A systems model is developed, component failure modes identified, and the failure mode behaviour recreated. Failure modes are then instigated at the behest of the analyser or the program (for expert systems) and the results examined. The extent of automation and degree of accuracy within each tool varies greatly. Very little published information is given as to how fault trees are generated automatically and what the failure criteria actually is, if there is any at all, or if the effect of the failures are just observed. The majority of the tools are, very problem specific(Flame, Fiabex, IFME, etc), or unpublished research prototypes under development or testing(ACCA). None of them deal with mechatronic products, relating to the combined analysis of software, mechanical systems and electronics.

The fault modelling approach taken, depends greatly on how the failure for a system is specified and the behaviour of the system components. If the failure specification is very loose, for example, system fails if actuator B does not move when requested, the model can be very coarse grained and the dynamic behaviour very simple. This approach is used in many of the expert system based software packages (Palumbo[94] ,Brehm[95], Murdoch[92]). Whereas if the failure specification is very tight, for example, the load must have zero velocity within 0.06 seconds of a system failure being detected, then a detailed dynamic model is required (Murdoch[92]). Because the Automated Gamma Radiography system, under development, is part of a robot and the safety specification will be very tight, following the observation of Zhang[96], a near continuous dynamic model will need to be created.

The author proposes a modelling approach based upon a state-space model observer system(Ogata[70]), see figure 4-5. In this a model is used to observe the system's actual

behaviour, differences between the model and the actual system are then used to generate the control signals.

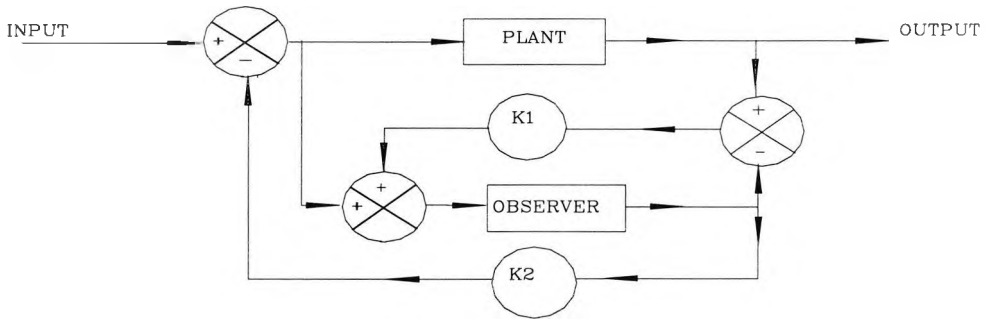


Figure 4-5 Model-Observer control system

In the proposed method, see figure 4-6, dynamic models of a component's failure modes are created in parallel to the component's correct dynamic behaviour. Component failure modes are selected and a behaviour sequence is initiated, and the dynamic behaviour of the failed component compared with the dynamic behaviour of the perfect component. When these outputs differ, a flag propagates through the system to indicate that a failure has occurred.

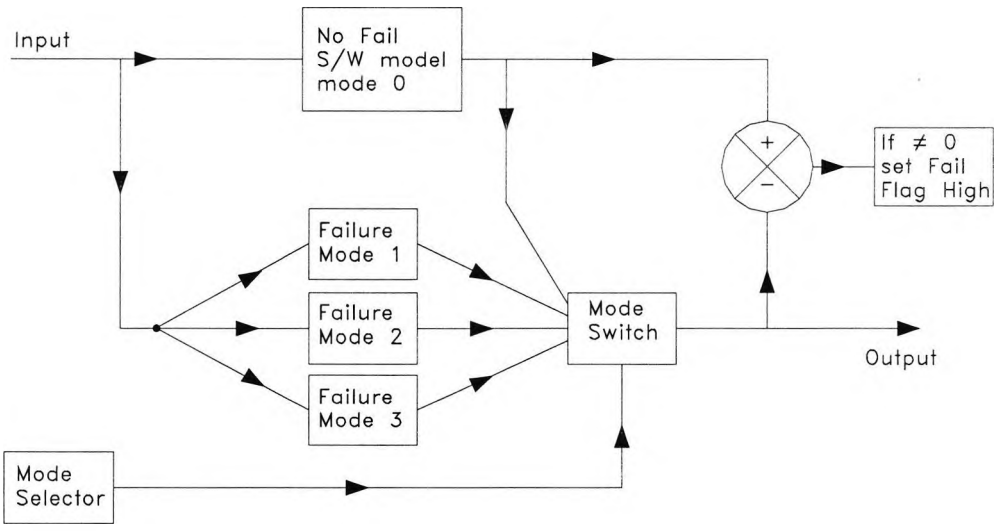


Figure 4-6 Schematic of software implementation of failure mode detection

If the overall computer control system does not recognise that a failure has occurred and react within a certain time, then the system can be said to have failed. When a failure has occurred, the combination of failure modes which contributed to the failure are logged. Using this method it should be possible to carry out a systematic search of failure mode combinations.

In this chapter and the previous two, a number of methods and tools were reviewed, which could be used for the proactive safety based design of safety critical mechatronics. A number of tools were selected which could possibly be integrated together. Chapter five, gives a detailed overview of the proposed methodology, plus examples of how the tools, selected in chapters 2-4, are employed.

5 Design methodology overview

5.1 Introduction

The objective of this chapter is to show how the design, systems modelling and failure analysis tools, selected in chapters two, three and four, contribute to the proposed design methodology and are integrated. A general overview of the proposed methodology will be presented, then an example of the use of each tool described. The examples used come from a variety of sources, ranging from common texts, to original research material.

5.2 Methodology Overview

The proposed methodology should have, as discussed in section 1.1.3, the following features.

1) A proactive stance to risk and reliability analysis of the product to be designed
2) A concurrent approach to the product design, based upon functional decomposition.
3) Employ some form of systems modelling as an aid to understanding the product behaviour.
4) Extend the system model into a software model, so that automation of component selection and failure analysis can take place.

The figure 5-1, below, shows the outline for the proposed methodology. It is primarily based upon Extended Quality Function Deployment with extensions to cover, proactive risk analysis, systems modelling and automated component selection and failure analysis.

The first stage consists of carrying out an initial survey of general concepts or products already on the market. From this, it is possible to identify various customer requirements and targets for improvement. It is also possible to carry out a basic analysis of the suitability of any general concepts which may occur to the designer. This prevents wasted time by removing clearly unsuitable concepts from the solution space. For example, the application of flying robots for radioactive source positioning.

Once a conceptual direction has been decided upon, CLASH can be applied to identify possible top level consequences of hazards. For example, crushed limbs possibly caused by incorrect control sequences for the robot end effector. These causes are

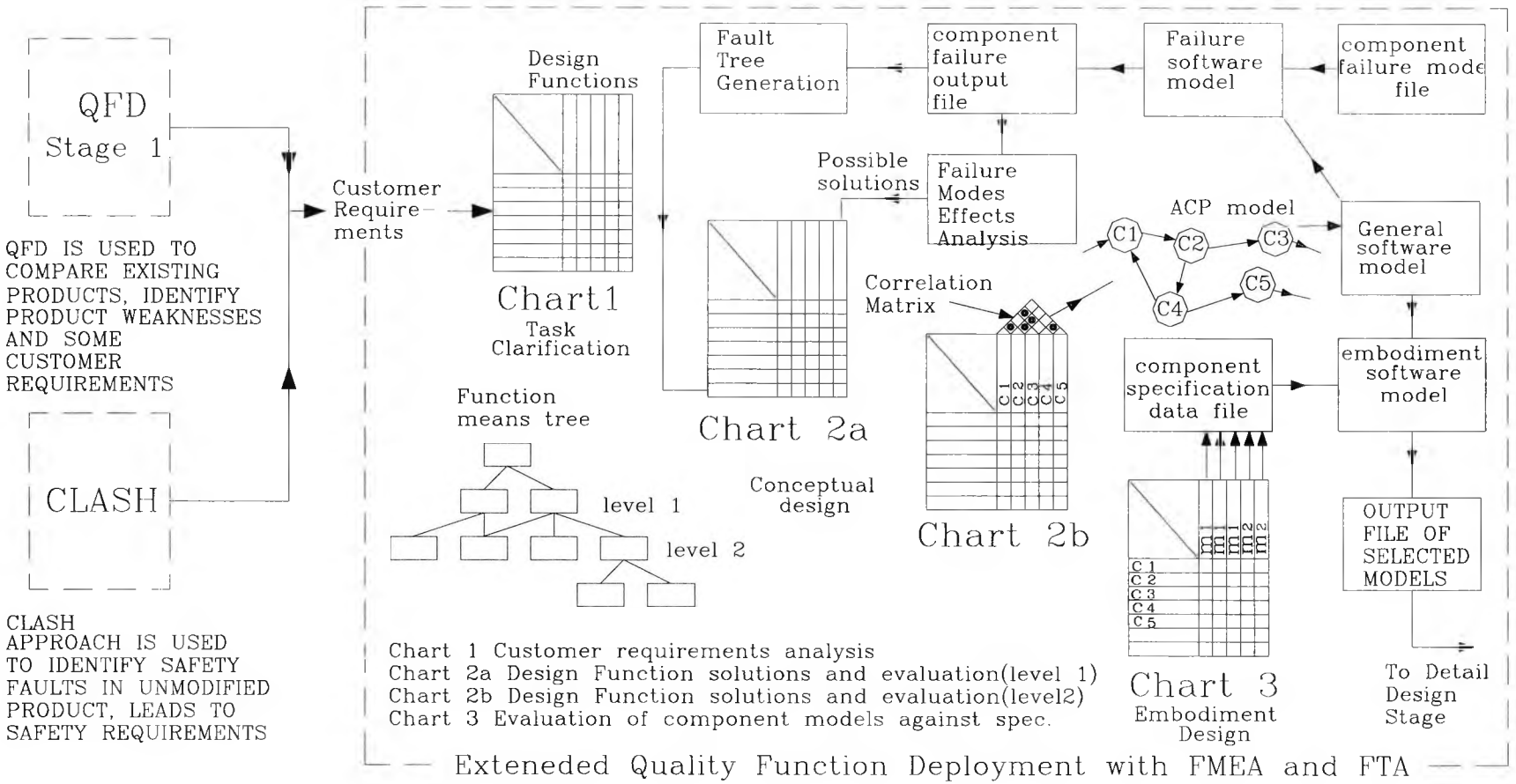


Figure 5-1 CLASH led EQFD with integrated Failure Analysis

expanded upon in a tree like manner to determine possible causes of the accident. The removal of these causes becomes part of the EQFD customer requirements.

The first stage of EQFD is to identify the possible functions which relate to the customer requirements. Each customer requirement should have an impact on at least one function. If a customer requirement does not have any impact on the function proposed, then that customer requirement is either, not required (unlikely), or the functionality to realise the customer requirement does not exist (Edney[93])

Once the functions have been identified, ways of realising those functions are generated. This is the conceptual design stage and involves tools, such as, Functional Analysis, Function Means Trees, Morphological Analysis and concept evaluation. Concept evaluation can be carried using Pugh's Concept Selection or Cross's Weighted Objectives. For this case, we will be using Cross's Weighted Objectives.

Once a concept has been firmed up, Embodiment Design is carried out, equating to a 3rd QFD stage. In this stage a systems model is produced, this is initially derived from the roof of the house of quality and comes in the form of the Correlation Chain (Comincini[94]). This indicates how various functions or components interact with each other, via either energy/power flows or signal flows, thus denoting the initial structure of the system. Using the Correlation Chain and the functional hierarchy of Function/ Means Trees, it is possible to generate ACP diagrams of the systems, subsystems and components and give the system behavioural characteristics.

The ACP diagrams provide the initial model for the software model, which is produced using VISSIM. This first software model is used to determine the specific electro/mechanical components for the system, i.e. the sizing of motor/gearbox/lead screws, etc., see figure 5-2 which shows a schematic of the methodology. This is achieved by creating a stage three QFD chart which looks at combining different ways to realise the specification for any moving components. The specification for each of the components is entered into a database. The database is searched for compatible components by setting the specification for the system and then combining different component specifications into the software model. The components are then dynamically tested by applying an input and then examining the system output to determine whether the specification has been reached. A component combination, which meets the specification, is then logged as being successful.

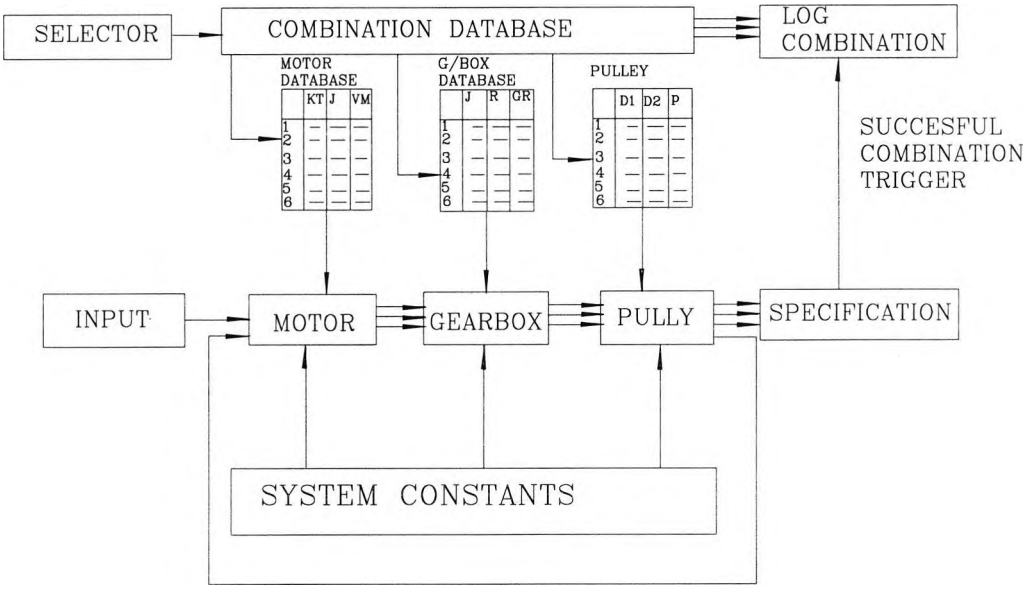


Figure 5-2 Schematic for approach to automatic component selection

Once the essential components for a system have been selected the VISSIM model can be expanded to include other components, control software and hardware, the behaviour of the system is expanded to include safety features, such as hardwired interlocks. It is this model which is used as the basis for the initial failure analysis. For each component an FMEA is carried out and the failure mode behaviour for each component modelled in the VISSIM model. Combinations of these failures are introduced into the software model and the effects observed. The system can be said to fail if a failure is detected and the system is not in a safe state. The failure combinations are systematically searched to determine the component failure modes. A failure is detected when the behaviour of the perfect model differs from the behaviour of the model with a particular failure function, see figure 5-3. When a failure is detected the failure combinations which contributed to that failure are logged. These combinations are then analysed to produce the reduced fault tree.

The removal of the causes of system failure are then entered as requirements in a sub QFD stage 2 chart. Functions are realised, concepts generated and evaluated. Because new components and behaviour will be introduced into the system, a further iteration of FMEA, system model creation, failure simulation and FTA analysis need to be carried out.

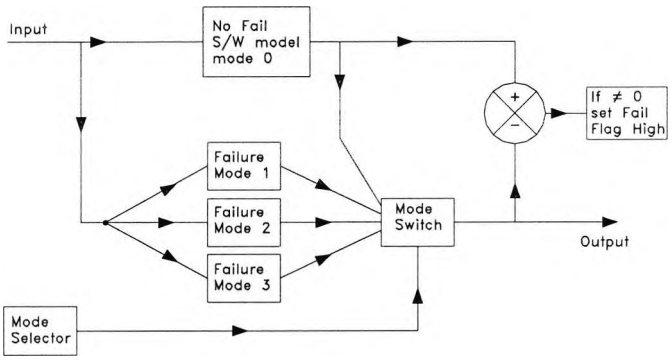


Figure 5-3 Schematic of software implementation of failure mode detection

5.3 Initial risk analysis (CLASH)

The first stage in the proposed methodology is a CLASH analysis of the product. This is derived from the initial specification and experience of producing similar products. By way of demonstration the product to be designed is a portable log splitter. The specification, adapted from Edney[93] is given below:

> Design a portable, low cost, easy to operate machine to carry out the initial breaking up of logs, 2ft long and 1.5 ft in diameter, for firewood. The machine should be capable of producing firewood from the toughest timber in the USA and Canada.

The first stage of CLASH is to identify possible consequences of using such a device. In this instance, death is the worst consequence that can occur if the device is hazardous. This is based upon the inference that some sort of violent cutting, crushing or splitting operation would be carried out to realise the device's functionality. Once the consequences have been identified, reasons for the fault are determined. These are laid out in a logical manner, similar to a Fault Tree Analysis, where consequences are derived as a logical combination of events.

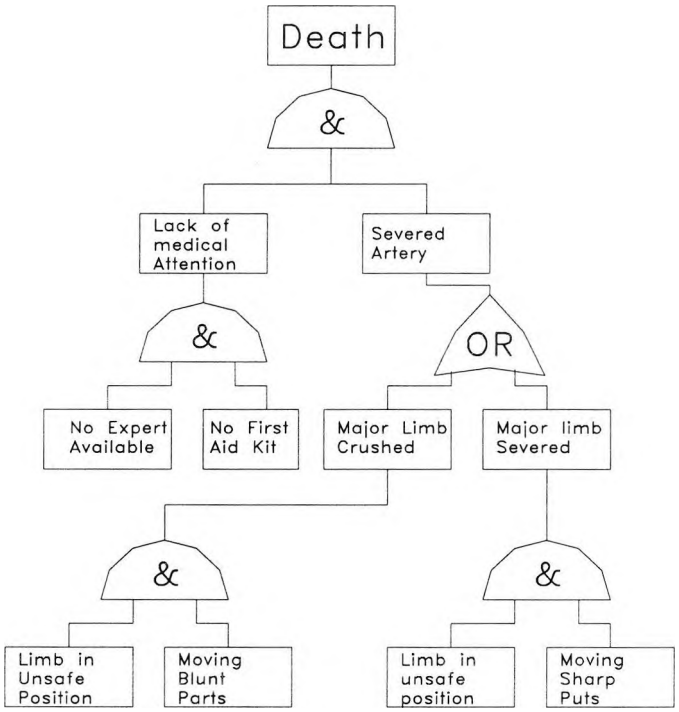


Figure 5-4 Example of CLASH analysis as applied to a wood splitter

In the example detailed in figure 5-4. one way in which death could be caused, is by severance of a major artery and the lack of medical of attention. This is an AND relationship because it is unlikely that a person would survive a severance of limb without rapid medical attention. The cause of the lack of medical attention can be attributed to the lack of availability of trained personnel and medicinal supplies. These are personnel considerations.

In terms of the machine being hazardous, a severed artery can be caused when a limb is severed or crushed. Actions which are necessary to produce a crushed limb are if a limb is in an unsafe position and there is access to moving blunt components. A similar situation exists for severed limbs. From this analysis the following safety based customer requirements were generated.

From lack of medical attention

1) Prevent lone operation
2) Available medical supplies
3) Available medical skills

From crushed and severed limbs

1) Prevent access of limbs to blunt or sharp, moving parts

These requirements are added to the specification and entered into the first level QFD chart, where functions can be generated to realise the requirements.

5.4 Current market analysis

In Edney[93] it was noted that no solution presently existed for the Log Splitter, so this stage was not carried out. Most products, however, are not truly original, they are normally the result of incremental or adaptive design (Pahl & Beitz[84]), the process whereby the product evolves over a long time period. When this is the case, a comparison of the ability of current products on the market to meet customer requirements, is carried out. If other solutions existed for the log splitter, the safety features indicated by the CLASH analysis would be identified as customer requirements.

Customer Requirements \ Competing Products or Initial Concepts	Importance rating (5 high, 1 low)	Thermometer 1	Thermometer X	Thermometer Y	Target direction	Target rating
Quick to read	5	3	4	3	+	4
Easy to use	2	4	4	2	=	4
Easy tip replacement	4	2	3	3	+	3
Rugged	1	2	2	3	+	3
Low cost tip replacement	3	3	3	3	=	3
Safe to use	1	4	4	4	=	4
Small	1	2	5	4	+	5
Lightweight	1	3	5	5	+	5
Product rating		51	69	56		

Figure 5-5 1st stage QFD analysis of competing electronic Thermometers

An example of this product comparison process can be seen in figure 5-5, (adapted from Salome[95]). The example employed is based upon a comparison of three electronic thermometers.

Three types of design are compared, Thermometer 1 is the current product produced by a company, and Thermometers X and Y are competitors products. The first stage is to identify the customer requirements, this information is obtained from what the customer

perceives to be the most important features of a product. In this instance the customer identifies:-

1) Quick read time:- i.e. rapid response
2) Easy to use
3) Easy tip replacement, each patient should have a new tip placed on the thermometer
4) Rugged
5) Low cost tip replacement
6) Safe to use
7) Small
8) Lightweight

The customer is then asked to indicate which if these features is the most important and rank them on a scale of 5-1 (5 high, 1 low). Each thermometer is then ranked under the various requirements, for instance thermometer 1 is easy to use (4) but is not very small (2).

Each product is then ranked by multiplying the customer requirement ranking by the importance weighting. These are then summed to give a final product score. By comparing these scores, it is possible to determine the best product, in this instance thermometer X, with a score of 69.

A comparison across the customer requirements is also carried out, this identifies the areas in which the company product should be improved. For instance, examining the scores for the requirement 'quick to read', show that it should be improved to compete favourably with thermometer X. This is indicated by a + and a target rating of 4, in the last two columns. Because the importance rating for this requirement is very high (5), the improvement in this area for a new product is vital. The other area of improvement is the ease of tip replacement, (importance rating 4, requirement rating 3, target rating 4).

The customer requirements and improvement direction indicators are then passed to the first stage of EQFD, which examines further customer requirements and develops the functions for those customer requirements.

5.5 Customer requirements and function analysis

The 1st stage of EQFD is to expand the Explicit customer requirements identified in the CLASH and Customer analysis by identifying Implicit requirements (requirements that a system specifier may not know are required). The idea of Implicit and Explicit requirements is taken from DFD but it has also been noted in other parts of the literature, (e.g. Sontow[92], Ulman[92]), under different titles.

Customer requirements are entered in the left hand side of a matrix and become WHATS (WHAT the customer wants). Implicit requirements are based upon the definition of 'Customer', in this instance defined as any department or person who comes into contact with the product throughout the design and manufacturing process (Edney[93], Sontow[92]). The requirements of manufacturing, for example, could relate to constraints such as the processes available for manufacture, or the rate of production. Another requirement maybe as vague as 'easy to manufacture'. These Explicit and Implicit customer requirements are what the design must satisfy.

In figure 5-6 (from Chamberlain[94]), the top level customer requirements for the gamma radiography robot are shown down the left hand side of the matrix. The explicit customer requirements are features such as 'easy to use', 'cost effective', or 'low cost'. Implicit requirements, are features such as 'remote operation', (no personnel are allowed to be within a certain distance of the source due to the radioactive nature of the source), 'safety' and 'maintainable'. Manufacturing concerns are not considered too deeply at this stage. Also note the use of a importance weightings relating to each customer requirement.

Along the top edge of the matrix are entered solution neural functions or HOWS (HOW the requirements are to be met). The term 'solution neutral function' is used to describe a function which does not specify how the function it is to be met, i.e. the function, 'main power supply', does not specify any means to realise the function, where as 'electric power supply' does. Other functions, indicated in figure 5-6, are the tool handling configuration (i.e. cartesian, SCARA, etc.), materials, and ease of assembly.

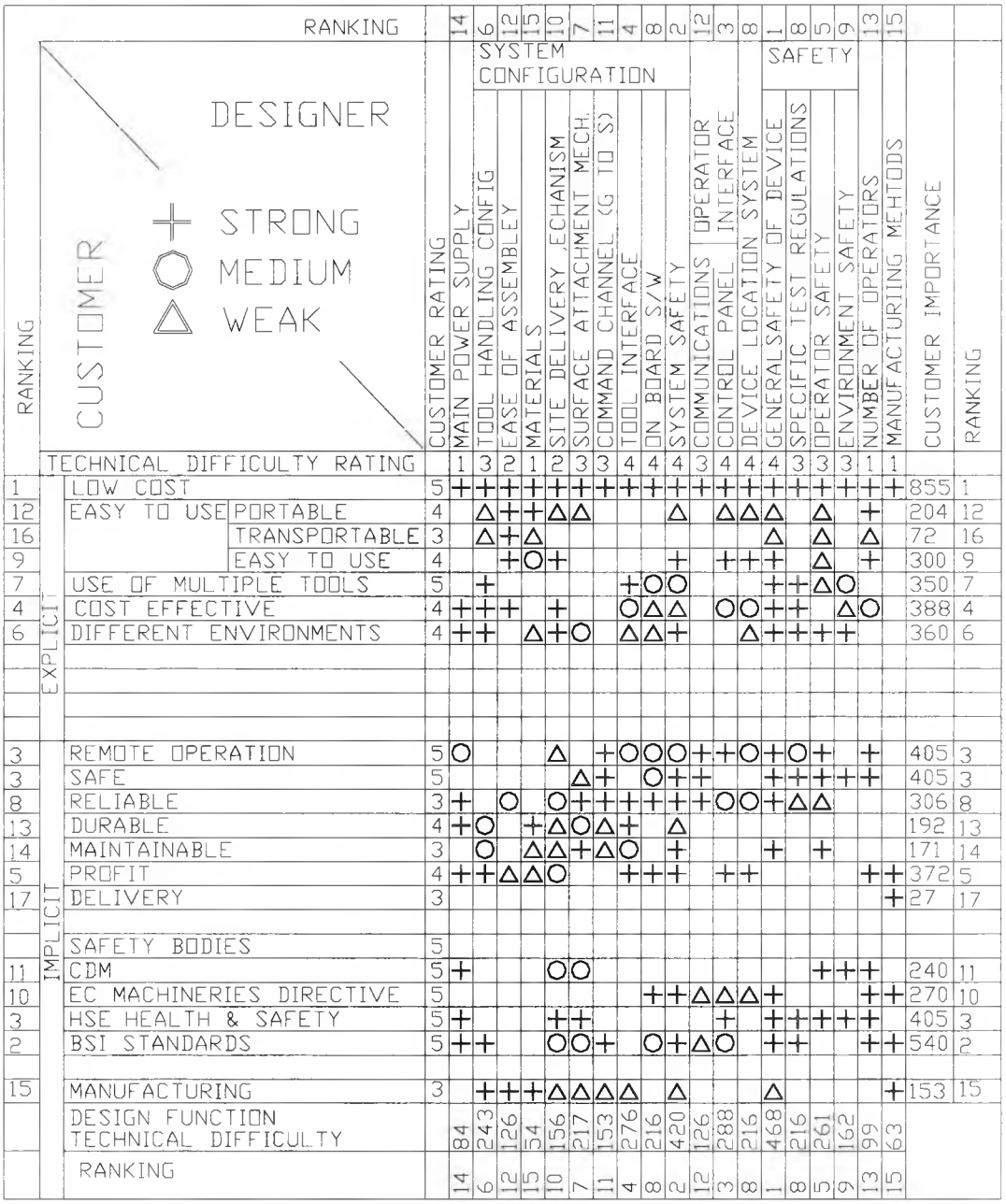


Figure 5-6 Top level EQFD chart for Gamma Radiography Robot

The strength of the relationship between each customer requirement and fucntion is then indicated in the centre of the matrix. A + signifies a strong relationship, a O, a medium relationship and a Δ, a weak relationship. For instance, the function, 'materials' has a strong relationship with portable, because the density of the material used, affects the portability of the system. It has a weak relationship with 'transportable' because this requirement transportable relates to the method used to get to the locality

of the test, a medium relationship with 'easy to use' is shown because it relates to the machining characteristics.

Each of the relationship symbols have a numeric weighting associated with them of 3, 6 and 9. These can be used to determine the relative importance of customer requirements and functions. Each of the relationship symbols for a particular requirement, are summed across the columns and multiplied by the customer rating (importance weighting) to give the customer importance. These totals are then ranked to indicate the most important requirements. A similar process is carried out for the functions, the sum for each column ranked to give an indication of the most important functions, this can be further adapted by multiplying the column totals by a technical difficulty rating. This new total, when ranked, indicates the areas of greatest technical risk to the project.

5.6 Concept selection

The next stage in the EQFD process is concept generation and selection. This process starts by moving the functions generated as HOWS in the previous section to the WHATS in the next EQFD chart. Conceptual methods for realising each function are then generated using various creativity techniques. Some functions may be realised by more than one concept and some concepts may adversely affect other concepts.

Function Means Trees, Morphological Charts and benchmarking tools, such as, Cross's Weighted Objectives Method, are all useful tools which allow the designer to map the functions, select design combinations, and evaluated those combinations logically.

To show the use of these tools an example will be given based upon the development of a device to transmit the spoken voice over a distance.

5.6.1 Function Mean's Trees

The top level solution neutral function, 'transmit spoken message over a distance', See figure 5-7, can be realised in an electrical , optical or functional from. In this instance, the electrical form is analysed. The top-level functions are, converting sound to electricity, transmitting the electrical signal, and converting the electrical signal back to sound. Electro-mechanical and electrical means can be used to realise these functions. Each of these could be considered to be organs or sub-systems as they are further broken down into sub-functions. The sub functions relate to the stages of transforming sound, into electrical signals, conditioning the signal, transmitting the signal,

conditioning again and converting the signal back to sound . It is these sub functions which will be used for the 2nd Stage EQFD analysis. Ways of realising the sub-functions are indicated by the components (or parts) at the lowest level.

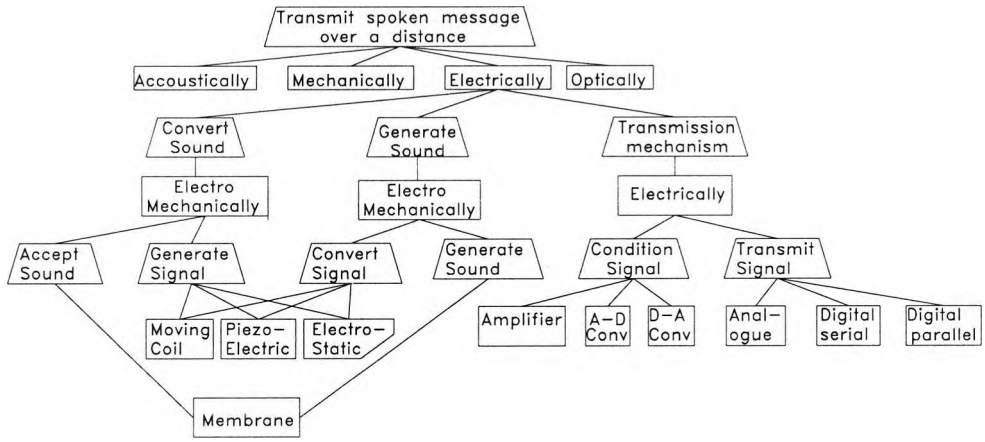


Figure 5-7 Function Means Tree for "transmit spoken message over a distance"

Hows / Whats	Membrane	Moving coil	Electro-static	Piezo-electric	A-D converter	Amplifier	Digital Serial	Digital parallel	Analogue	D-A converter
Accept sound	●									
Generate Signal		●	●	●						
Condition Signal					●	●				●
Transmit Signal							●	●	●	
Convert signal		●	●	●						
Generate sound	●									

Figure 5-8 Stage 2 EQFD chart

5.6.2 2nd Stage EQFD

The relationship between the sub-functions and ways of realising them can be indicated on a second stage EQFD chart (see figure 5-8). Note that every function has a way of being realised and some means can be used to realise more than one function. For instance the membrane is used to accept and generate sound, also the same means can be used to convert sound to electricity and vice versa.

5.6.3 Morphological charts

At this stage, no decision has been made about bringing together parts into organs and complete concepts. To achieve this task, morphological charts are used, which provide a way to indicate combinations of components that meet the top level function.

Functions	Means		
Accept sound	Membrane		
Generate Signal	Moving coil	Electro-static	Piezo-electric
Condition Signal	Amplifier	D-A converter	A-D converter
Transmit Signal	Analogue	Digital Serial	Digital parallel
Convert signal	Moving coil	Electro-static	Piezo-electric
Generate sound	Membrane		

Figure 5-9 Morphological Chart for "transmit spoken message over a distance"

To generate a morphological chart, see figure 5-9, the functions to be achieved are written down the left hand side of a matrix, and the concepts which are used to realise these functions are placed next to them. The concepts for each function are examined and mixed accordingly, and complete concepts generated, indicated by arrows. In the example, figure 5-9, three possible concepts are indicated.

1) Concept 1:- A membrane is used to accept and generate sound. Electrostatics and an amplifier are used to transform the signal into an electrical signal and condition it. Analogue transmission is chosen and a moving coil meter is used to manipulate the membrane for sound generation.
2) Concept 2:- A membrane is used to accept and generate sound. A moving coil meter is used to change the sound signal into an electrical signal. The transmission medium is a digital serial link, this requires A-D and D-A

conversion. A moving coil meter is used to manipulate the membrane for sound generation.

3) Concept 3:- This concept is almost identical to concept 2 the only difference is the transmission medium, which is in this case, a digital parallel link.

In the terms of EQFD, these concepts are given the title of System Architectures.

5.6.4 Benchmarking

Once the System Architectures have been developed, they need to be compared to see how well they achieve the customer requirements. This task will be carried out using Cross's Weighted Objectives method.

The explicit customer requirements and weightings (1-9) for this example were:-

1) High Quality sound (Very Important, therefore 9)
2) Low Cost (Medium to Low importance, therefore 6)
3) Small size (Medium to High importance, therefore 7)

The implicit requirements and weightings were:-

1) Long distance before re-amplification (Very Important, therefore 9)
2) Ease of manufacture (High importance, therefore 8)

Each of the System Architectures are evaluated against this criteria as shown in table 5-1.

Customer Requirements		Weighting	Concept 1	Concept 2	Concept 3
Explicit	High Sound Quality	9	7	10	10
	Low-Medium Cost	6	7	6	5
	Small Size	7	5	7	7
Implicit	Amplifier Distance	9	7	8	5
	Ease of Manufacture	8	8	6	5
	Totals		267	295	254

Table 5-1Cross's Weighted Objectives chart

For example, taking the requirement of high sound quality, Concept 1 receives a 7 as analogue sound transmission is good but not perfect, whereas Concept's 2 and 3 get the maximum of 10 because digital sound is of a very high quality. For the requirement, amplifier distance, Concept 1 receives an 7 whilst, concept 2 receives an 8, concept 3 only receives a 5, because parallel digital signals are only good over short distances.

The totals for each concept are generated by summing the product of the importance weighting and the score for each requirement. In this instance it can be seen that concept 2 would be chosen because it has the highest score of 295, compared with 267 for concept 1 and 254 for concept 3.

5.7 Systems modelling

The objective of this section is to give an introduction to the use and generation of system models using Activity, Channel, Pool diagrams (ACP). It starts with an overview of how to create a systems model using ACP diagrams. This is expanded upon in the second section, where it will be shown how to integrate this model with EQFD. Once the ACP model has been created, it is possible to recreate it dynamically, in software, using VISSIM, the final stage in the modelling.

5.7.1 ACP diagrams

The ACP method is a method for defining complex systems and sub-systems, the version presented here is adapted from the version presented in MASCOT[87], enabling it to be used for the design of mechatronic products. As in the original version three types of diagram exist in the notation, these being:-

ACTIVITY - this is the name for a complete system or subsystem with a system boundary. Signified by circle,

CHANNEL - This is anything that conveys information across the system boundary. It may be a length of wire, an operator button or a display, or a mechanical link. A channel has two ends and no branches or junctions in the middle.

POOL - This is an identifiable engineering component; it may be software, electronic hardware, mechanical nuts and bolts. At this level there is no distinction between hardware and software, electronics or mechanical components.

5.7.2 Constructing an ACP Diagram

The example of a motor speed control system is used to explain the five stages in the application of the method. The top level ACP diagram for this is shown in figure 5.10 and the breakdown for the power supply unit (PSU) Activity , in figure 5.11

Stage I Draw a system boundary and mark all the channels as either INPUTS or OUTPUTS. Figure 5.10 demonstrates this.

Stage II. Identify the main SUB-SYSTEMS and mark the connections between them. Figures 5.10 and 5.11, demonstrates this.

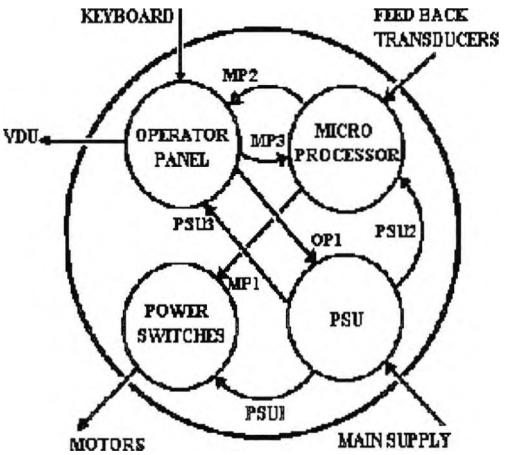


Figure 5-10 Example of Top Level ACP diagram (motor speed control system)

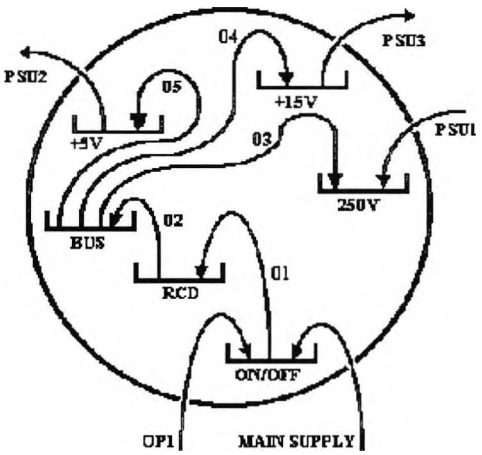


Figure 5-11 Break down of PSU activity

Stage III. Write down the PSEUDO CODE to identify the system, and apply the theorem that each channel should be referenced exactly TWICE. Each channel has two ends only. If the same signal goes to different sub-systems a JUNCTION BOX is required as an identifiable activity.

The formalisation of activity for the example in figure 5.10 is given by: -

ACTIVITY: Motor_Speed_Control (INPUTS: Main Supply, Keyboard, Feedback,
OUTPUTS: VDU, Motors)

ACTIVITY: Operator_Panel (INPUTS: Keyboard, MP2, PSU3
OUTPUTS: MP2, MP1)

ACTIVITY: Microprocessor (INPUTS: Feedback, MP3, PSU2
OUTPUTS: MP2, MP1)

ACTIVITY: Power_Switches (INPUTS: MP1, PSU1
OUTPUTS: Motors)

ACTIVITY: PSU (INPUTS: Main Supply, OP1
OUTPUTS: PSU1, PSU2, PSU3)

END

Stage IV. Breakdown the sub-activities into POOLS or identifiable engineering equipment. In a similar manner to the way in which the ACTIVITES were defined, a POOL is described by its inputs, outputs, constants, variables and behaviour. The formalisation of the PSU Activity for the example in figure 5.11 is given by:

ACTIVITY: PSU (INPUTS: Main Supply, OP1
OUTPUTS: PSU1, PSU2, PSU3)

POOL: ON/OFF (INPUTS: Main Supply, OP1,
OUTPUTS: 01)

BEHAVIOUR{ IF OP1 = 1 THEN 01 = Main Supply
ELSE
01 = 0}

POOL: RCD (INPUTS: 01,
OUTPUTS: 02
CONSTANTS:Current Limit)

BEHAVIOUR{ IF 01 > Current Limit 02 = 0
ELSE
02=01}

POOL: BUS (INPUTS:02
OUTPUTS:03, 04, 05)

BEHAVIOUR{ 03 = 02
04 = 02
05 = 02}

POOL: +5V (INPUTS: 05,
OUTPUTS: PSU2)

BEHAVIOUR{ PSU2 = 05/50
IF PSU2 < 0 PSU2 =0 }

note:- If the system is running correctly, the input 05 = 250 volts AC, therefore 250/50 = + 5 volts output on PSU 2.

POOL: +/-15V (INPUTS: 04,
OUTPUTS: PSU3)
BEHAVIOUR{ PSU2 = 04/16.667}

POOL: 250V (INPUTS: 03,
OUTPUTS: PSU1)
BEHAVIOUR{ PSU1 = + 03}

Stage V. Each of the pools and associated channels can now be related to the equipment list:

POOLS

ON/OFF	-	Main Switch
RCD	-	Residual Current Device (Circuit Breaker)
BUS	-	Distribution Bus Bars
+5V	-	5 volt PSU
15V	-	+/-15 volt PSU
250V	-	Connector Plug and Socket

CHANNELS

OP1	-	Operator's fingers
Main Supply	-	13 Amp plug and lead
PSU1	-	3 core mains cable
PSU2	-	2 core cable
PSU3	-	3 core cable, low voltage

Although the logic in this example is simple it is possible to define very complex behaviour, which can be either logical or mathematical.

5.7.3 ACP diagrams and the roof of the House of Quality

The ACP methodology can be applied to EQFD, by employing the interaction matrix of the roof of the 'House of Quality'. The interaction matrix is used to identify positive and negative relationship between different functions. In this example the product to be designed, is a drive for an axis of a robot.

Figure 5-12, shows a section of a stage 2 EQFD chart with an interaction matrix, the means of realising the drive and its controls. If the Brushless Servo motor is taken as an example, there is a relationship between the Motor the Tachometer and the Encoder. The relationship is neutral because they make very little difference to the behaviour of the motor, however, a negative relationship exist between the motor and the belt and pulley, this is because the belt and pulley increase the inertia of the system and increase the torque the motor has to produce to accelerate the load.

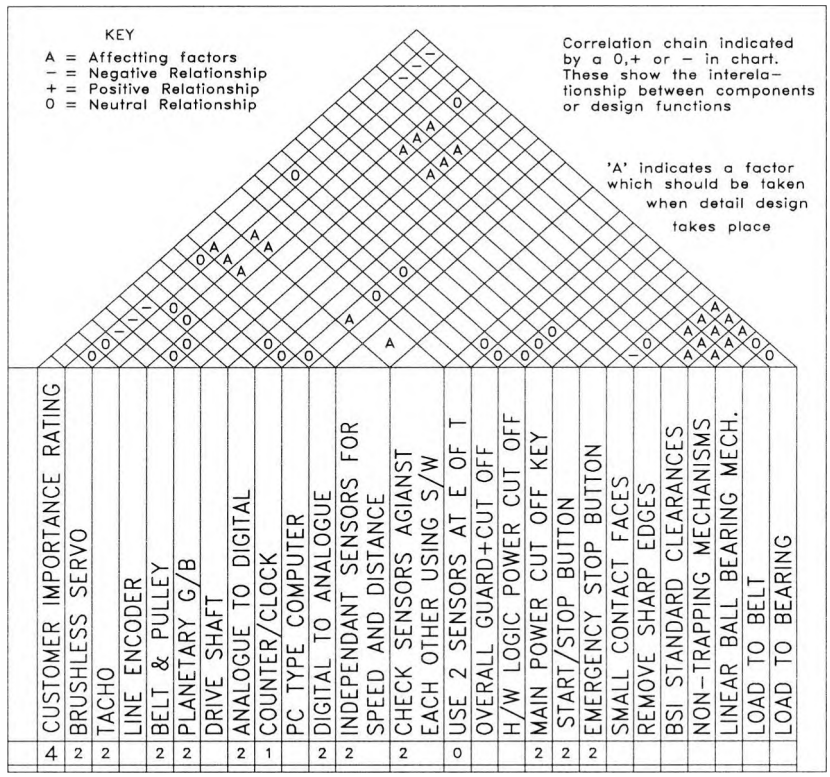


Figure 5-12 EQFD stage 2 chart showing the interaction matrix

Once a concept has been decided, the interaction matrix is used to derive the top level ACP diagram, see figure 5-14. Channels between Activities are indicated by the relationships between the means selected. The functionality of an Activity is contained within Pools buried in the Activity. In this instance the notation has been extended further by indicating whether the Channels contain continuous (solid line) or discrete (dashed line) signals or power.

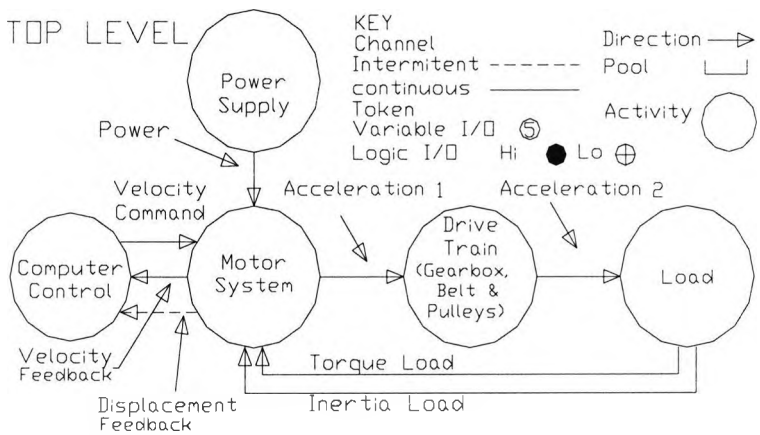


Figure 5-13 Top level ACP diagram for motor drive system

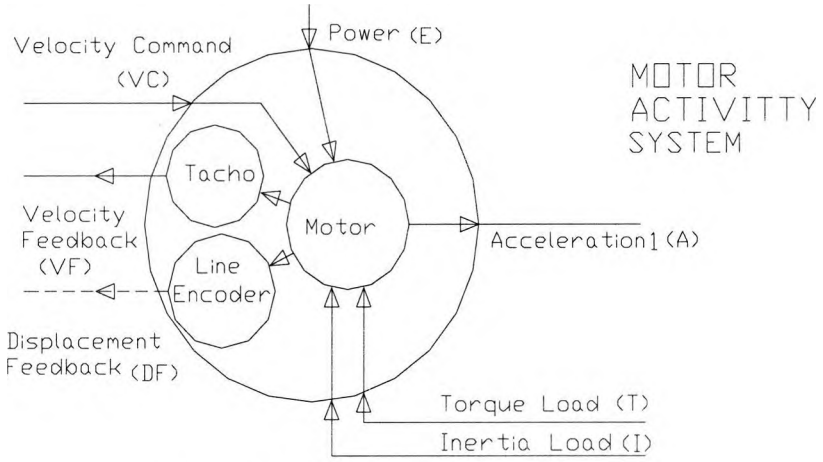


Figure 5-14 Motor Activity System

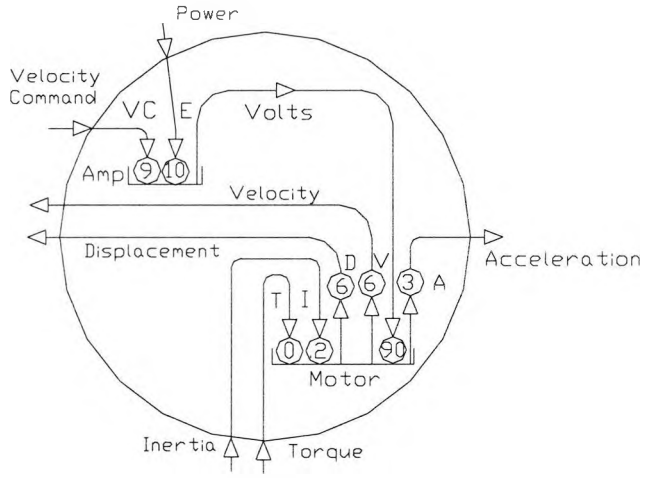


Figure 5-15 Motor Activity

Figure 5-14 shows the breakdown of the motor activity system, the view is taken to a lower level in figure 5-15. In this figure, tokens are placed at the bottom of the channels to indicate the values entering and leaving the various Pools at a particular time. The functionality of the Amp pool is based around amplifying the voltage, input to the motor is based upon the velocity command:-

$$\text{Volts} = \text{Velocity Command} * \text{Power}$$

$$\therefore \quad \text{Volts} = 9 * 10 = 90 \text{ Volts}$$

The motor Pool functionality can either be defined by the torque speed curve or depending upon the sophistication of the model the motors Laplace transform.

5.7.4 ACP and VISSIM

The Vissim model is generated directly from the ACP diagram, each hierarchical level of the ACP is represented as a block diagram within Vissim, the lowest levels being a

pool. The system behaviour for a pool, defined by the pseudo code in the ACP diagrams is recreated using blocks which represent mathematical or differential equations and boolean operations, these blocks are joined by links representing channels. The example chosen is that of the motor speed control unit used in the section 5.7.1. The Vissim equivalent of the top level ACP level diagram (figure 5-10), is shown in figure 5-16 and the PSU activity of ACP diagram, figure 5-11, shown in figure 5-17.

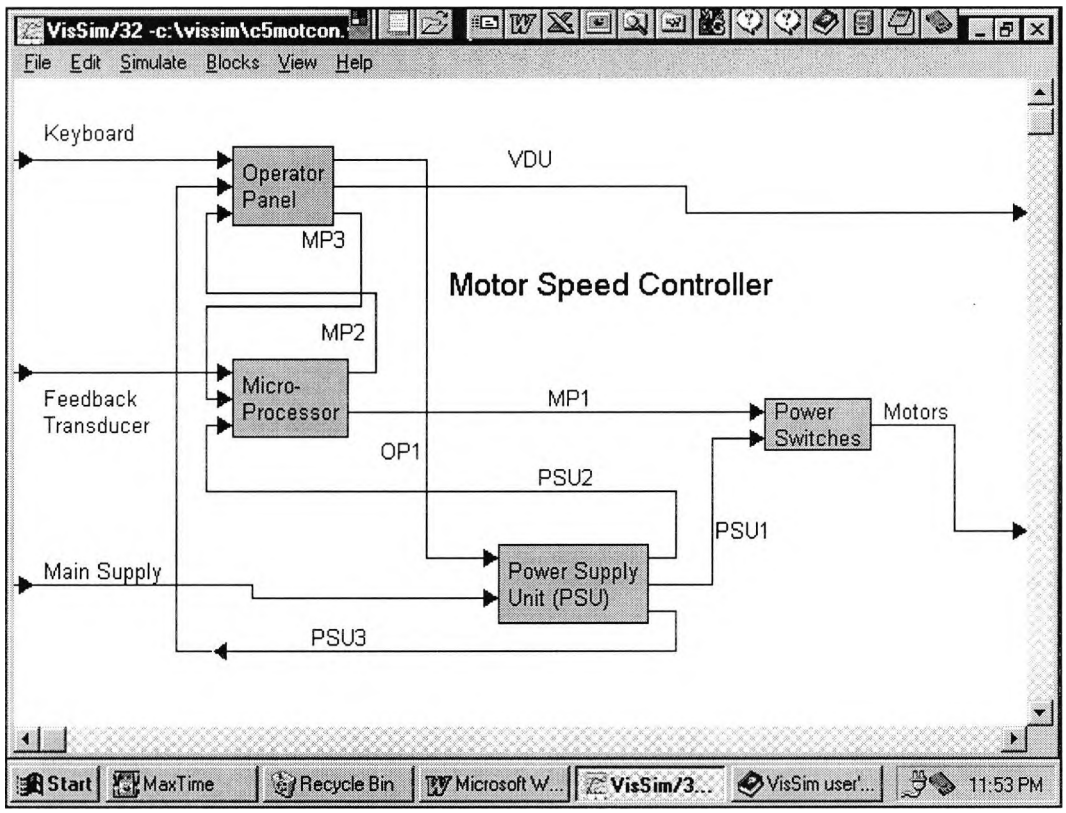
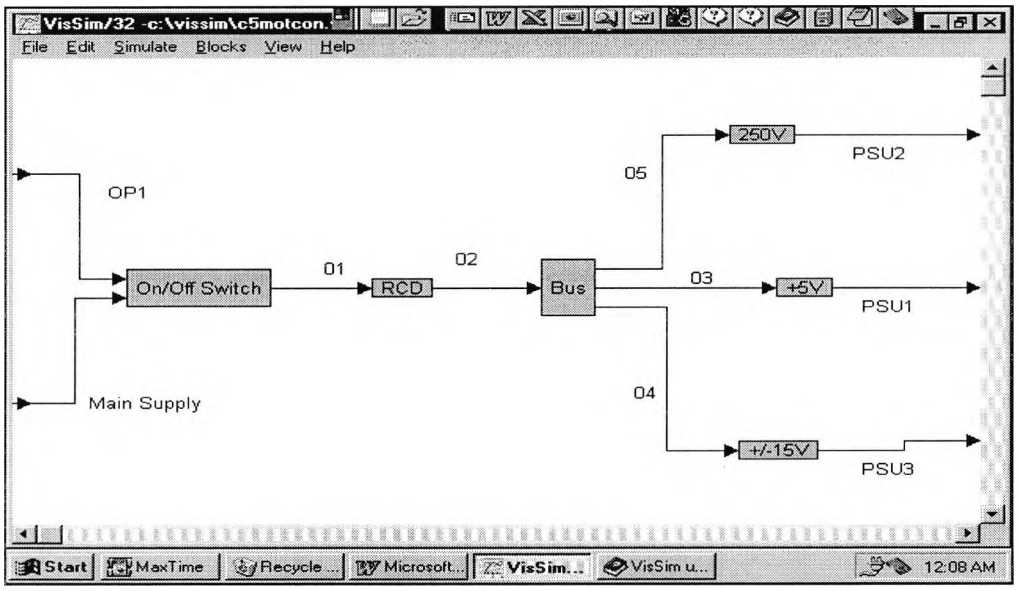


Figure 5-16 Vissim diagram of Motor controller

At this level, the blocks represent pools and the contain the behavioural information required for the simulation of each block. For instance the On/Off switch (figure 5-18) is represented by a 'merge block' , when the switch is 'on' (OP1 = 1) and the 250 V Mains



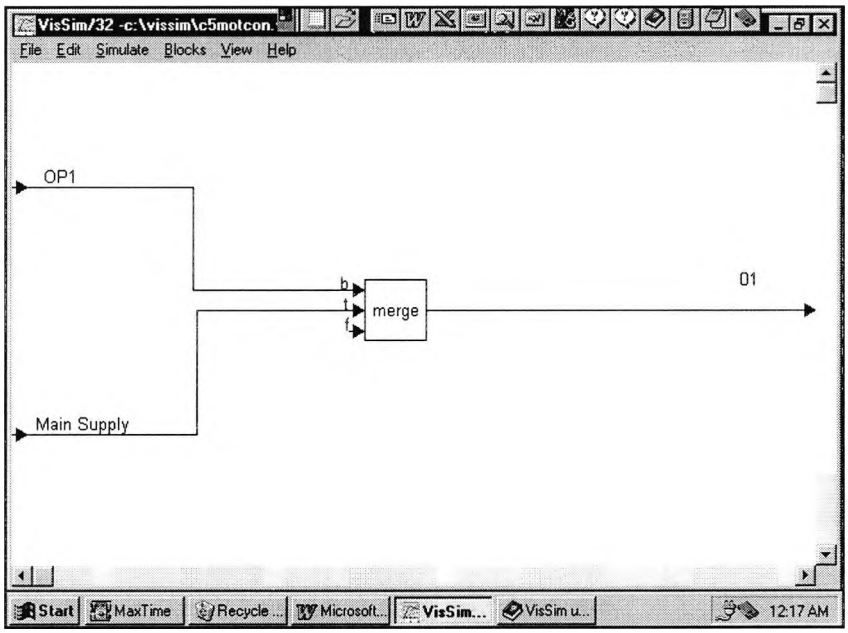


Figure 5-17 Vissim model of PSU activity

Figure 5-18 Vissim model of ON/OFF SWITCH Pool

Supply is on, this power is passed to channel 01. When the switch is turned off the Mains Supply is not passed to OP1. Another example is shown in figure 5-19, in this instance, the +/- 15 transformer is represented. The output from the bus, via channel OP4 (250 volts), is divided by 16.6666 to give the required +/- 15 Volts, which is output via PSU3.

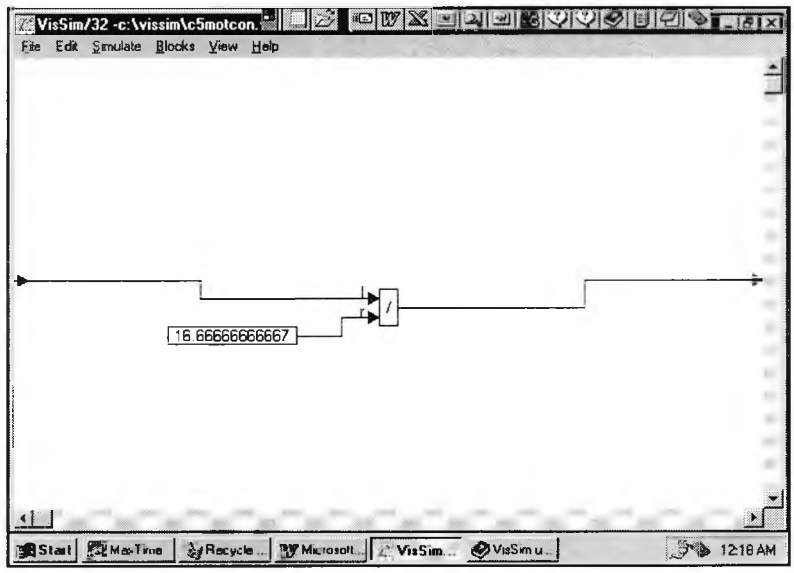


Figure 5-19 Vissim model of +/-15 V PSU Pool

5.8 Computer models for component selection

Computer programs for component selection have been around for many years, however, they are normally restricted to one type of product(e.g. select a motor, given the torque/speed requirements) and are limited in the calculations that they can carry out. With an example program supplied by Vickers (Vickers[93]) In this program the user is required to enter the velocity curve (trapezoidal or triangular) required of the motor, the gearbox or drive details (including inertia's), loads and efficiencies. Once this information is entered, a database is searched to find a suitable motor. No guidance is given as to different drive combinations, which may produce a more suitable selection. The velocity profiles are statically produced, no dynamic analysis is carried out, and the control method employed in normally ignored. These features are very important if the motor/drive system is of a high specification, and will undergo unusual velocity acceleration curves, the case normally present in drive systems for robot arms. The development of ACP based, VISSIM models for the computerised selection of components for embodiment design was carried out by the author to address these problems.

The method is based around creating a VISSIM model of the system, from the ACP diagram. Each component has a certain functionality attached to it, which can be modelled mathematically. Combinations of components are systematically selected

The method is based around creating a VISSIM model of the system, from the ACP diagram. Each component has a certain functionality attached to it, which can be modelled mathematically. Combinations of components are systematically selected from databases, and their important details entered into the model. An input is given to the model and the output evaluated against the specification criteria. When the criteria is met, the combination of components are logged in a separate file.

This form of modelling is identical to an Object Orientated approach. Booch [91] states that objects have Identity, State and Behaviour. Each object of a particular type of component in the model has a separate identity (e.g. catalogue code), state is defined by the components attributes or specification details (e.g. pulley radius, piston diameter) and the components behaviour defined by physics/mathematics (torque produced at a shaft from a pulley = Force * Radius). An example based upon the selection of specific components for a simple pneumatic lifting mechanism will now be given.

5.8.1 System analysis and ACP diagrams

The concept selected for analysis consists of a pneumatic piston and two connected pulleys of different diameters which rotate around the same axis.

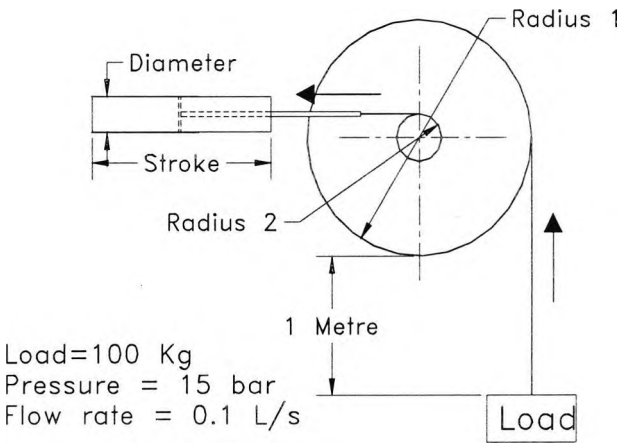


Figure 5-20 Schematic solution for lifting problem

The complete concept is shown in figure 5-20, a belt is attached to one end of the piston and connected to the smaller pulley, as the piston retracts, both pulleys rotate and a weight attached to the larger pulley is raised. The weight is 100 Kg and it must be raised through 1 metre in 10 seconds. Limitations are that the pressure is limited to 15 bar and the volumetric flowrate limited to 6 litres a minute. Four small pulleys, four large pulleys and four pistons are available to fulfil the specification.

Figure 5-21 is a basic EQFD chart showing the functions, means of realising the functions and a simple interaction matrix .

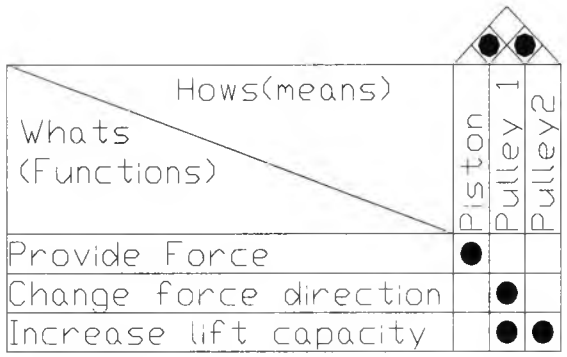


Figure 5-21 2nd Stage EQFD matrix for lifting problem

The ACP diagram (figure 5-22), shows the interaction of forces and displacement caused by the input of air pressure at a particular flow rate.

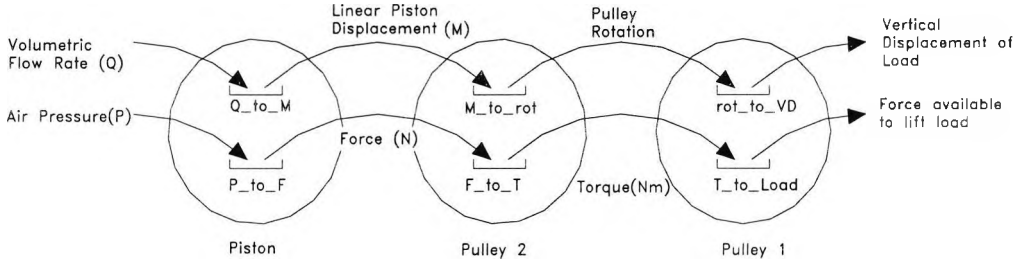


Figure 5-22 ACP model of lifting problem

For the above diagram the formal specification for the piston activity, whose function is to translate air pressure and flow rate into force and displacement, is given below.

ACTIVITY:Piston (INPUTS: Flow rate(Q), Air Pressure(P),
OUTPUTS: Displacement(M), Force(N))

POOL: Q_to_M (INPUTS:Flow rate(Q),
OUPUT: Displacement,
CONSTANTS: Stroke, Diameter)

BEHAVIOUR{Displacement = Q* Time/ Stroke*(Π*Diameter^2/4)
IF Displacement >= Stroke
THEN Displacement = Stroke
ELSE Displacement = Displacement}

POOL:P_to_F, (INPUTS: Pressure,

OUTPUTS: Force,

CONSTANTS: Diameter)

BEHAVIOUR{Force = (Pressure-1)*(π*Diameter^2/4)}

The behaviour for pulley 2 is based upon, the input, the force, and the displacement of the piston. The outputting torque and rotation depend upon the radius of pulley 2 (R2), for pulley 2

rot = Displacement/2* π*R2 (ratio of displacement to pulley circumference) (5.1)

T= F x R2 (from torque = force * distance) (5.2)

Equations 5.1 and 5.2 are integrated into the Pulley 2 Activity to give the following formalisation:-

ACTIVITY:Pulley 2 (INPUTS: Displacement(M),Force(N)
OUTPUTS: rot(θ), Torque(NM))

POOL: M_to_rot (INPUTS:Displacement(M),
OUPUT: rot(θ),
CONSTANTS:R2 (Pulley 2 radius))

BEHAVIOUR{ rot = Displacement/2* π*R2}

POOL: F_to_T (INPUTS:Force(N)
OUPUT: Torque(NM)
CONSTANTS:R2)

BEHAVIOUR{ Torque= Force x R2 }

The behaviour for pulley 1 is based upon the input torque and rotation of pulley 2. the outputting force and displacement, depend upon the radius of pulley 1 (R1),

for pulley 1

F=T/ R2π (5.3)

displacement = rot * 2 * π * R2 (5.4)

Equations 5.3 and 5.4 are integrated into the Pulley 1 Activity, described below.

ACTIVITY:Pulley 1 (INPUTS: rot(θ), Torque(NM))
OUTPUTS: VD(M), Load(N)

POOL: rot_to_VD (INPUTS:rot(θ)
OUPUT: VD (Vertical Displacement)
CONSTANTS:R2 (Pulley 2 radius))

BEHAVIOUR{ rot = Displacement/2* π*R2}

POOL: T_to_Load (INPUTS:Force(N)
OUPUT: Torque(NM)
CONSTANTS:R2)

BEHAVIOUR{ Torque Force x R2 }

5.8.2 The use of VISSIM for component selection

The ACP diagram, figure 5-22 provides the basis for the complete component selection VISSIM model (figure 5-23).

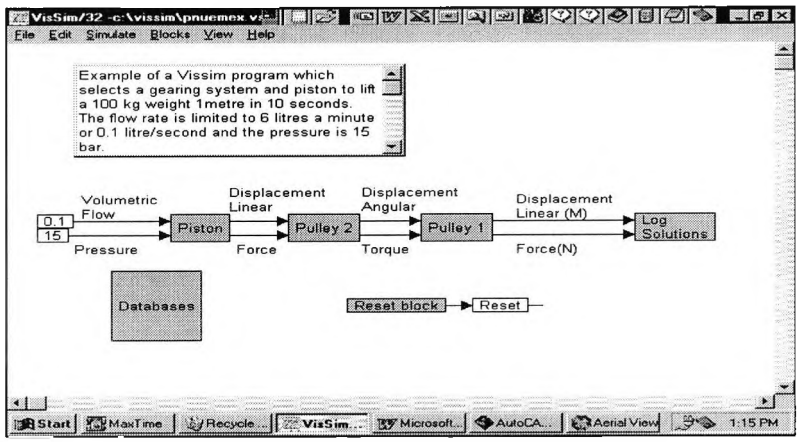


Figure 5-23 VISSIM model for component selection for lifting problem

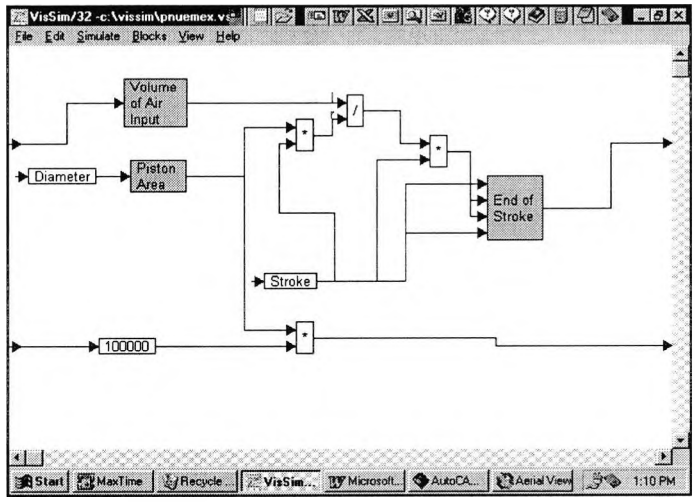


Figure 5-24 Piston Activity

Each block in the diagram represents a particular function. The function of the components described in the ACP specification are identical to the components behaviour in the ACP diagram. Figure 5-24 shows the behaviour of the piston activity (note the variables for the piston diameter and the stroke length).

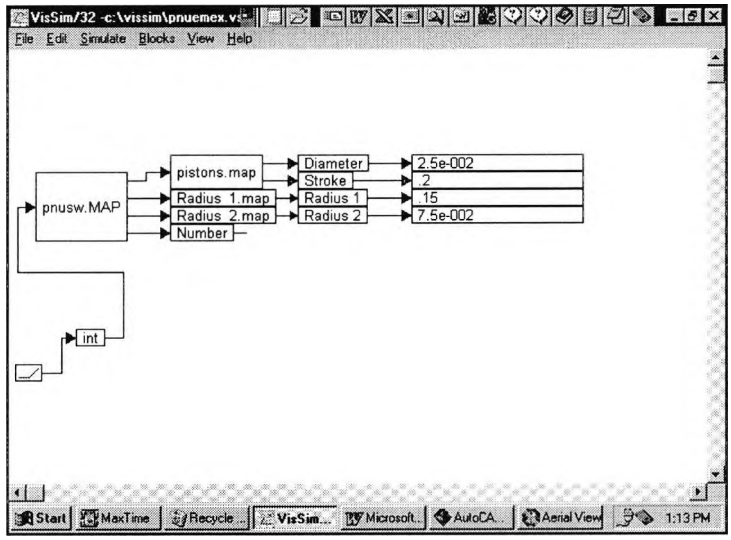


Figure 5-25 Component Database block

The database block (see figure 5-25) is used to contain all the relevant data relating to the components, for instance, the piston database contains an identification number, the pistons diameter and its length of stroke (PISTON.DAT). Similarly, the pulley databases contains identification numbers and pulley radius's (RADIUS_1.DAT and RADIUS_2.DAT). All the combinations of these components are contained in file PNUSW.MAP. Appendix C details the location of these files on the supplementary discs. The ramp is used to systematically select a combination of components, the variables associated with this combination are inputted into the model and the output evaluated.

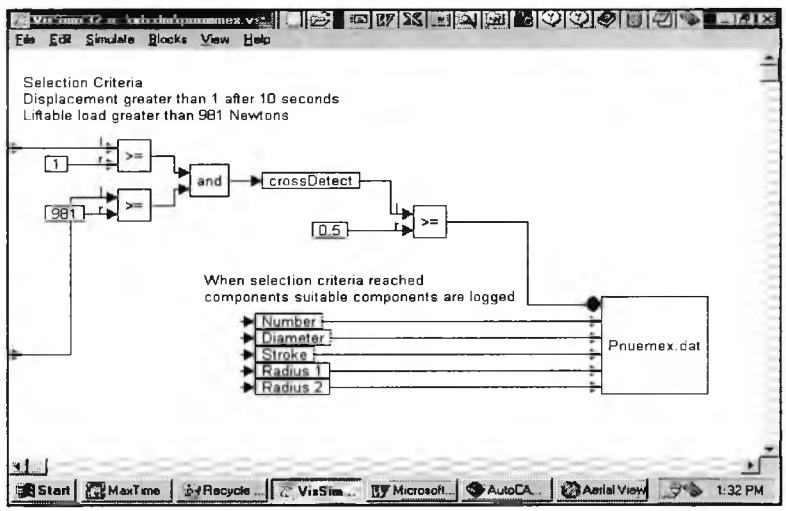


Figure 5-26 Solution logging block

The output for each combination is assessed using the Log Solutions block, see figure 5-26. In this, the output of the displacement and the force available for each

combination of components are evaluated against the specification. If the specification has been matched by a combination, it is logged into the file PNUEMEX.DAT.

The RESET block in figure 5-23 is used to reinitialise the integrator block, which is used to assess the volumetric flow rate in the VOLUME OF AIR INPUT (PISTON block, see figure 5-24), for each combination.

5.8.3 Results

The VISSIM model took 2 hours to create and analysed 64 combinations (4 pistons x 4 pulley 2's x 4 pulley 1's = 64) of components in 50 seconds and produced two solutions, see table 5-2

Solution No	Piston Diameter (M)	Piston Stroke (M)	Pulley 2 radius (M)	Pulley 1 radius (M)	Lifting Force (N)	Possible Displacement of load (M)
1	0.075	0.4	0.05	0.25	1325	1.13
2	0.075	0.4	0.05	0.3	1104	1.36

Table 5-2 Solutions to Lifting Problem

Based on this, solution one was selected as the lifting force produced is greater and the smaller diameter of pulley 1 (0.25) produces less inertia in the system to move. It should be noted that when these solutions were checked by hand it was found that a piston of only 0.2 metres was actually required to achieve the desired lift distance. The reason why a 0.4 metre piston was selected was that a 0.2 metres piston was not entered into the database, if it had of been, a solution incorporating the 0.2 metre piston and the pulley combination of solution would have been found.

In the example shown here, a number of factors were not taken into account, such as efficiency, pulley inertia's, air compression, acceleration criteria, size of the input ports to the piston. All of these factors would have had a significant effect on the analysis of the system and they can be incorporated if required. However this greatly increases the complexity of the analysis. The user should be very aware of creating models that are suitable for the level of analysis required.

5.9 Failure modelling and analysis using VISSIM

Early work on using VISSIM models for failure analysis of mechatronic system revolved around an extension of work detailed in Chamberlain[95]. In this work, a paper based ACP model of the control aspects of the gamma radiography robot was created, the model was very complex and, because it was of a paper based nature, a number of mistakes made. These only became apparent when a VISSIM model was produced. The improvements suggested by the FMEA carried out in Chamberlain[95] were applied to the model and tested ad hoc. Although this produced a safer model, it was time consuming and lacked a structured approach hampering the development of the model. For this reason that it was decided to develop a methodology, which could systematically analyse complex systems automatically and generate a fault tree.

The fault analysis process is based on identifying the failure modes and associated failure behaviours or failure functions of each component using FMEA. Each component can have several modes of failure, but it is possible that these failure modes can be represented by a lesser number of failure behaviours or failure functions. The failure mode behaviour for each component is recreated in the VISSIM model. Combinations of these, are introduced into the software model and the effects observed. The system can be said to fail if a failure is detected and the system is not in a safe state, (the safe state being decided by the failure specification). The system designer should be aware that any safety measure incorporated into the system does not prevent the system from fulfilling the systems basic functionality. For instance, a very blunt knife can be said to be in a safe failure mode, (right side failure), because it cannot cut anyone. However, it also prevents the knife fulfilling its basic functionality of being able to slice vegetables or meat.

Failure combinations are systematically searched to determine the component failure modes. A failure is detected when the behaviour of the perfect model differs from the behaviour of the model with a particular failure function. When a failure is detected, the failure combinations, which contributed to that failure, are logged. These are then analysed to produce the Reduced Fault Tree.

The example used to illustrate this process will be the safety analysis of a hardwired, 'momentary on switch' with a latched relay and a 'momentary off/emergency stop switch.'

5.9.1 System behaviour and modelling

The first stage in the process is to develop an ACP diagram of the system and its VISSIM counterpart. These can be seen in figure 5-27 and 5-28, respectively.

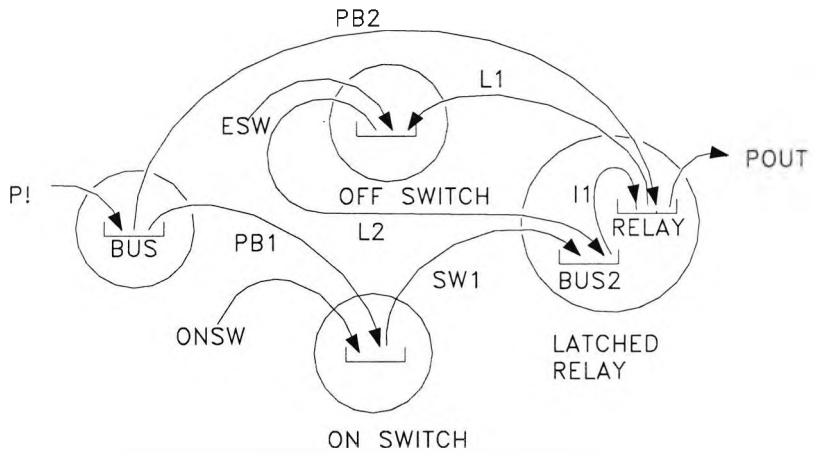


Figure 5-27 VISSIM model for Latched On, Emergency Stop Off switch

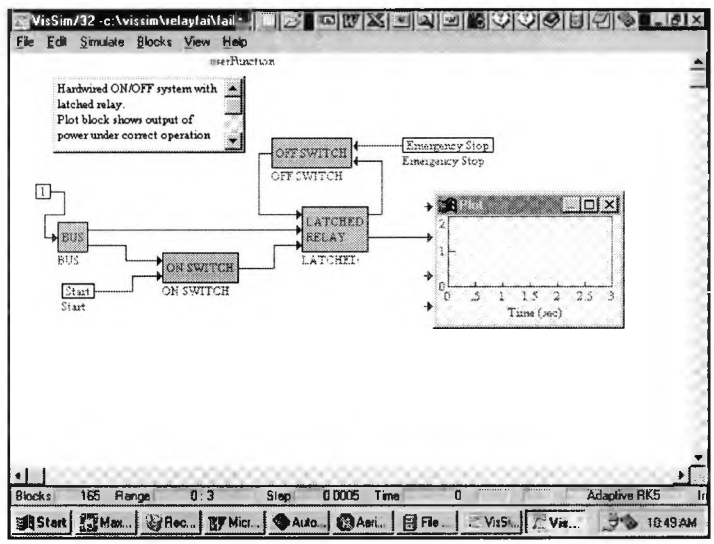


Figure 5-28 ACP diagram for Latched On, Emergecy Stop Off, switch

At the most basic level, power is output from the system when the Start button is pressed momentarily and turned off when the Emergency Stop is pressed momentarily. The behaviour is modelled using four elements (or Activities), a BUS, ON SWITCH, OFF SWITCH and a LATCHED RELAY. The BUS Activity is used to split the power into two lines, PB1 and PB2. The power from PB1 is passed to the ON SWITCH Activty, thus allowing the power to pass through to SW1 when the Start button is high. As soon as the Start button is sent low the power is turned off (the switch is

momentary). The power sent through to SW1 passes through to the LATCHED RELAY Activity and the Pool BUS2, which is used to take a signal from either SW1 or the latch line L2. The output from BUS2, L1 is used to activate the LATCHED RELAY.

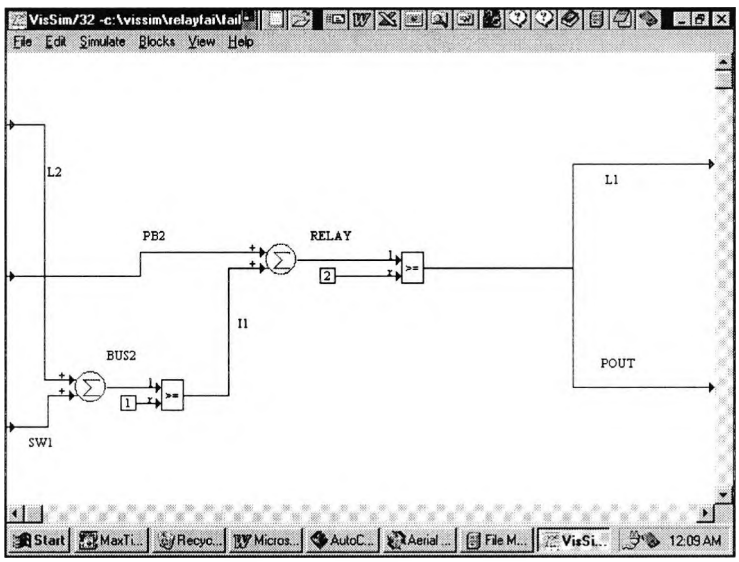


Figure 5-29 Latched Relay Activity

The behaviour of the LATCHED RELAY, see figure (5-29), is more complicated. It is modelled using the size of the signal and boolean logic. When SW1is high, the power from PB2 is used to latch the LATCHED RELAY closed via LI, ESW and L2. The signal from PB2, is also an output to the real world via POUT. L1 provides the first stage of the relay latching process, this is achieved by sending the signal to a normally closed switch, OFF SWITCH, which when closed, the signal is passed directly to L2 and BUS2, thus latching the relay. When ESW is high the contact is broken and the interlock also broken and power ceases to pass to the outside world via POUT.

5.9.2 Failure Mode identification and VISSIM modelling

Once the system components have been decided upon, a FMEA for the system components, see table 5-3 (on the following page), is carried out. The next stage is to model the behaviour described for each failure mode in each component in the VISSIM model and set up a state selector. This selects the behavioural mode of the component and can be seen in figure 5-30, which shows the VISSIM model for the ON SWITCH activity with the failure behaviour introduced.

The remaining block in this section is the Signal Comparator and Component Failure Monitor. This compares the signal produced by the perfect behaviour of the component with the signal produced by the failure behaviour, when these behaviours differ the output from the block is sent low, to indicate that a component failure has occurred. This signal is then passed to other components directly connected to the first component.

Component	Failure Mode	Cause	Behaviour Mode
BUS	OK (no fail)		continuous signal on P1 and P2
	P1 open circuit	disconnection of P1	no signal on P1
	P2 open circuit	disconnection of P2	no signal on P2
	P1 & P2 open circuit	disconnection of P1 & 2	no signal on P1 & P2
ON SWITCH (normally open)	OK (no fail)		Power out when switch is pressed, power off when switch is released.
	open circuit	rocker failure in switch	no power out
	short circuit	switch contacts fuse	continuous power out
		object in switch	continuous power out
	momentary action fail	broken spring	continuous power out after the switch has been pressed
OFF SWITCH (normally closed)	OK (no fail)		Power off when switch is pressed, power out when switch is released
	open circuit	contact disconnection	no power out
	short circuit	switch contacts fuse close	continuous power out
	momentary action fail	broken spring	no power out after switch has been pressed
LATCHED RELAY	OK		If (SW1 or L2) and PB2 high then L1 and POUT high
	Latch disconnected	Broken wire	If SW1 and PB2 high then L1 and POUT high
	Short Circuit	Relay fused closed	If SW1 high then L1 and POUT high
	Open Circuit	Relay broken	POUT and L1 low
		Outputs disconnected	

Table 5-3 FMEA table for hardwired on/off switch with latched relay

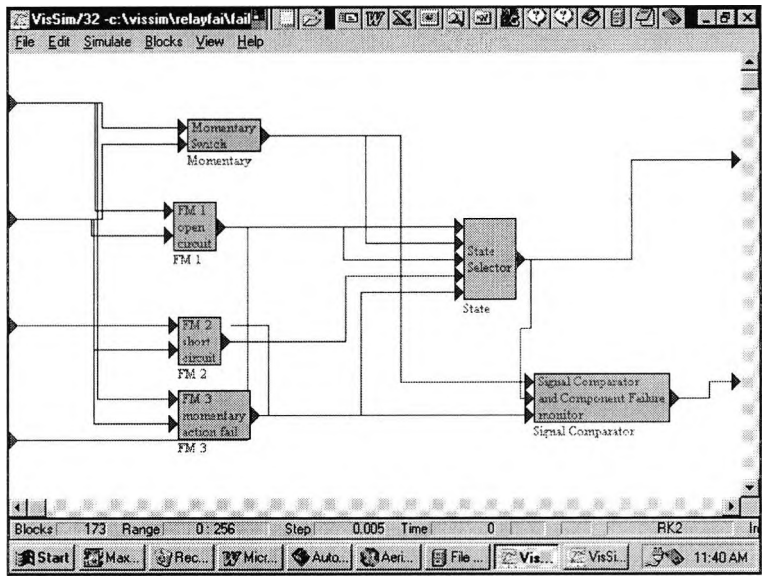


Figure 5-30 Failure model for ON SWITCH Activity

This process is necessary because the system can sit in a failed state with the behaviour of the failure effect not realised until the component is activated. For example a user would not be aware of a switch failure until the user tries to turn the switch on and no power is output to the intended device. It is also necessary to indicate when a failure has occurred so that a measurement of the system output can be made and compared with the failure criteria.

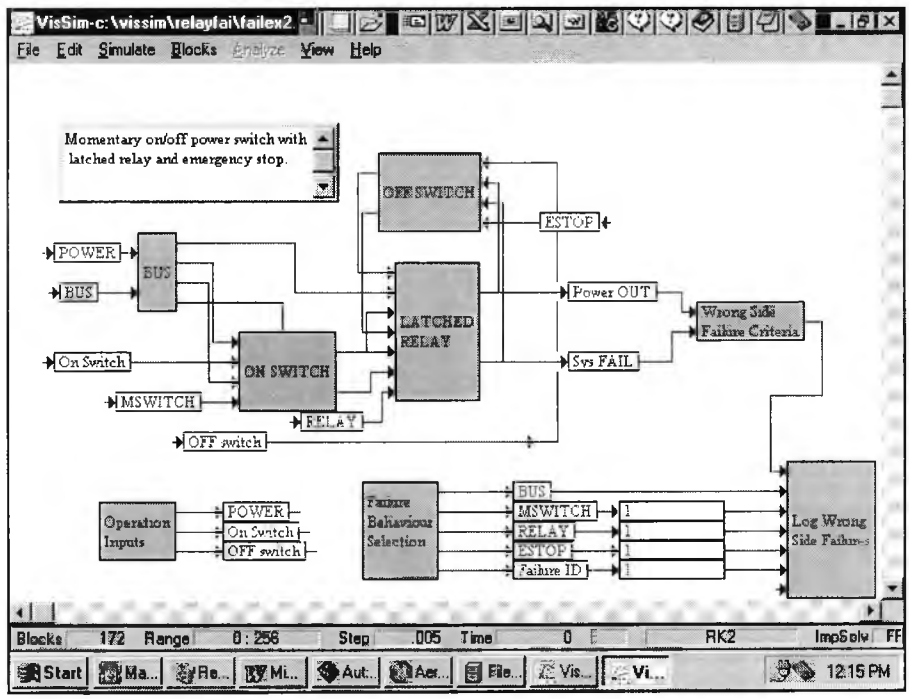


Figure 5-31 Complete VISSIM model for failure simulation and analysis

Once the component failure behaviours are embedded in the VISSIM model, a number of support blocks need to be created. These provide information relating to the combinations of failures, the sequence of operations of the various switches, the failure criteria and a log of the combinations of failure modes, which cause system failure. The complete VISSIM model can be seen in figure 5-31.

The Operation Inputs block contains a database of the sequence of operations for each switch and the power into the system. This sequence starts by having no POWER. POWER is then sent high, the On switch is activated and then the OFF switch activated. This sequence lasts one second, and is initiated in 0.2 second steps. This sequence is applied to all failure mode combinations.

The Failure Combinations block contains a database of all the failure combinations, in this case there are 256 failure combinations. This number is derived from the multiplication of the number of component behaviours for each component, in this instance this is 4 * 4 * 4 * 4 = 256, the virtual simulation time therefore 256 seconds (1 second/sequence of operations * 256). Each of these failure combinations are systematically applied to the model using the State Select blocks in each component block.

A Wrong Side Failure (defined as when the system fails in a unsafe manner) is found when the Power Out from the system is high when a failure has occurred. When this condition is met, the failure combinations that caused the condition are logged in the Log Wrong Side Failures block.

5.9.3 Results and Analysis

The output file of logged failure mode combinations, once produced, is then examined, in this example, 42 failure modes were logged. A section from this file is shown in table 5-4.

The first column indicates the number of the failure combinations in the failure combination file, the second relates to component 1, in this instance the BUS, column 3 ON SWITCH, column 4 LATCHED RELAY and column 5 the OFF SWITCH. The numbers in each column relate to the failure mode for the component.

#I=0,256,0.005; c:\vissim\relayfai\failex2.vsm @ Mon Apr 26 22:25:03 1999				
9	0	2	0	0
13	0	3	0	0
25	0	2	1	0
33	0	0	2	0
34	1	0	2	0
37	0	1	2	0
38	1	1	2	0
41	0	2	2	0
42	1	2	2	0
45	0	3	2	0
46	1	3	2	0
73	0	2	0	1
89	0	2	1	1

Table 5-4 Logged failure combinations for latched, ON/OFF switch

The analysis of this information is initiated by identifying all Single Point Failures (SPF), any failure combination which contains a SPF is removed from the analysis. This is based upon the logic that if a SPF is present in another failure combination, system failure would be caused by the SPF instead of the other failure modes. This reduces the number of failure combinations from an unmanageable 42 (this leads to very complex fault tree being produced, which then has to be analysed further to be understandable) to 4 SPF see table 5-5 and gives rise to the Reduced Fault Tree, shown in figure 5-32.

#I=0,256,0.005; c:\vissim\relayfai\failex2.vsm @ Mon Apr 26 18:14:16 1999				
9	0	2	0	0
13	0	3	0	0
33	0	0	2	0
129	0	0	0	2

Table 5-5 SPF failures for ON/OFF switch

These failure designations correspond to:-

Comp2 Failure Mode 2 :- ON SWITCH, short circuit

Comp2 Failure Mode 3:- ON SWITCH, momentary action failure, broken spring

Comp3 Failure Mode 2:- OFF SWITCH, short circuit

Comp4 Failure Mode 2:- LATCHED RELAY, short circuit

In this case the FMEA, VISSIM failure model, associated files and analysis, took approximately 8 hours. The virtual simulation time of 256 seconds takes approximately 100 seconds to run on a 100 MHz 486 computer.

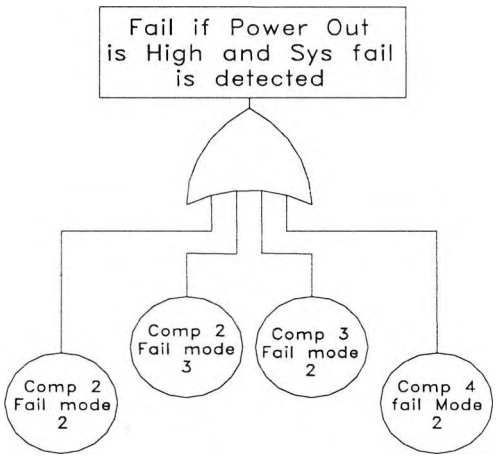


Figure 5-32 Reduced Fault Tree ON/OFF switch

Once the failure modes have been identified, it is possible for the design engineer to generate solutions in a structured manner. This is achieved by examining each failure combination separately, identifying the causes of failure, generating and evaluating solutions, using EQFD and the weighted objectives method. Once solutions have been selected, the VISSIM failure model is updated with the new components and the safety design cycle repeated from the component FMEA stage.

5.10 Summary

This chapter has given an overview of the methodology employed, covering the initial failure analysis using CLASH, the first three stages of EQFD, going from customer requirements through to concept selection. Once the concept is selected, component selection is carried out automatically using a VISSIM model derived from the EQFD, correlation chain, interaction matrix and ACP model. This model provides the basis for

failure analysis, which involves FMEA, from which a VISSIM failure model is generated, the results being a RFT. The information gained from the failure analysis is used to provide input to the next iteration of the design. Each stage of the methodology can exist by itself, as seen by the variety of the examples shown, or can be combined and employed through out one project.

The next four chapters show the application of the above methods to a single project, in this instance the safe design of 2 axes of the NDT Gamma Radiography Robot detailed in Chamberlain[95]. Chapter 6 defines the system specification, Chapter 7, demonstrates the use of HAZOP/CLASH combined with the techniques and tools employed in EQFD. The result of this chapter will be the conceptual design of the project. Chapter 8 develops the ACP diagrams from the correlation chain and the VISSIM model for component selection. The results of this analysis are compared with the results generated by hand. Chapter 9 extends the VISSIM model further by integrating supporting components into the design and carrying out a VISSIM failure model analysis and redesign.

6 Case study, justification, specification and scope of work

The objective of this chapter is to describe the case study work. This involves justifying the project, outlining the task to be performed, defining the task specification, and finally, the scope of the case study. In the last section, the contents of chapters 7, 8 and 9, describing the application of the various design tools and methods to the case study, is introduced.

6.1 Case study justification

The objective, as stated in the original research proposal was to determine the most effective means for remote deployment of radioactive sources in on-site gamma-radiography of storage tanks (Chamberlain[95]). Such work is currently carried out using a semi-automatic device, which is fastened to a temporary structure such as scaffolding. When the device is in place, a radioactive source is exposed to the air using a manual or semi-automatic winder for a predetermined length of time. When the exposure time is complete, the radioactive source is retracted.

Present methods are inadequate for two reasons (Chamberlain[95]). In the first instance, they are inefficient, given the predicted rise in the number of storage tanks which require testing in the near future (see section 1.1), and the current speed of operation (at present 15-20 exposures a day are the norm). Secondly, they are unsafe, this is due to the improvised nature of the methods used to hold the source collimator and source projector unit (see section 6.2) in place. Methods observed have included use of G-clamps and adhesive tape. Therefore, to improve productivity and safety, by removing the human element as much as possible, it was proposed to develop an automated tool to carry out the gamma radiography task.

The Gamma Radiography task can be dealt with in two parts. The first part deals with the development of a safe control mechanism for Source exposure. This is detailed in Chamberlain[95], Chamberlain[96], Edney[96] and provided the author with the experience of applying various tools and methods to a practical design problem, and led directly to the development of the structured methodology and VISSIM based tools detailed in sections 5.3 to 5.9.

The second part deals with the design and safety analysis of the source positioning mechanism and its control system. This provides the case study that demonstrates the application of the combined methods and tools described in sections 5.3 to 5.9.

6.2 Task description and system specification

It was felt, for safety reasons, that existing gamma-radiography equipment should be adapted for use in the automation solution. This is because a lot of portable equipment already exists, which has been tried and tested by the gamma-radiography operators. The use of this equipment has a direct bearing upon the automation solution specification, as it defines the payload of the system , the handling requirements and allows basic problems with the operating procedure to be eliminated.

The equipment used for gamma radiography comprises the following components:-

- Source:- Shaped piece of radioactive material such as IR-192. Located on the end of a wire know as the source holder or Pigtail.
- Source Projector Unit:- A device made from depleted uranium that contains the source and shields users from dangerous levels of radioactivity, between exposures. It also incorporates various safety features to prevent premature source exposure.
- Guide Tube and Collimator :- The guide tube (variable between 0.3 to 20 Metres) , is attached at one end, to the source projector unit and the other to the collimator. It guides the source holder to the target. The collimator directs the focus of the gamma rays to the appropriate spot.
- Drive guide tube:- Connected between an automatic winder system or manual winder system and the source projector unit. Provides the connection that enables the source to be exposed from a distance, normally within 20 Metres. This distance is required to limit exposure of the operators to excessive radioactivity.
- Control panel and automatic winder system:- Operator set automatic control system which controls source exposure times and wind out/wind in of the source. Can be integrated with safety features such as a dose rate meter and intruder alarms.
- Dose rate meter:- A device to measure levels of radiation. If the dose level exceeds the pre-set level, an audible warning and light alarm is given.

The sizes and masses of the important pieces of equipment are given in table 6-1.

Component	Length (mm)	Width (mm)	Height (mm)	Mass (Kg)	Primary Power Source
Source projector unit	240	110	120	16	None
Automatic winder system	600	355	220	16	115/230 V AC. / 12 V DC 6 AH
Dose rate warning system.	250	105	230	3.2	12 V DC, 6 AH
Collimator	200	100	100	5	None
Guide Tube	300mm- 20 M	25	N/A	Variable	None
Drive Guide Tube	20 M	25	N/A	20	None
Batteries	400	250	300	10	12 V DC, 6 AH

Table 6-1 Gamma radiography equipment specification.

Two problems have been identified with the present procedure. Firstly, the long exposure times caused by long wind out/wind in period and, secondly jamming of the wire guide mechanism. Both of these problems are caused by the length of the drive guide wire and the guide wire. In many instances, guide tubes of 20 metres have been used, where the time taken for the source to reach the collimator is 32.7 seconds (using an automatic winder system). In an automated solution, with a guide tube of 200-300 mm, the one way projection time can be reduced to 400-500 milliseconds. When combined with exposure times of 1 to 5 minutes, the use of an automated handling system could lead to 20% to 50% reduction in total exposure time (wind out time + exposure time + wind in time). The use of shorter guide tubes (limited to 300 mm) and drive guide tubes (limited to 2 Metres) would also prevent the jamming of the source in the guide wire, as there is less friction and chance of twisting the guide tubes. If this approach is taken, the automatic winder unit, source projector unit, and collimator plus the associated guides tubes and power sources, all need to be located close together, on the automated source position system. For safety reasons, the dose rate warning system meter must also be located on the source positioning system. This gives a general payload requirement for the source positioning system of approximately 100 Kg. The positioning requirement for gamma radiography is for a 5 Kg collimator to be located 200-400 mm from the surface, with an accuracy of +/- 2 mm & +/- 1 degree direction. Any solution must also be able to position the source quickly and accurately over an area of approximately 1 metre square.

Although the primary function of the system under consideration is for automated gamma radiography, the system should also be capable of carrying out secondary non-destructive testing (NDT) tasks, such as Magnetic particle inspection, dye penetration, and ultrasound. The ability to perform these tasks improves the economic viability of employing an automated source positioning system, and turns it into an automated NDT positioning system.

6.3 Safety Specification

Because of the potentially hazardous nature of gamma radiography and the use of an automated solution, a number of safety criteria have to be determined. These safety criterion reflect safety in two areas. Firstly, the safety of the operator with respect to being injured by the source positioning system, and secondly, the safety of the control system of the source positioning system, with respect to the source and preventing its movement when a failure has been detected..

Safety of the automation device relating to the operator is the prevention of the device harming the operator when manipulating the source positioning system or maintenance is being carried out. This is dependant upon the configuration of the source positioning mechanism finally used. However, guidelines can be made once a general concept has been decided upon.

The safety criterion for the control system is based upon two criteria:-

- That the position of the collimator (and hence the source once projected), be known to within +/- 1 mm within 0.065 seconds of a failure occurring.
- That the velocity of the collimator be 0.0001Metres/second within 0.065 seconds of a failure occurring (i.e as near as possible to zero velocity that VISSIM can model).

A time limit is part of the safety criteria because the system is dependent upon the control system detecting a failure and acting upon it. It is impossible for the system to react instantaneously. The author, with agreement of his project supervisor, agreed that a time of 0.065 seconds was a reasonable response time to react to a system failure. If either of the above criteria is not met the system has failed. However, if a failure does occur and the above criteria are met, the system can be said to have failed safe, i.e. in a condition identified as safe by the system designer. This is also know as a right-side

failure. A wrong-side failure occurs when the system fails in an unsafe condition, as defined by the system designer.

Safety analysis of the control system is very dependent upon the specific components used and the software developed to identify and react to failures. Because of this, the safety analysis of the control system can only be fully carried out late in the design process.

6.4 Scope of the case study

So far this chapter has identified the basic safety, positioning and payload specification for the source positioning device. These requirements provide the initial inputs into the design process.

The first stage of the design process is covered in chapter 7. A QFD analysis of possible solutions to the gamma radiography problem is carried out, this used to justify the selection of a basic concept based upon the specification given in section 6.2 and taking into account general customer requirements derived from gamma radiography operators. Once a general concept has been decided upon, a HAZOP/CLASH analysis will be carried out, providing further customer requirements. The first two stages of the EQFD process will then be applied to the design of the source positioning system, thus covering the detailed concept design of the drive mechanisms.

Chapter 8 deals with the embodiment design of the drive mechanisms, employing ACP diagrams to lay out the top level system from which a VISSIM model is developed. The VISSIM model is then adapted so that automated component selection can take place to determine specific components.

Chapter 9 deals with the failure analysis for the source positioning system. This process starts by developing a sophisticated VISSIM model from which automated FTA can be carried out. The FTA results are used to guide the redesign of the system for safety using EQFD. The VISSIM model is updated with the redesigned software and components to check that the required safety level has been achieved.

6.5 Summary

This chapter has described the components used for gamma radiography and how they affect the specification of the source positioning system. The basic specification is for a device that can :-

- Carry a payload of 100 Kg
- Position a 5 kg Collimator to within +/- 2mm and +/- 1degree of the required position.
- Position the Collimator 200mm to 400 mm from the test surface.
- Carry out secondary NDT tasks such as magnetic particle inspection, ultrasound and dye penetration.
- Ensure the source position system cannot harm the operator whilst being located or being maintained.
- Ensure that if a fault is detected or occurs the velocity of the source is 0.0001 metres 0.065 seconds after the fault occurs.
- Ensure that if a fault is detected or occurs the position of the source is known to within +/- 1 mm , 0.065 seconds after the fault occurs.

Finally, the scope of the case study is discussed along with contents of the chapters 7, 8 and 9. Chapter 7 deals with initial safety analysis, concept design and details design. Chapter 8, develops a VISSIM model for components selection and uses it to select specific components for the drive mechanism. Chapter 9 completes the design process by carrying out a computerised failure analysis of the source positioning system the results of which are used to refine the design and improve its safety.

7 Defining Customer Requirements and selecting concepts: CLASH led EQFD

7.1 Introduction

The objective of this chapter is to demonstrate the use of CLASH led EQFD up to and including detailed concept selection. This will be achieved by applying the methods and tools detailed in chapter 5 sections 3 through 6. Fulfilling one of the original objectives of the study.

The process starts by developing the case study brief and defining the initial customer requirements. The initial requirements drive the development of the basic concept selection, which is achieved by employing QFD to compare various initial conceptual solutions and computer simulation to finalise the concept. Once the basic concept has been decided upon, implicit safety based requirements are defined and a CLASH analysis can be carried out, leading to an expansion of the initial customer requirements.

The next stage involves the development of the detailed conceptual design within the framework of EQFD. This takes the explicit and implicit customer requirements as the starting point for the design process, with inputs from CLASH to deal with the basic safety issues. The first two stages of EQFD are applied to the design process. This covers functional analysis, concept generation, function means trees, morphological charts and the weighted objectives method for detailed concept selection. The law of vertical causality is employed to control the design process by ensuring that no sub-functions to a function are derived before a solution is found to the high level function.

7.1 The use of QFD for initial concept selection

The first stage in the QFD analysis is to determine the basic requirements for the source positioning system. These are described in sections 6.2 and are summarised as: -

- Carry a payload of 100 Kg
- Position a 5-kg Collimator to within +/- 2mm and +/- 1degree of the required position.

- Position the Collimator 200mm to 400 mm from the test surface.

The source positioning system can be considered to consist of two parts, a working platform which supports the automatic winder unit, source projection unit, collimator, dose rate meter, guide tubes and associated power sources in the general vicinity of the test location. And, secondly, an accurate positioning system which should rapidly locate the collimator to within a specified tolerance. In conventional systems, this function is realised by human involvement, whilst in automated solutions it is generally realised by use of some form of robotic arm.

7.1.1 Selection of Working Platform Concept

Considering both conventional and automated/robotic approaches to the working platform problem, there are numerous possibilities. The key factors which affect the final choice are, safety, range and reach, stability, payload, reliability, productivity, cost, ease of use, training, and evacuation. The last factor, evacuation, refers to the ease of use and speed with which personnel can escape to a safety zone.

The solutions for the working platform can be spilt into three basic types, conventional, robotic climbing robots and robotic non-climbing robots. Conventional systems are currently in use in a wide variety of applications. The robotic, climbing and non-climbing solutions are mostly experimental prototypes rather than commercially available systems.

7.1.1.1 Conventional platforms

Conventional platforms will be classified under the following headings:-

- Scaffolding:- This covers all static access equipment. Fairly expensive to hire and install, but it becomes more economically viable when renovation and repair is also being carried out. Productivity is low, as is the ease of evacuation. The stability of scaffolding is generally high but can be variable depending upon how well the scaffolding has been erected.
- Booms:- Comprises all types of self powered telescopic booms and long reach articulated arms. They can be employed where site access is possible. The end of the boom is not very stable due to the boom length, and operating and ownership costs are high, also, safety is a problem, with the

possibility of the equipment carrier colliding with the tank. However, productivity improves due to the rapid positioning of the equipment.

- Suspended cradles:- Two types of cradles exist, these being permanent and temporary systems. They have quite high payloads, but poor stability, unless attached to the work surface by some means. Evacuation is poor, but productivity is reasonably good, due to rapid positioning. They are cheap to operate and many portable systems exist.
- Lifts: - cover mobile plant such as mast climbers, scissors lifts and vertical telescopes. These have much the same profile as suspended cradles, however, they are more expensive to operate.

All of the above solutions are very reliable, can carry the 100 Kg plus payload and have the necessary range and reach.

7.1.1.2 Climbing robots

Climbing robots use a number of magnets or vacuum suckers to attach themselves to the working surface. Propulsion over the surface is achieved by maintaining the attachment of some of the contact points whilst moving the remaining unattached points in the required direction.

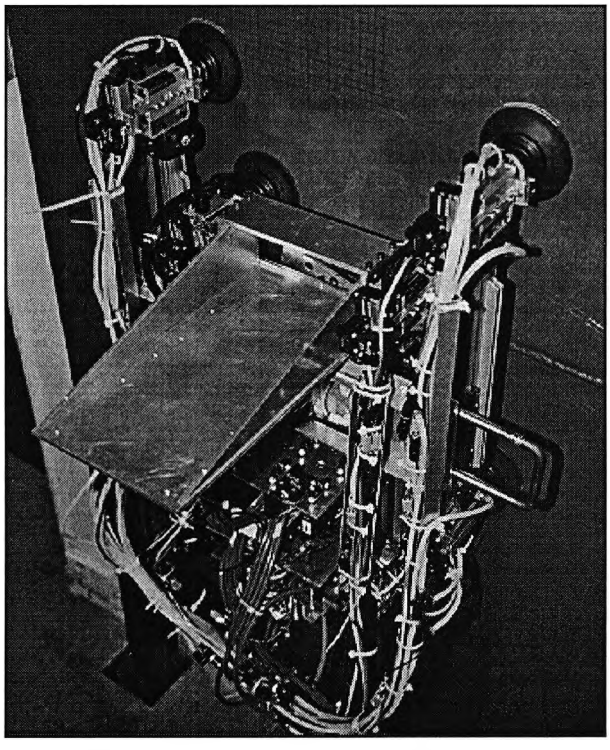


Figure 7-1 South Bank climbing robot

Figure 7-1 shows a climbing robot with suction feet (developed at South Bank University), the plate at right angles to the wall provides a platform for a small 6 axis robot. Figure 7-2 shows a crawler type robot (Carra[92]) with a set of magnets built into the circumference of the robot wheels. Movement is achieved by alternating the power to each magnet and the small pistons, which are attached to a crawling mechanism and wheels.

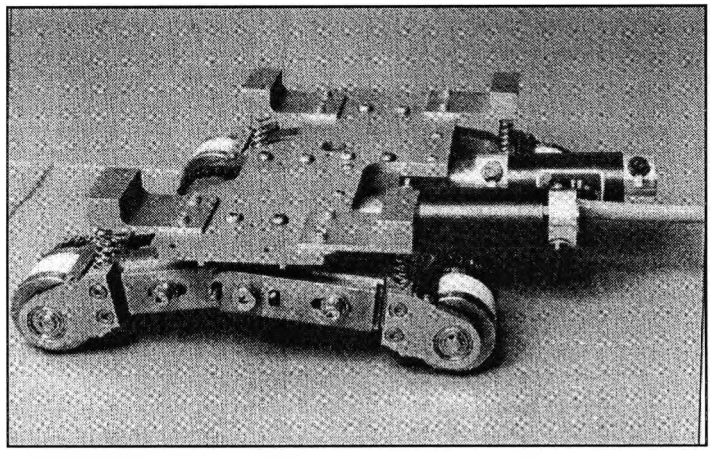


Figure 7-2 Magnetic crawler robot

Figure 7-3 shows one of the most well known of pneumatic walking/climbing robots, Robug 2, (Collie[90], Bevan[93]), developed at Portsmouth University. Whilst the legged robot can often operate on general terrain the others normally require flat or low curvature surfaces. Other problems associated with the robots mentioned above are slow movement and low payload, the payload of Robug 2, for example, is 25 Kg.

In the case of the robots attached by vacuum, this is produced either directly via a vacuum pump or indirectly through a venturi. In both cases it is necessary to have on board air-supply in order not to limit the range of the platform. For ranges in excess of 8-10 metres, the weight of the umbilical and the flow loss though the supply line are unworkable. Because the vacuum cups used have a degree of elastic compliance they can mould themselves over irregularities in the surface. This feature is useful given the possibly rough surfaces of the storage tanks, however the presence of elastic compliance can cause problems with accurate location.

Magnets do not have the compliance problem, however, complete and close contact with the target surface is essential. The effectiveness of magnets is further diminished by, the presence of corrosion and thick layers of paint. Whatever attachment method is employed, it must provide a factor of safety to overcome gravitational shear forces,

wind loading, motion induced forces, and turning moments. None of the robot platforms presented here are capable of moving a payload of 100 Kgs.

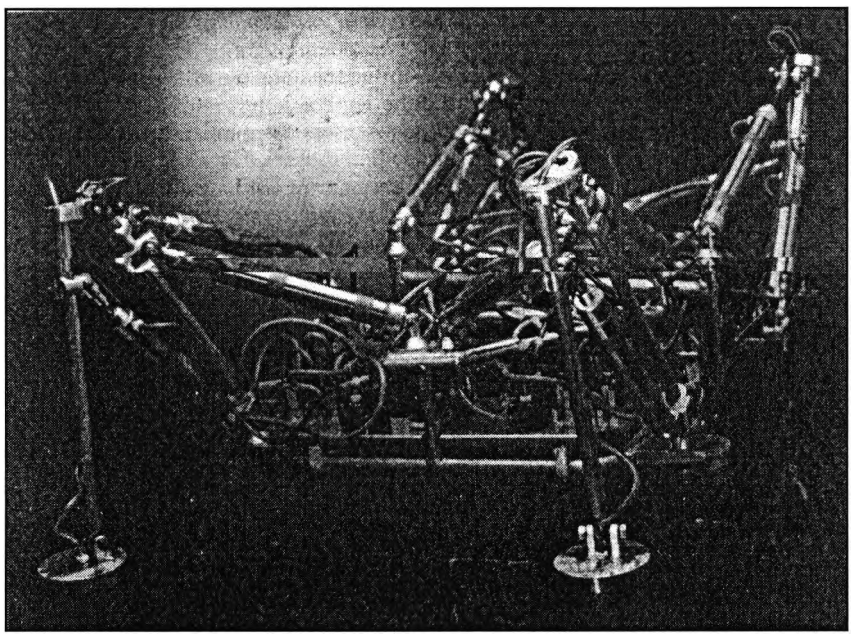


Figure 7-3 Robug II (Collie[90])

Considering the overall safety issues, the operation of climbing robots on vertical surfaces, whether relying on vacuum cups or magnets, must be considered extremely hazardous. This observation inevitably leads to the addition of an independent cable support system, which continuously pays out or gathers cable according to the robots motion.

7.1.1.3 Non- Climbing Robots

Non climbing robots are those robots that do not depend upon surface contact for robot motion over the work surface. Boom mounted robots can be considered but are impractical due to the reasons given in the section on telescopic booms. Also they do not posses the high precision actuation necessary to position the radioactive source accurately.

Suspended access robots have been used in a number of applications, including, wall-painting (Gokyu[96], Miyajima[96]), window cleaning (Farsi[94]), and wall inspection (Wing[89]). The idea is to suspend a piece of specialised equipment from cables so that it hangs on the side of the structure. Using winches the vertical position of the robot can be easily altered. The horizontal position also can be changed by moving the winches

along a track located at the top of the structure. A system such as this would provide very accurate and rapid location of the gamma-radiography equipment. The operating costs of such equipment are fairly low. The main problem associated with this method is the stability of the working platform, which must be held still whilst an exposure is taking place. This can easily be achieved using vacuum suckers or magnets. Because cables already support the platform, very little effort is required to hold the platform in place. One of the simplest solutions is to use magnets with sliding keepers, this is because no vibration would be induced whilst a source exposure is taking place. This would not be the case if vacuum suckers with an on board compressor were used.

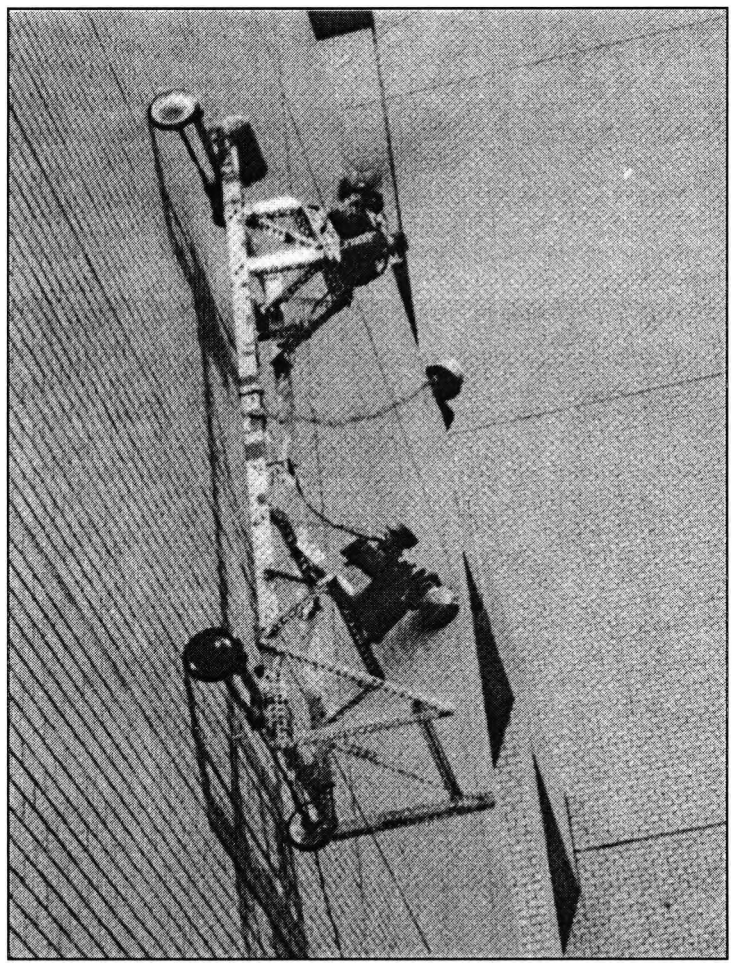


Figure 7-4 Flying robot (Nishi[93])

Flying robots, see figure 7-4, (Nishi[93]), have been included for completeness. However their low payload, operator control difficulties and danger inherent to flying machines make them unsuitable for radiographic survey work.

7.1.1.4 QFD Assessment of Platforms

Using the key factors given in section 7.2, table 7-1 sets out a comparison of the relative performance of conventional and robotic working platforms. The scores are given on the basis of 0,3,6,9 in order of increasing merit and the totals are based on equal weightings.

Competing Concepts	Conventional				Robotic					
					Climbing			Non-Climbing		
Customer Requirements	**Scaff-olding**	**Boom**	**Lift**	**Cradle**	**Walk-ing**	**Climb -ing**	**Crawler**	**On Boom**	**Suspended**	**Flying**
Safety	9	3	6	6	6	6	6	6	9	0
Stability	6	3	6	6	6	3	6	3	6	0
Range - Reach	9	9	6	9	3	3	3	9	9	9
Payload	9	9	6	9	3	3	3	9	9	0
Reliability	9	9	6	9	0	3	3	6	9	3
Productivity	0†	9	6	6	3	3	3	6	6	3
Costs	3	3	3	9	0	3	3	3	6	3
Ease of Use	9	3	6	6	0	3	0	3	6	0
Training	9	6	6	6	0	0	0	3	3	0
Evacuation	6	3	3	3	9	9	9	9	9	3
TOTALS	69	57	63	69	30	36	36	57	72	21

† includes erection time

Table 7-1QFD analysis of working platforms for remote Gamma Rradiography

The assessment of conventional platforms, which is based on the opinions of field workers, is somewhat speculative on account of their lack of experience with methods other than scaffolding. The conventional methods considered all score low on evacuation as they depend upon personnel being located nearby to set them up accurately. In the case of the lift, boom and cradle, that person would have problems in removing himself from the danger area, because of the remoteness of the location

device from any other piece of equipment. Scaffolding scores slightly better as an escape route is available to personnel. Of the conventional methods considered, the use of the scaffold and cradle are considered to be the best solution for the working platform.

The robotic methods considered suffer from a number of general problems, these are:-

- Payload: -Limited by the suction or magnetic forces available to withstand gravity, wind forces, self-load, and moments.
- Low productivity: - The speed of movement of many climbing robots is very slow. This is partly due to time taken to ensure that the robot is attached to the wall sufficiently well before the next move is made and the slow speed of physical movement of legs, frame or whatever means is employed.
- Safety: - To ensure reasonable safety of the climbing robot and personnel a support line needs to be attached to the robot, which follows the robots movement and can prevent the robot hitting the ground if it detaches from the tank.

Of the non-climbing variety of robot, the suspended cable type robot is the most successful this is because it is relatively stable, if attached using magnets or vacuum suckers at the location point. In this instance the attachment mechanism provides positioning stability and very little support, all the support for the robot taken up by the cable. This also allows the robot to have a high payload. Productivity would be very high as gross vertical and horizontal motion can easily be realised, the speed of local positioning being dependant upon the robot configuration employed. Because the working platform is the base for a robot, the scores for evacuation and safety are also high, the source can be positioned remotely without any operator being present. With the boom-mounted robot, issues of safety are important due to the possibility of colliding with the tank. Other problems associated with boom robots are site access, accurate gross positioning of the robot, high costs and stability, unless an attachment mechanism is included in the design.

From the above analysis, it can be concluded that a non-climbing robot suspended from cables provides a good solution to the working platform problem. This is because it is a hybrid of the best features of all the concepts considered. It combines safety, stability and high payload of the cradle with the remote operation of a climbing robot, without the wall attachment problems associated with a climbing robot.

7.1.2 Selection of robot configuration for collimator handling

The next stage of the analysis was to determine the type of robot to be located on the working platform. The type of robot has a direct bearing upon productivity and its ability to carry out the task. Three factors need to be taken into account, the joint accelerations, joint velocities, and the dextrous working envelope. The joint accelerations and velocities affect how the fast the robot can move the end effector or collimator from one position to another, whilst the dextrous working envelope deals with the robots ability to orientate the collimator in the correct direction at a particular point in space. This is due to the limited rotation provided at the robot joints and its proximity to the working surface. Whilst there are a number of classes of robots, only a conventional, serial jointed, robot arm and a cartesian robot have been considered in this study.

Computer simulation has been used to investigate the performances of a PUMA type arm and a comparable, conceptual cartesian robot. The choice of the PUMA was influenced by its adoption in previous NDT automation trials at South Bank University, London. Whilst a cartesian type robot was selected because its dextrous working envelope is not as limited as the PUMA robot due to its simpler construction, and the actual frame of the cartesian robot acts as the working platform.

GRASP (BYG ltd. Nottingham) robot simulation software has been used to investigate the performance of the two robots in NDT. It allows three-dimensional models to be created of the work surface and the robot. Inverse kinematics, which model the relationships between the endpoint of the end effector and the joint motor positions, are combined with the joint specifications, such as acceleration, velocity and limits of travel, to determine the time taken for a specific series of moves to be carried out.

7.1.2.1 Task Models for Gamma Radiography Robot

Because the functionality of the robot was not only intended for Gamma radiography, but to have secondary NDT tasks to carry out, such as, magnetic particle inspection and dye penetration (see section 6.5), two different tasks were created for the robots to carry out. Both tasks reflect operations, which would be carried out if the NDT methods mentioned above were automated.

- Task 1: - Surface contact to be maintained on three parallel lines, 250 mm apart and 250 mm long. This represents a typical scanning operation for the

location of flaws in dye penetrate or magnetic particle inspection. In gamma radiography, such a task would be used to find Image Locators, small radioactive sources that are located inside the storage tank to identify where the film is located.

- Task 2: - The probe is moved in and out of the surface over 250 x 250 mm grid, covering an area of 1000 x 500 mm. This task mainly relates to gamma radiography and represents the positioning of the collimator at a precise offset from the surface in a precise position. This task would be carried out after the position of the Image Locators has been identified, using task 1.

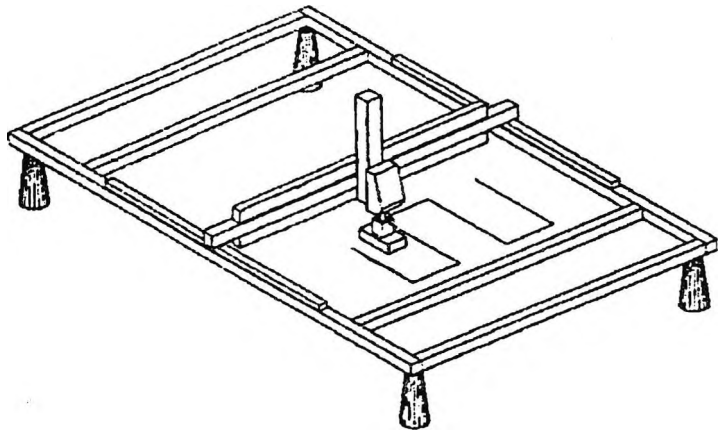


Figure 7-5 Working area of cartesian robot

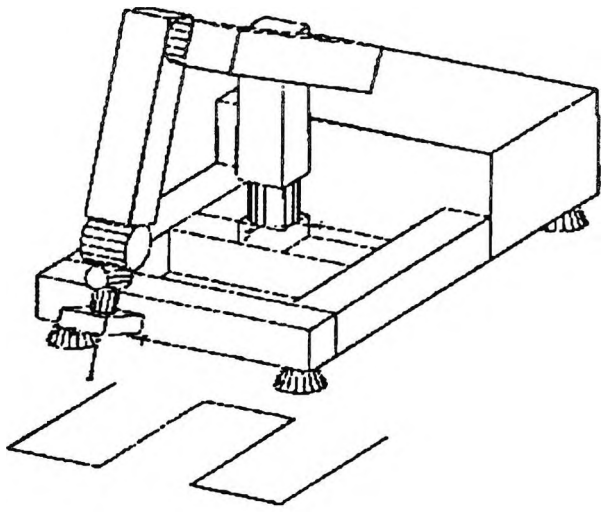


Figure 7-6 Working area of PUMA robot

In both tasks the probe is to be kept normal to the working surface. In the case of the cartesian robot the working surface is bounded by the frame (see figure 7-5. The working surface of the PUMA is the area in front of the working platform (see figure 7-6

The overall specification of the models can be seen in Table 7-2, together with the idealised joint specifications for both robots, (see figure 7-7).

Factor	Cartesian Configuration	PUMA Configuration
Degree's of Freedom	5(XYZ+ pitch + yaw)	6(inc. roll+pitch+ yaw wrist)
Maximum Velocity	500 mm/s	500 mm/s
Maximum acceleration	250 mm/s	250 mm/s
Mass	40 Kgs (assumed)	30 Kgs
Maximum reach	1.5 M (across diagonal)	1.3 M (as used)
Approximate Cost	20 K	30 K

Table 7-7-2 Overall specifications: PUMA and Cartesian Robot

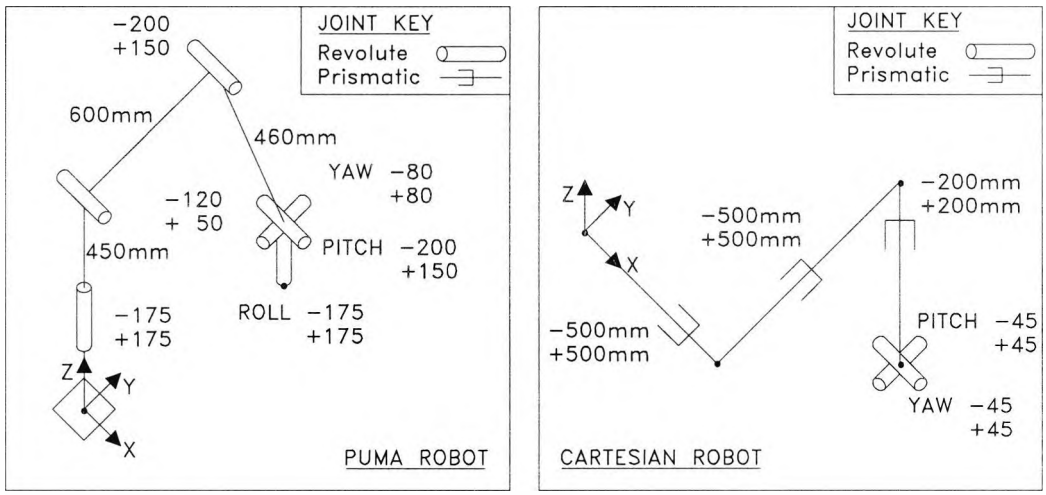


Figure 7-7 Idealized joint specifications for Puma and Cartesian robots

7.1.2.2 Relative Robot Performances

Both of the robots described in section 7.2.2.1 were applied to each task. The tasks were performed at different accelerations and a maximum velocity of 40 cm/s. The completion time for each task was then plotted on a graph of time against acceleration. This graph can be seen in figure 7-8 from which it can be seen that the 5 Degree of Freedom (DOF) cartesian robot, was on average 8.5% faster in the completion of both tasks, and the major factor which affects this is the acceleration of the robot arm. This

is not surprising, given the short distances between the points that the robot has to move. This was verified by running a series of simulations, where the percentage acceleration was set for a series of varying velocities for task 2, using the cartesian robot. Two types of move were considered, a synchronous move, where more than one axis moves at a time and an asynchronous move, where only one axis moves at a time. The results can be seen in figure 7.9. The acceleration has an important impact on the selection of the drive system, as it is limited by the maximum torque that the drive system can produce, thus a cartesian robot would require less torque to achieve the same results as the PUMA robot arm.

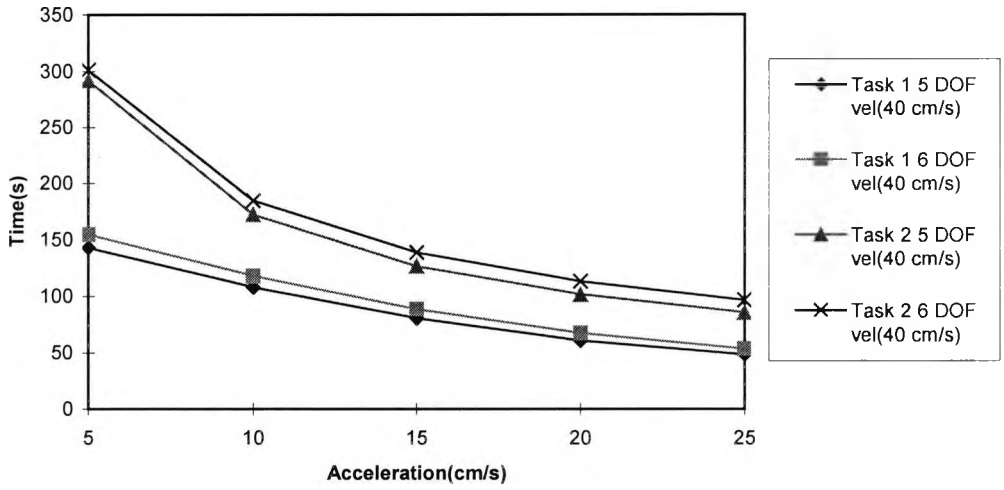


Figure 7-8 Time against acceleration for both tasks, and robots

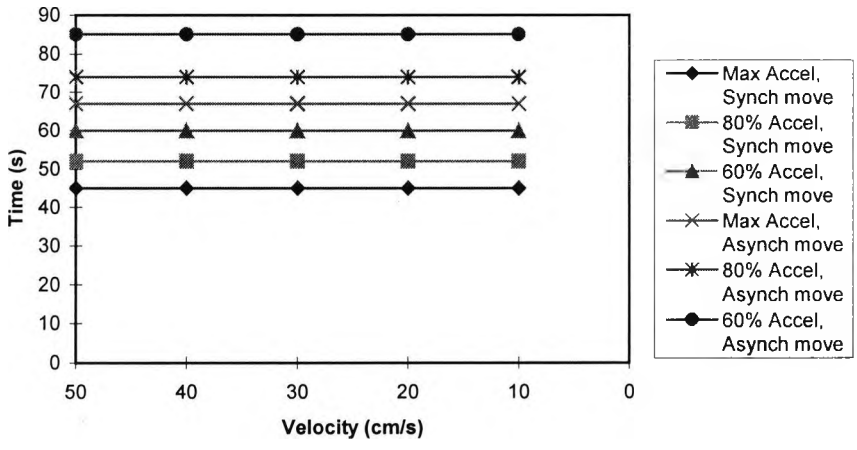


Figure 7-9 Comparison of Asynchronous to Synchronous moves for Task 2

Whilst the simulations were being carried out, it was observed that the changing position of the arm of the PUMA robot caused significant alterations in the position of the robots effective centre of gravity. This resulted in turning moments being produced at the lower attachment points, which were more than double the moments produced by the cartesian robot, thus increasing the attachment forces required to stabilise the PUMA robot.

It was also noted that the PUMA arm requires more careful positioning than the Cartesian robot. This is because the initial location of the arm does not guarantee that the arm will be able to subsequently position and direct the collimator correctly. To overcome this problem, it is necessary to ensure that the platform of the robot arm is as near as possible to the target area, particularly if a 300mm or more offset is required for the collimator. A further problem, which became apparent during the simulation, was that arm motion that would effectively point the collimator parallel to if not outwards from the work surface, a dangerous situation which must be avoided at all times. This could be guarded against in the robots motion controller, however it is envisaged that this would reduce the dextrous envelope of the robot, thus making the initial location of the robot even more critical.

Because the Cartesian robot completed the tasks in faster times, and in view of the safety, location, attachment and reduced working envelope problems associated with PUMA robot arm, it was concluded that the Cartesian robot represented the best option for the application.

7.2 Safety based customer requirements and general solutions

The initial system specification, summarised in section 6.5, gave three safety conditions:-

- Ensure the source position system cannot harm the operator whilst being located, operated or maintained.
- Ensure that if a fault is detected or occurs, the velocity of the source is less than or equal 0.0001M/S, at 0.065 seconds after the fault occurs.
- Ensure that if a fault is detected or occurs the position of the source is known to within +/- 1 mm , 0.065 seconds after the fault occurs.

Safety analysis can only be carried out properly on the first of the above conditions. This is because the other two are very specific and depend upon the integration of

specific components, yet to be decided, with the control software. However, as will be seen later, many of the faults which lead to the infringement of the first safety condition will also cause infringement of the second and third. This section deals with the detailed analysis for safety of the first condition using a CLASH/FTA approach (Seward[94]), as applied to the conceptual design of the cartesian robot and working platform selected in section 7.2.2.2. FMEA analysis is used to generate general solutions to the problems identified by the CLASH/FTA. These solutions can then be evaluated for suitability, later in the design process.

7.2.1 CLASH/FTA analysis

The first stage of the clash analysis is to identify possible dangerous consequences of employing the cartesian robot and working platform. These were identified as.

- Death, crushing or serious injury caused by the failure of the suspension mechanism.
- Back injury, caused by lifting or manipulating the robot and working platform.
- Death, caused by the possible removal of limbs if they get caught between sharp faces on the moving cartesian axes.
- The crushing of limbs or digits if they get caught between pieces of moving equipment or the cartesian axes and the side of the frame.
- The trapping of limbs or digits if they get caught between pieces of moving equipment or the cartesian axes and the side of the frame.

The consequence of injury caused by failure of the suspension mechanism leads to the requirement that the suspension mechanism should be sufficiently strong to support the robot on the wall.

The consequence of back injury caused by manipulating the robot can be dealt with in two ways. Either limit the weight of the robot so that two men can safely manipulate it, or if the robot weight is too great, develop a site delivery system that reduces the need to manhandle the robot.

The remaining consequences are caused by a combination of the operator placing a limb or digits in unsafe position and varying degrees of uncontrolled movement of the robot axes. These effects are summarised in figure 7-10, which depicts a high level CLASH tree. In this, the various consequences and possible combinations of causes

leading to the consequences are shown. The combinations of causes are depicted using boolean symbology derived from FTA.

From this initial analysis, three major causes are identified as being necessary to effect the shearing, crushing or trapping of limbs. These are, varying degrees of moving forces, colliding faces and limbs or digits in the danger zone. These causes are further analysed to identify what possible faults may exist in a system to induce them. These, are depicted using a FTA tree, see figure 7-11.

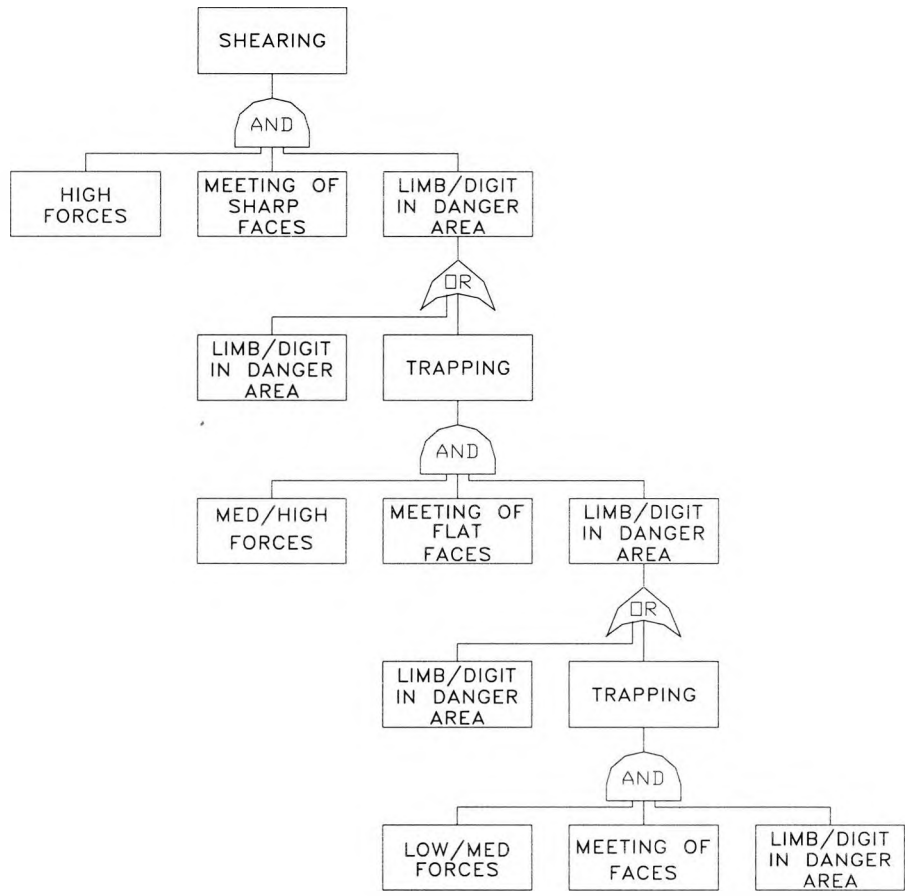


Figure 7-10 CLASH tree for Gamma Radiography robot

A number of faults were identified which can possibly lead to the consequences outlined above. The low-level faults identified mainly covered hardware and software failures. Although in the case of colliding faces and limbs or digits in the danger zone, they are mainly caused by lack of guards and the failure of safety switches.

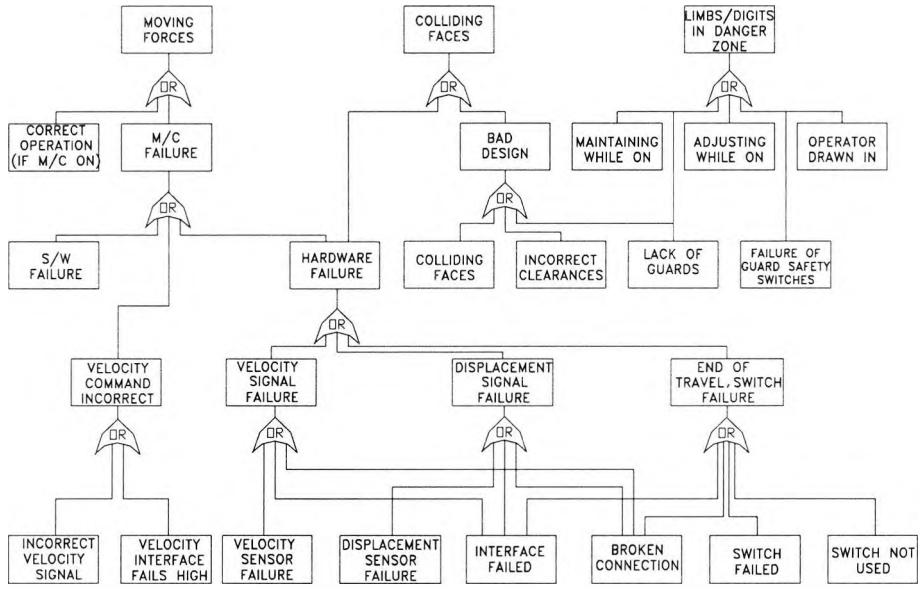


Figure 7-11 Expanded CLASH tree/FTA for Gamma Radiography robot

7.2.2 Initial FMEA

A FMEA (see table 7-3, see below) was carried out to determine the severity of the failures and possible means of negating failures. The reason for failure is dependent upon the means used to realise the function. However there are several standard techniques which can be used to implement the functionality. In this case, methods such as, sensor redundancies combined with safety software are used to generate solutions. The solutions proposed by the FTA are evaluated for suitability, later in the design process.

The FMEA chart, table 7-3, indicates that the most likely fault to occur (given by the highest RPN number) is 'access whilst the robot is on' (RPN = 540). This mainly occurs when maintenance is being carried out and is dangerous because it is possible that the safety cut-outs may be disabled. It is, however, a situation that cannot be avoided as it is necessary to get close to the robot to test and observe its behaviour. The only safety guidelines which are available in these circumstances is to ensure that when maintenance is being carried out, two people are present and full training given.

The majority of the other faults are caused by signal, interface or connection failures. These can be dealt with by the use of independent and redundant sensors to check each other. Hardwired safety interlocks could also be employed to some aspects of failure.

7.2.3 Summary of CLASH, FTA and FMEA

The negation of the faults identified above, lead to a number of customer requirements, guidelines and solutions which can be input into the second level EQFD chart. They are not included in the first stage EQFD charts because they are too detailed. However they do come under the general heading of 'SAFETY'. The generalised requirements/constraints identified from the CLASH are: -

- Employ independent or redundant velocity and displacement sensors to control axes movement.
- Ensure the distance between all moving surfaces and the robot frame is large enough to prevent the entrapment of a human limb.
- Employ guards with redundant, interlocked safety switches into the design.
- Ensure two man operation so that if an accident does occur, help is at hand.
- Limit weight of robot/working platform so that two people can manipulate it safely or design a safe delivery system.
- Ensure suspension mechanism and cable is sufficiently strong to support the robot and additional equipment.
- Ensure maintenance staff are properly trained to prevent accidents whilst safety features, such as guard cut-outs, may be overridden.

Some of the above customer requirements, such as, ensure the suspension system is capable of carrying the load, ensure two man operation, so that aid is on hand if an accident does occur. The training needs of maintenance staff are fairly obvious. It is likely that they would have been identified as implicit requirements, through the normal course of events. Requirements that would have been less obvious as being important, would have been the identification of distance between the frame and any moving axis to prevent limb entrapment. Another would have been the use of redundant sensors for displacement and velocity sensing to help identify when an axis is moving when it should not be.

Fault	Probability 1low -10 high	Detectability 1 easy- 10 hard	Criticality 1 trivial- 10 severe	RPN	Possible solutions
Sensor failure Velocity Displacement	Rare (1) Rare (1)	Likely (2-3), achieved by comparison, of displacement and velocity using software.	High (7), unaware of speed or position of load. Important to prevent motor burn and aid detection of trapping.	21	Use separate velocity and displacement sensors, check signals against each other using software.
End of Travel (EOFT)	Unlikely (2)	Hard (7), no redundancy in place.	Severe (9), unaware if motor stops at EOT can lead to burn out .	126	Use two sensors at each end of travel.
Interface Fail Velocity command	Rare (1)	Likely (2-3), compare output velocity to input from sensor	Low (2), unlikely to set line high, normally fail safe, zero velocity	6	Employ independant sensors for velocity and displacement measurement, integrate with safety software
Displacement dependant upon method used	Unlikely (2), Counter voltage	Unlikely (8), only at end of travel when datuminng	Severe (9), as there is no indication of position	144	
Velocity input Voltage (A-D)	Voltage (3) (spikes)	Likely (3) , only able to check input against a counter.	Medium (6), derive velocity from counter using S/W.	54	
Counter	Counter (3) (too fast)	Unlikely(8), as only method employed.	High (9), as there is no indication of position.	216	

Table 7-3Proactive FMEA chart for Gamma Radiography robot (continued overleaf)

Fault	Probability 1low -10 high	Detectability 1 easy- 10 hard	Criticality 1 trivial- 10 severe	RPN	Possible solutions
Interface Fail End of Travel switch	Unlikely (2)	Unlikely(8), only by checking displacement	High (9), could lead to motor burn out	144	set switch normally closed so failure causes it to open and stops movement.
Broken Connection	Unlikely (1)	Varies (2-8), dependant upon line down	Varies (2-8), dependant upon line down	144	effect is similar to interface failures, use method proposed above or hard wire logic and safety cutouts
Colliding faces or **Incorrect clearance**	High (9), dependant upon design	(3-4), analysis of mechanisms.	High (7-8), can cause trapping or shearing.	288	Prevent access by guarding Prevent occurrence by reference to BSI standards Good design
Lack of Guards	High (9) Dependant upon design.	High (1), if present use micro switches to detect guards,	Severe (9), allows access to moving parts.	81	use guards and hard wired safety cutouts
Access while on	High (9) whilst maintaining.	Low (7), as safety features probably turned off.	Sever (9), allows access to moving parts.	540	Employ two operators whilst maintenance is being carried out Training for maintenance personnel, Guarding + safety cut-outs.

Table 7-3 Proactive FMEA chart for Gamma Radiography robot (cont.)

Despite the fact that both the measures mentioned above are described in BSI[91a] and HSE[87,93], BSI[92], respectively , the documents are confusing and cover a wide range of circumstances. This makes it hard to identify where a particular guideline can be employed. The use of the CLASH approach acts as a tool to identify appropriate requirements or guidelines which can be used by the designer, who does not have a detailed understanding of the publications listed above. The other advantages of the CLASH approach is that the designer gains an understanding of why a certain safety requirement is made, and this knowledge is captured in documentation for future reference.

The FMEA chart provides a means to identify general solutions to a problem, however because the design has not progressed beyond a basic concept at this stage, they cannot be employed. They can, nevertheless be kept in mind when the solutions to other customer requirements and functions are being investigated. In this way, they force the designer to take a proactive stance towards safety.

7.3 Integration of general design tools into EQFD

This section focuses on the design of the X,Y, and Z, drive mechanisms and drive control hardware for the robot working platform. It does not cover the design of the suspension or attachment mechanism. The design was developed by employing EQFD and various general purpose design tools, which are used to control and map the EQFD process. It provides a contribution to the Ph.D. because it generates the information, from which the ACP models and computerised tools used later are derived.

7.3.1 1st Stage EQFD chart and top level functional analysis

In the original attempt at applying DFD/EQFD (Chamberlain[95]) it was found that one of the major problems of employing the method was that too many customer requirements and functions were identified at the first level. This led to a series of very large complex evaluation matrices. To get over this problem, it was decided to reduce the number of functions and customer requirements of the first level matrix to there barest minimum. This enables the functions and customer requirements to be expanded upon later, in a more controlled manner, so that the designer is not overwhelmed with information at an early stage.

The customer requirements identified in the previous section were expanded upon and rationalised, which led to a series of explicit and implicit requirements. The explicit requirements cover the performance of the robot, whilst implicit requirements cover good engineering practice, i.e. high efficiency, low friction, safety (in general terms), for example. The requirement for a payload of 22 Kg is derived from the requirement that a yaw and roll axis are required along with the payload of 5 Kg for the collimator.

In the original application of DFD/EQFD, a separate function was defined for almost each customer requirement. This invariably interacted with other requirements and led to the large matrix. To overcome this problem, it was decided to take a systems engineering approach, where the design engineer identifies a minimum number of high level functions which go towards the functionality of complete design. In this instance, the functions identified are:-

- The main power supply and drive type
- The drive mechanism
- Control system

A first level EQFD showing this information can be seen in fig 7-12 below.

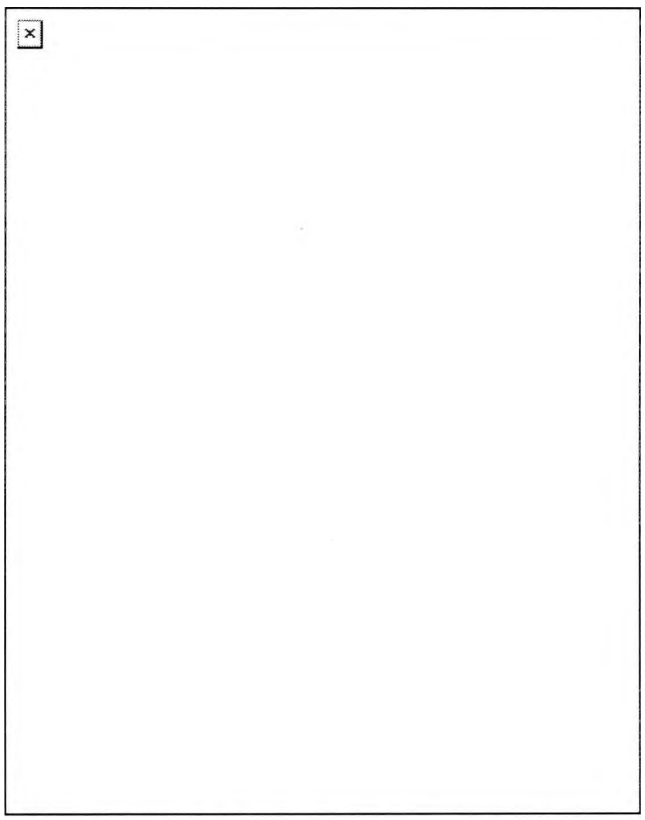


Figure 7-12 Top Level EQFD Chart

Weightings for technical complexity and customer importance are given for each function and customer requirement respectively. These are used to magnify the strength of the relationship between a customer requirement and a function. These relationships are given scores of 3, 2 or 1 for strong, medium or weak, respectively. Scores down the left-hand side are the product of the sum of the relationships multiplied by the customer importance weighting. They are used to indicate the most important customer requirements. Here safety and cost are indicated as the major drivers.

	WHATS \ HOWS	CUSTOMER WEIGHTING	ROTARY DRIVES			LINEAR DRIVES			LINEAR BEARINGS		ROTARY BEARINGS		CONTROLLER TYPES	
			HYDRAULIC	ELECTRIC	PNUEMATIC	HYDRAULIC	ELECTRIC	PNUEMATIC	BALLS	PLATES	BALLS	BUSHS	CONTINUOS	MICRO BASED
EXPLICIT	TECHNICAL COMPLEXITY													
	LOW COST	5	-2	0	0	-2	-2	0	0	2	0	2	2	-2
	LARGE DISPLACEMENT (1000 mm)	4	2	2	2	-2	-1	-1	0	0	0	0	0	2
	HIGH VELOCITY (500 mm/S)	4	2	2	2	2	2	2	2	0	2	0		
	MEDIUM LOAD (22 Kg.)	4	-2	0	0	-2	0	0	2	0	1	0		
	SPEED & DISTANCE CONTROL	4	2	2	-1	2	2	-1					0	2
	PATH GENERATION	3											0	2
	MULTI-AXIS CONTROL	4											0	2
IMPLICIT														
	LOW FRICTION	5	0	0	0	0	0	0	2	0	2	0		
	HIGH EFFICIENCY	5	1	1	0	1	1	-1	2	0	2	0		
	LOW WEIGHT	5	-2	0	0	-2	-2	0	0	2	0	2		
	IMPART STRUCTURAL INTEGRITY	3	0	0	0	1	2	1	2	1	2	1		
	DELIVER DRIVE TO BOTH SIDES	5	1	1	1	-2	-2	-2	0	0	0	0		
	EASE OF MANUFACTURE	4	-2	1	0	-2	-2	-2	1	2	1	2		
	USE STANDARD PARTS	3	0	0	0	0	0	0	2	2	2	2	2	2
	EASE OF MAINTENANCE	4	-2	2	0	-2	0	0	1	2	1	2		
	SAFETY	5	-2	0	-1	-2	0	-1	0	0	0	0	0	2
			-32	46	12	-43	-15	-11	56	45	52	45	16	36

Figure 7-13 First Stage EQFD Chart

Technical difficulty ratings are produced in a similar manner, except that they are calculated vertically. These are used to indicate areas were the most technical effort should be apportioned. In this instance, the control system is the most complex, followed by the drive mechanism and finally the power and type of drive.

The functions were expanded upon and vague solutions proposed for each function, options covering electrical, hydraulic or pneumatic powered drives of either a rotary or linear type. When these options are combined with types of bushes, the drive mechanism is beginning to be defined. Alongside the drive mechanisms, two types of control system are proposed, microprocessor based or a continuous system. These options are presented in a more detailed first stage EQFD chart shown in figure 7-13, below and at the top level of a function means tree, figure 7-17, shown in section 7.4.3. The function means tree is used as general mapping of the process and shows the results of decisions made at each stage of the EQFD process.

From this first level EQFD chart, a basic concept is selected, based upon use of Cross's weighted objectives (Cross[89]), the strength of the relationship between a customer requirement and a function or means is multiplied by the customer weighting. These products are then summed to give a final score. In this instance, the combination of means to realise the drive, drive mechanism and controller are an electric rotary drive (a type of motor), linear bearings and a microprocessor/ computer control drive system. This is indicated by the size of the scores for each option. A computer controlled drive system is the only viable option, given the complexity of the safety circuitry, which would be required, if the safety options identified in section 7.3.2 were employed.

This process of expanding upon the means to realise a function and then selecting a particular means of realising the function, is typical of divergent/convergent design (Roozenburg[95]) . It is a process that is carried out repeatedly through the design development, and it can be seen in the function-means tree in figure 7-17.

7.3.2 2nd Stage EQFD charts and sub function analysis

The functions for the second stage charts are derived from the expansion of the concepts already selected. To achieve the transformation of electrical power into linear motion a number of transformation functions are required. They cover: -

- Electrical to rotary: - types of motor.
- Velocity signal to electrical: - rotary or linear velocity transducers.

- Displacement signal to electrical: - rotary or linear velocity transducers.
- Rotary to Linear displacement: - means of affecting the change from rotary motion to linear motion.
- Inertia matching: - used to match inertia of driver and driven, the degree of this matching affects the rate of acceleration (Unimatic[93]).
- Effort transfer: - means of transferring torque to both sides of the frame simultaneously.
- Electrical to information: - input interface electronics.
- Information manipulation: - computer hardware used for control, path planning and safety.
- Information to electrical:- output interface electronics
- Safety procedures :- Means of realising the various safety related customer requirements

Customer requirements that relate to the transformation functions are then selected and entered down the left-hand side of the matrix. They are expansions of the earlier requirements and are closely related to the transformation functions. For instance, the requirements, good torque/speed curve and low inertia, relate directly to the use of low weight, electric motors. This can be seen in the EQFD stage 2 chart in figure 7-14.

The strength of the relationships between the transformation functions and customer requirements are then plotted on the chart. This is used as a guide when the chart is expanded to include the means of realising the various functions. It also ensures that each requirement is met by at least one transformation function. Also note that the safety requirements, outlined earlier, are now included as customer requirements.

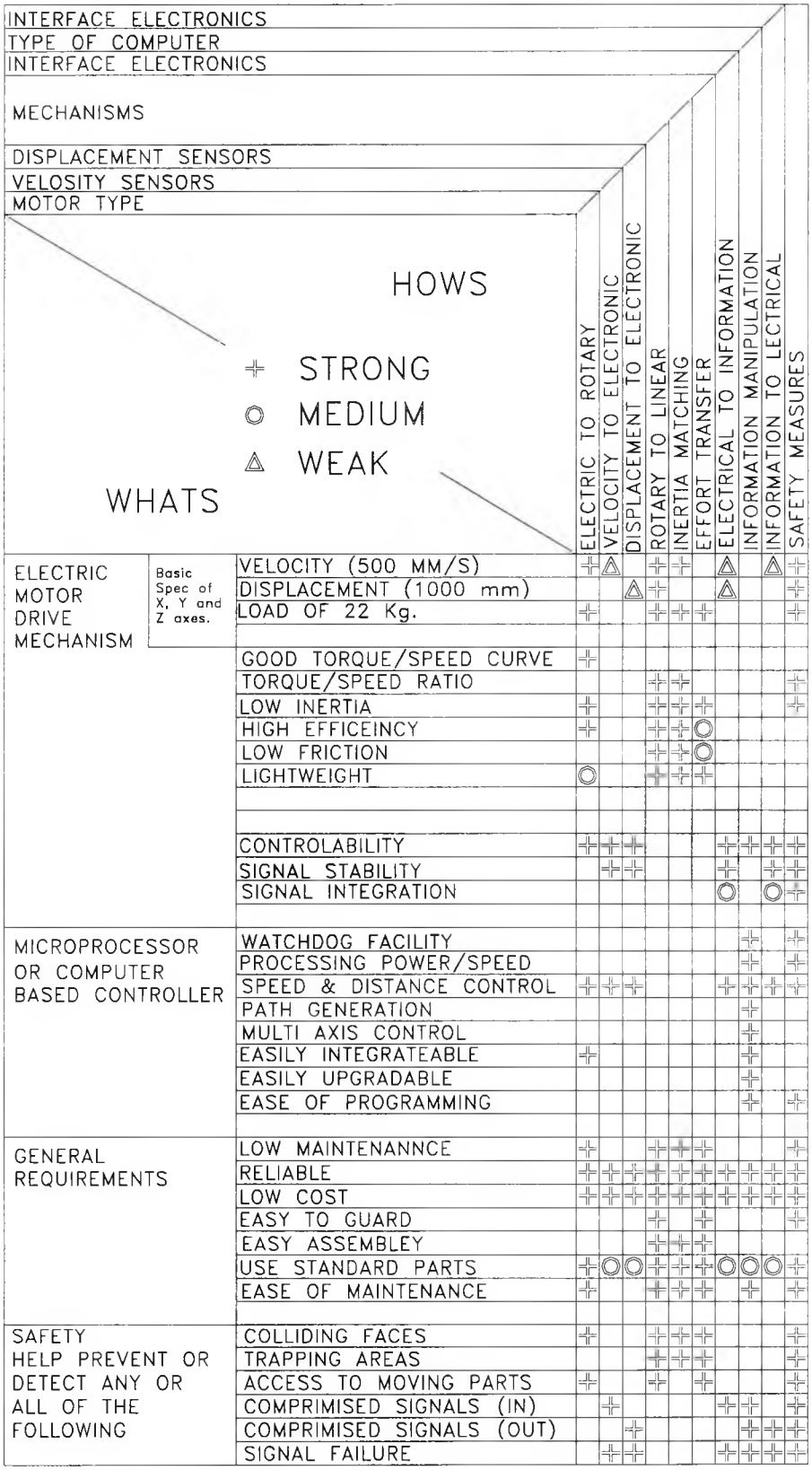
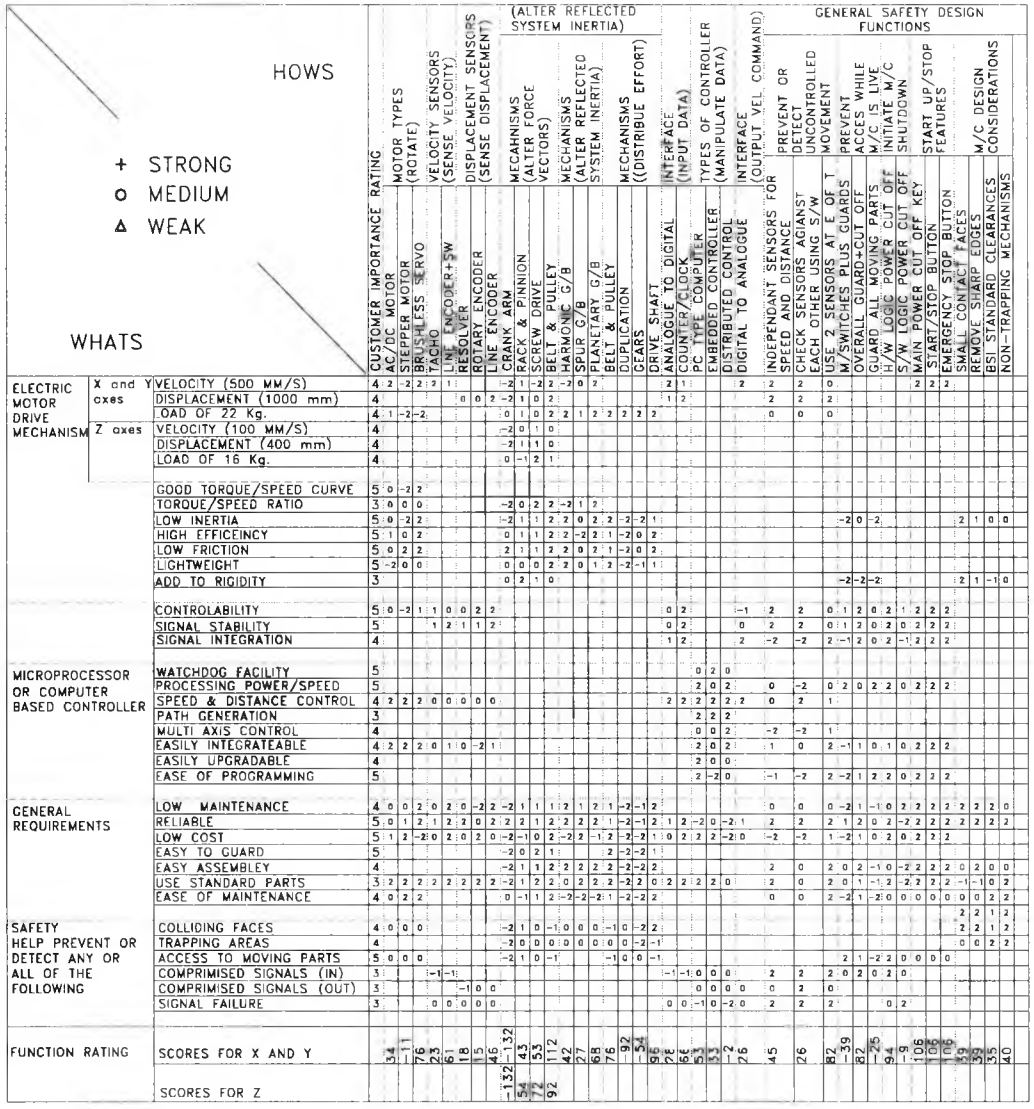


Figure 7-14 Stage 2 EQFD chart

7.3.2.1 Development of complete concept solutions

Each of the functions in the first stage 2 chart are expanded upon by finding solutions to them. These are initially mapped into a morphological chart. This allows all options to be placed in front of the designer. The morphological chart for this example can be seen in table 7-4. The number of combinations available to the designer, in this case, is in excess of a million, (calculated by multiplying the number of options for each function). A large solution space is a common problem in design (Roozenberg[95]), however, it can be reduced by evaluating the means of realising each function against the customer requirements, laid out in the first, stage two chart. The evaluation was carried out using Cross's weighted objectives method, and can be seen in figure 7-15. From this evaluation, various options can be automatically excluded from the morphological chart, because of the low scores they achieved. The options removed from the evaluation are highlighted in table 7-4.

The design can now be split into three parts, the drive mechanism and transducers, control system and safety measures. Of the safety measures, all the options remaining after the removal of the highlighted options can be implemented without a significant increase in cost or complexity. For the control system, an A-D, D-A counter and I/O ports are all necessary, if any form of voltage or discrete signal is to be used for control. Cards that support this functionality are fairly accessible and cheap. The use of these cards also allows the line encoder to be used for velocity checking (for safety). The Selection of a PC over an embedded computer was based purely upon cost, PCs being considerably cheaper than industrially based, embedded, computer bus systems. This was a decision that was regretted later, as the PC was found to be not as reliable (computer boot up failures and communication interrupts causing program seizure), as a computer with an industrial type bus.



igure 7-15 Expanded stage 2 EQFD chart

TRANSFORM-ATION FUNCTIONS	REALISATION OPTIONS			
ELECTRICAL TO ROTARY	AC/DC MOTOR	STEPPER MOTOR	BRUSHLESS MOTOR	
VELOCITY TO ELECTRICAL	TACHOMETER	LINE ENCODER & SOFTWARE		
DISPLACEMENT TO ELECTRICAL	RESOLVER	ROTARY ENCODER	LINE ENCODER	
ROTARY TO LINEAR	CRANK ARM	RACK AND PINION	SCREW DRIVE	BELT AND PULLEY
ALTER SYSTEM INERTIA	HARMONIC GEARBOX	SPUR GEARBOX	PLANETARY GEARBOX	
DISTRIBUTE EFFORT	BELT AND PULLEY	DUPLICATE SYSTEM	GEAR TRAIN	DRIVE SHAFT
ELECTRICAL TO INFORMATION	ANALOGUE TO DIGITAL	COUNTER CLOCK	INPUT/OUTPUT PORTS	
MANIPULATE INFORMA TION	PERSONAL COMPUTER	EMBEDDED CONTROL	DISTRIBUTED CONTROL	
INFORMATION TO ELECTRICAL	DIGITAL TO ANALOGUE	INPUT/OUTPUT PORTS		
PREVENT ERRORS IN MOVEMENT	INDEPENDANT SENSORS FOR SPEED AND DISTANCE	CHECK SENSORS AGAINST EACH OTHER	REDUNDANT MICRO-SWITCHES	
PREVENT ACCES WHILE ON	GUARD ALL MOVING PARTS	MICRO-SWITCHES & GUARDS	OVERALL GUARD & MICROSWITCH	
INITIATE M/C SHUTDOWN	H/W LOGIC SUTDOWN	S/W LOGIC SHUTDOWN		
START/STOP FEATURES	START/STOP BUTTON	EMERGENCY STOP BUTTON	MAIN POWER CUT OFF KEY	
M/C DESIGN FOR SAFETY	SOFTEN CONTACT FACES	REMOVE SHARP EDGES	STANDARD CLEARANCES	NON TRAPPING MECHANISM

Table 7-4 Morphological chart for Y and X axis concept generation

The drive mechanism and transducers are the only features that have not been selected. Possible concepts are derived by mapping combinations of the remaining functions to arrive at, viable solutions. These are shown in tables 7.5-7.7 A line encoder was selected over the resolver, in all cases, because it is less complicated to interface computer hardware to and analyse the signal.

Table 7.5 shows the functional layout of a solution, which is based upon a rack and pinion system. Table 7.6 shows the function layout of a solution that is based upon a screw drive system. Table 7.7 shows the function layout of a solution based upon a belt and pulley system. The arrows on the chart indicate flow of forces and signals.

TRANSFORM-ATION FUNCTIONS	REALISATION OPTIONS			
ELECTRICAL TO ROTARY	AC/DC MOTOR	STEPPER MOTOR	BRUSHLESS MOTOR	
VELOCITY TO ELECTRICAL	TACHOMETER	LINE ENCODER & SOFTWARE		
DISPLACEMENT TO ELECTRICAL	RESOLVER	ROTARY ENCODER	LINE ENCODER	
ROTARY TO LINEAR	CRANK ARM	RACK AND PINION	SCREW DRIVE	BELT AND PULLEY
ALTER SYSTEM INERTIA	HARMONIC GEARBOX	SPUR GEARBOX	PLANETARY GEARBOX	
DISTRIBUTE EFFORT	BELT AND PULLEY	DUPLICATE SYSTEM	GEAR TRAIN	DRIVE SHAFT

Table 7-5 Morphological chart for gear train drive

In this solution, an AC/DC or brushless motor, located on the Y-axis, is used to drive a planetary gearbox. This is connected to a gear train, which transfers the drive to a shaft that has pinion gears on the end which locate into the rack. A line encoder and tachometer are used for velocity and displacement measurement.

TRANSFORM-ATION FUNCTIONS	REALISATION OPTIONS			
ELECTRICAL TO ROTARY	AC/DC MOTOR	STEPPER MOTOR	BRUSHLESS MOTOR	
VELOCITY TO ELECTRICAL	TACHOMETER	LINE ENCODER & SOFTWARE		
DISPLACEMENT TO ELECTRICAL	RESOLVER	ROTARY ENCODER	LINE ENCODER	
ROTARY TO LINEAR	CRANK ARM	RACK AND PINION	SCREW DRIVE	BELT AND PULLEY
ALTER SYSTEM INERTIA	HARMONIC GEARBOX	SPUR GEARBOX	PLANETARY GEARBOX	
DISTRIBUTE EFFORT	BELT AND PULLEY	DUPLICATE SYSTEM	GEAR TRAIN	DRIVE SHAFT

Table 7-6 Morphological Chart for screw drive

Again, an AC/DC or brushless motor with line encoder and tachometer are employed for the initial drive mechanism. In this instance the drive is passed directly to a screw drive located on one side of the X axis. The screw acts as speed reducer and torque

magnifier. The drive power is passed to the other side using a belt and pulley arrangement.

TRANSFORM-ATION FUNCTIONS	REALISATION OPTIONS			
ELECTRICAL TO ROTARY	AC/DC MOTOR	STEPPER MOTOR	BRUSHLESS MOTOR	
VELOCITY TO ELECTRICAL	TACHOMETER	LINE ENCODER & SOFTWARE		
DISPLACEMENT TO ELECTRICAL	RESOLVER	ROTARY ENCODER	LINE ENCODER	
ROTARY TO LINEAR	CRANK ARM	RACK AND PINION	SCREW DRIVE	BELT AND PULLEY
ALTER SYSTEM INERTIA	HARMONIC GEARBOX	SPUR GEARBOX	PLANETARY GEARBOX	
DISTRIBUTE EFFORT	BELT AND PULLEY	DUPLICATE SYSTEM	GEAR TRAIN	DRIVE SHAFT

Table 7-7 Morphological chart for pulley belt drive

In the final solution the motor is connected to a planetary gearbox on the side of the X axis frame. The gearbox is used to drive a shaft which transfers the power to the opposite side of the frame. Toothed pulleys are located on either end of the drive shaft, secondary idler pulleys are located at the opposite end of the frame on either side. A toothed belt is placed over both sets of pulleys, the belt is then connected to the load on either side.

7.3.2.2 Final concept selection

The final choice for the design of the X axis drive was decided upon by taking the concepts listed above and evaluating them in a matrix which compares their functionality against the original customer requirements. This evaluation is shown in figure 7-16.

From this, it can be seen that the pulley belt drive system with planetary gearbox and drive shaft has the highest score, followed by the screw drive mechanism and finally the rack pinion. With this ranking in mind, it was decided to select the pulley belt drive mechanism option for the X drive. It was also decided to use a brushless motor (it scored highest in chart 7-15) as they are more flexible in their voltage requirements and have the best inertia/torque ratio.

	WHATS \ HOWS	CUSTOMER WEIGHTING	BELT AND PULLEY	RACK AND PINION	SCREW DRIVE
EXPLICIT	TECHNICAL COMPLEXITY				
	LOW COST	5	5	3	4
	LARGE DISPLACEMENT (1000 mm)	4	4	4	3
	HIGH VELOCITY (500 mm/S)	4	5	5	2
	MEDIUM LOAD (22 Kg.)	4	5	4	4
	SPEED & DISTANCE CONTROL	4	5	5	5
	PATH GENERATION	3	5	5	5
	MULTI-AXIS CONTROL	4	5	5	5
IMPLICIT					
	LOW FRICTION	5	4	3	3
	HIGH EFFICIENCY	5	5	3	3
	LOW WEIGHT	5	5	1	3
	IMPART STRUCTURAL INTEGRITY	3	2	3	4
	DELIVER DRIVE TO BOTH SIDES	5	5	2	4
	EASE OF MANUFACTURE	4	4	3	3
	USE STANDARD PARTS	3	5	4	5
	EASE OF MAINTENANCE	4	4	4	3
	SAFETY	5	4	3	4
			304	231	247

Figure 7-16 Stage 2 EQFD/ Weighted objects chart for concept selection

The evaluation process was repeated for the Y and Z axes. In each case the load displacement and velocity requirements changed, due to load orientation and the system configuration. From this evaluation, it was decided to use a pulley belt configuration for the Y axis, (very similar to the X axis mechanism). Screw drive configuration was selected for the Z axis, this being the most compact mechanism where the least speed is required.

Finally, because the solution for the X and Y axis mechanism was to employ a pulley, three further functions were generated, attach load to the pulley, attach load to bearing

and tension the belt. The solutions to these functions were basic and the use of a EQFD matrix for solution evaluation was thought to be extreme. The problem was solved by attaching the load to a plate, which was in turn, attached to a linear bearing, a small assembly was then used to attach the belt to the load and tension the belt. This simple mechanism can be seen at the base of the function means tree, figure 7-17.

7.3.3 Mapping the design process

Each of the stages of the above design process follow the law of vertical causality (Hubka[82]). The essence of this law is that a sub-function of a function cannot be defined, unless a solution to the original function has been selected. This is because the generation of sub-functions from functions, without a solution, presupposes a solution to the function and restricts the solution space before a proper search of the space has been made. The Law of Vertical Causality is in agreement with the EQFD philosophy of describing a function in neutral solution terms, but not in agreement with the DFD philosophy (Edney[93]) of describing primary secondary and tertiary functions without finding a solution to a high level function. This was one of the causes that resulted in large, unmanageable evaluation matrices in the early work on DFD (Bleakley[97]).

To ensure that every stage of the design process follows the law of vertical causality, function means trees (Buur[90]) are used to map the design process, figure 7-17, shows how the design progresses through EQFD.

At the top of the diagram the main objective of the design is described in neutral solution terms, in this instance, the objective it to produce a mechanism which will, 'Move a load along a linear axis, in a controlled manner.'

The functions describing the second stage 1 EQFD matrix are entered in polygons and the means of realising them entered into rectangles directly beneath them, indicated by lines. At the end of the first stage, some form of microprocessor control, an electric rotary drive (motor) and linear ball bearings have been selected as the means to realise the functions, type of controller, main power and driver, and finally bearing mechanism. It is now possible, for example, to determine the form of electric rotary drive to use. In this instance, the choice is between a basic AC/DC motor, stepper motor or a brushless servomotor, a brushless servo is selected. Each of the remaining functions are broken down in a similar manner and mapped directly onto different levels of EQFD matrix.

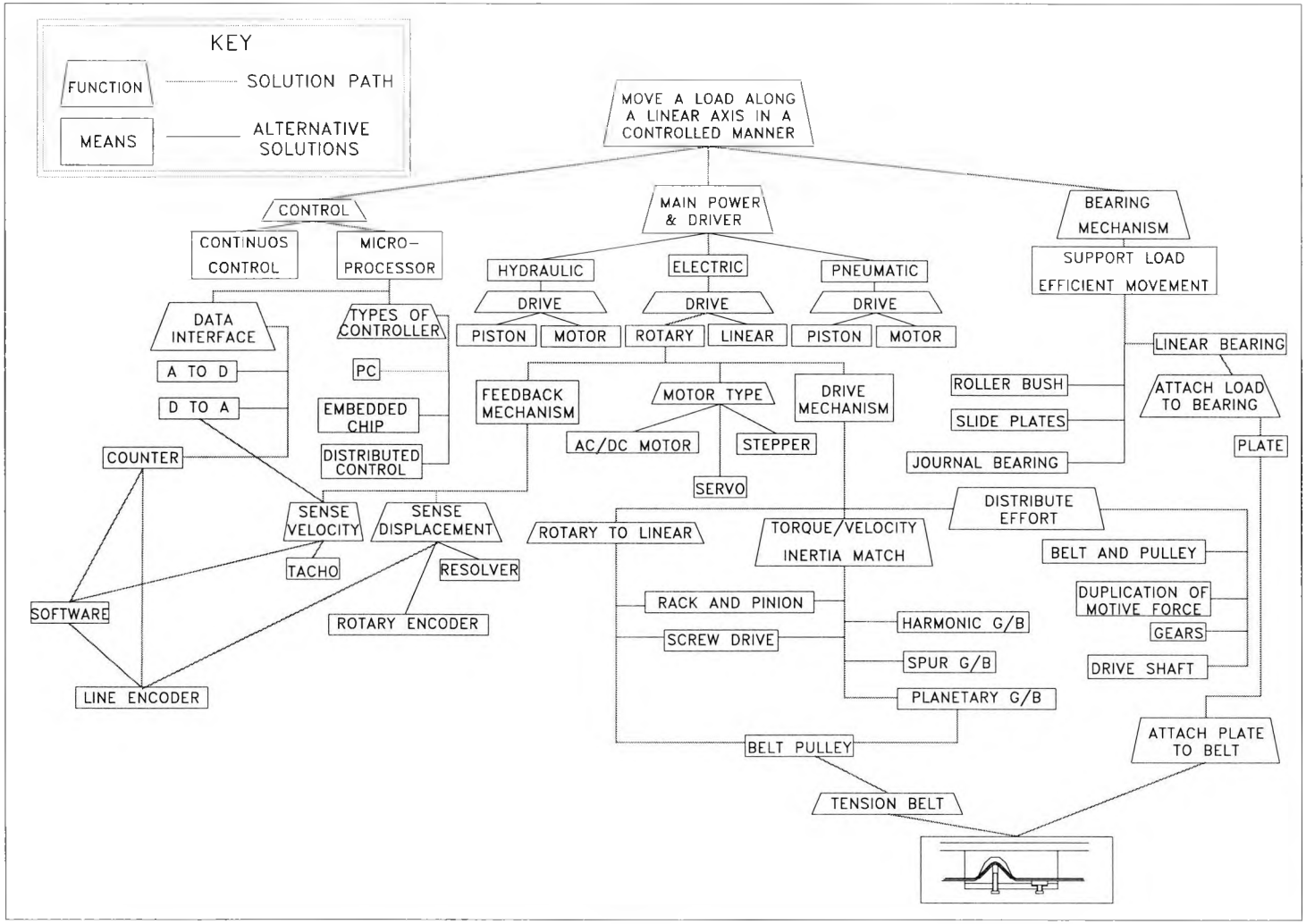


Figure 7-17 Function means tree for X and Y axis

The function means tree is drawn at the same time as the EQFD matrices are being developed. The tree show's the progress of the design without the detail of an EQFD matrix and allows the designer to review the choices that were made in a graphical manner. The thick line in figure 7-17 shows the final solutions that were used for each function.

7.4 Summary and Conclusions

In this chapter, the application of the proactive safety design tools and EQFD, described in chapter five, sections 5. 3 to 5.6, were described in respect to the safe design of the gamma-radiography robot developed at City University. The QFD analysis and computer simulation of the various working methods, and possible robots, provided the justification for initial concept of a cartesian robot with linear X, Y and Z axes. With this solution in mind, using CLASH and FTA, it was possible to identify possible ways in which personnel could sustain injury and the faults, which could lead to the injury. A FMEA was carried out at this stage to rank the faults and determine solutions to the various problems. The FMEA could be carried out at the appropriate detail level of EQFD chart for safety analysis, but this would not give a proactive stance, whereby the safety methods identified, control a large part of the design. For instance, the solution to the function of checking velocity in two different ways implies the use of computer interface cards and a microprocessor system to collect and compare the signals, respectively. Another example relates to the requirement that a user should not be able to trap a limb between the frame and an axis. This result's in detailed design constraints on the frame size and, X, Y and Z, axis dimensions, based upon BSI standards for safe clearances or anthropomorphic data.

Once the safety-based requirements were generated, the EQFD process was started. This involved two levels of diagrams. The first level gave more detail to the basic concept, determined basic drive mechanisms and control, whilst the second was employed to carry out detailed conceptual design. Eventually a design was arrived at which, employed brushless motors to power two pulley belt mechanisms for the X and Y axes, whilst the Z axis was driven using a screw drive mechanism. Load attachment and belt tensioning could not be dealt with until this stage because it was unknown what form the drive would take. Function means trees were used to map the EQFD design process.

The use of CLASH at the earliest possible stage of design was found to be very useful, identifying potential problems which could have easily been missed, and when finally recognised could have been expensive to fix.

Because EQFD was applied in a controlled manner (using the laws of vertical causality and function means trees), the design progressed in a much more structured way, forcing numerous solutions to be evaluated without the solution space becoming overwhelming. The final selection process of comparing complete concepts against the original customer requirements (from DFD, Edney[92]) was an effective way of ensuring that a near optimum solution had been found.

Function means trees were created as the design progressed and acted as an aid to ensure that the Law of Vertical Causality was observed. It was also used, as a record, to determine the design decisions that had been made.

The results from this design process are now used to generate the ACP diagrams from which computerised component selection for the X and Y axis is carried out.

8 Embodiment Design: Software models for component selection

8.1 Introduction

The objective of this chapter is to take the results from the stage 2 EQFD charts, the generic types of components, and determine specific components which fulfil the system specification for the X and Y axes of the robot. This process is initiated by setting the specification of the drives according to the criteria used to benchmark the Cartesian robot against the PUMA robot. The next stage involves, the generation of a correlation chain from a stage two chart correlation matrix, Commincini[94] from which a systems based ACP diagram (Bleakley [97]) is created. The ACP diagram encompasses the basic behaviour of the system, which is represented as mathematical equations buried within the diagram. From this information, dynamic VISSIM models are created which represent the system. Combinations of component variables are input systematically into the dynamic model and the results compared against the specification, to determine which combination of components fulfil the specification in the best manner.

The advantages of producing a computerised dynamic model for component selection are :-

- The motor response time and the effect of the back EMF on the changing motor output velocity and torque are taken into account.
- The effect of complex velocity curves (a common occurrence in robotics) on the motor loads can be analysed by inputting the velocity curve and examining the motor outputs.
- Increases the speed of the selection process and its accuracy.
- Once the model has been created it automatically takes into account the complex interactions of acceleration, velocity, torque, inertia and efficiency, which pass through the system.

Once the component combinations have been selected, the model is adapted to represent the selected configuration and computerised failure analysis is carried out (see chapter 9).

This chapter details the authors research into how computer tools can be integrated with EQFD using ACP diagrams and the correlation chain, essentially automating the use of the chain for component selection. By referring to Appendix C, the models and supporting files created for this analysis can be located and viewed.

The VISSIM models, produced at this stage, also provides the basis for the model that is employed for the systems safety analysis, component combination files, and log files, as detailed in chapter 9.

8.2 System specification

The specification for the selection of the X and Y axis motors is based upon the specification used to compare the cartesian robot against the PUMA robot in section 7.2.2. The important parts of the specification, with respect to the X and Y axis selection, relate to the velocity, acceleration and mass of the load to be moved along each axis. The specification for the X and Y axis can be seen below, table 8-1.

Specifcation	X-Axis	Y-Axis
Velocity	500 mm/s	500 mm/s
Acceleration	250 mm/s^2	250 mm/s^2
Load	24 Kg	16 Kg
Inertia matching ratio	3.5	3.5

Table 8-1 X and Y axis specification

The 24 Kg load for the X-axis is an estimate of the total mass of the Y,Z, Wrist and collimator. Similarly, the Y axis load is an estimate of the Z axis, wrist and collimator.

The inertia-matching ratio is set to 3.5, as an aid to optimise system performance. It is a measure of the inertia of the load, as seen by the motor, to the inertia of the motor. For optimum performance (high acceleration), the inertia seen by the motor should be equal to the load inertia (Unimatic[93]). A 1:1 ratio is unrealistic and implies a direct drive solution, therefore requiring a very large motor to achieve the torque necessary to drive the load at the specified acceleration rates. To solve this problem, some form of reduction mechanism (i.e. gearbox) is required to reduce the motor torque and increase the range of motor velocity. The inertia seen by the motor is known as the reflected inertia and is calculated from:-

Reflected inertia = Load Inertia / (Reduction ratio)^2 (Ogata[70])

The inertia matching ratio is calculated from:-

Matching ratio = Reflected Inertia/ Motor inertia

Guidelines(Unimatic[93]) suggest that a matching ratio of between 1 - 5 is generally acceptable. A ratio of 1: 3.5 was chosen, based upon the gearbox/pulley functionality to balance torque/velocity relationships as well as match system inertia's. An inertia matching ratio above 3.5 would imply lower reduction ratios and therefore higher motor torque requirements. Whilst below 3. 5, the reduction ratio would be very high (and as a consequence expensive), whilst the motors can be small but need high motor speeds to move the load at the correct velocities and accelerations.

8.3 Correlation chains and ACP diagrams

A number of physical and mathematical relationships exist between the various components, systems and subsystems. These interactions can be mapped on to the correlation matrix of the second stage EQFD matrix that contains all the generic components, systems and subsystems see figure 8.1, thus providing the basis for the ACP systems diagrams. The ACP diagrams are based upon two levels of model a systems model and a behavioural model (as described in section 5.7).

8.3.1 The correlation chain

As mentioned above, the correlation chain shows the physical and mathematical relationships between all the systems components. The correlation chain for the motor, gearbox, pulley system is shown in figure 8.1,. It is based upon the HOWS of figure 7.15. The correlation matrix is placed above the WHATS, and allows a relationship to exist between any of the components in the system.

Four symbols are used to indicate the presence of a relationship and the direction of influence of the relationship. The influences are:-

- Positive relationship (+):- The relationships between 'Remove sharp edges' , 'BSI standard clearances' and 'Non trapping mechanisms' are positive because by removing sharp edges, it can make it easier to comply with BSI standards and reduces the occurrence of trapping.

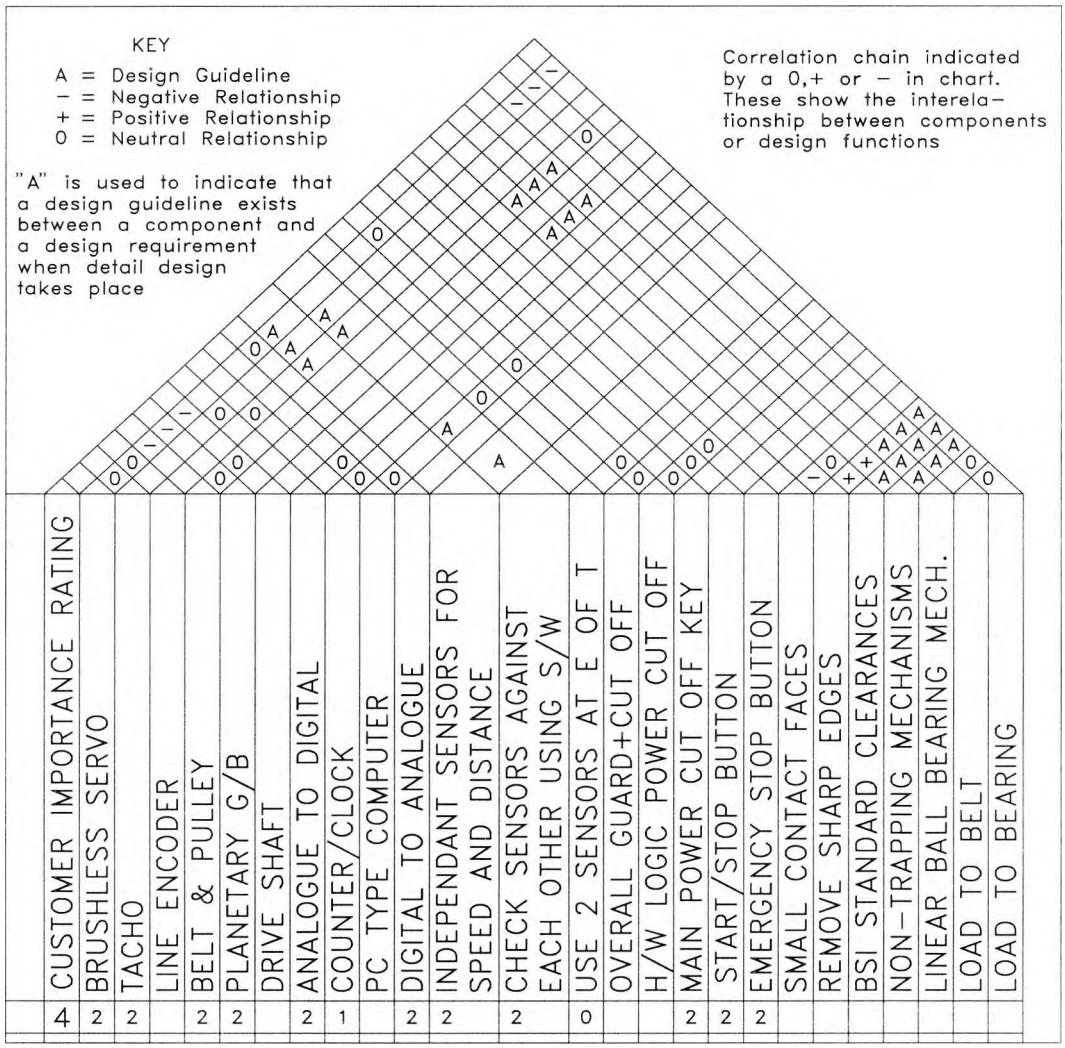


Figure 8-1 Correlation Chain for belt pulley drive

- Neutral relationships (O):- A neutral relationship indicates that the relationship between the components has very little positive or negative effect on the components. However, it can also be used to signify a physical link between components. For instance, the choice of encoder has little effect on the Counter /clock card to be used, but a physical relationship exist due to the wiring that connects them.
- Negative relationships (-):- Negative relationships exist when the effect of one component upon another produces a negative effect. The relationship between the motor and the majority of the components connected to it are negative because any load placed upon the motor increases the torque and inertia requirements.

- Design Guideline (A):- Design guidelines were added to the other three types of relationships to highlight the components affected by a particular guideline when the detail design takes place. In this case, the guideline to check sensors against each other should be taken into account when programming the computer and it affects the motor, tachometer and line encoder.

Firstly it must be noted that the correlation chain at this stage is a first estimate of relationships between components. The development of the ACP diagrams firm up the correlation chain as more understanding of the design is gained. Secondly, although the correlation chain indicates the relationships between the various components, in this form it cannot indicate the direction of the relationship, or energy. In many instances the relationship between two components can be positive in one direction and negative in another. For instance, the relationship between the motor and gearbox is negative due to the speed reduction and inertia effects, but it is positive in terms of the increase in torque. Recent work by Atherton[99] has proposed the use of square correlation matrices to counteract this problem and signify the positive and negative aspects of correlation chain relationships.

8.3.2 ACP systems model

The ACP diagram, which corresponds with the correlation matrix in figure 8.1 can be seen in figure 8.2, below. A system boundary is indicated around the motor drive system to indicate all the elements required to analyse and select the system components.

The diagram sets out the basic system configuration showing the physical and energy based connections between components. The arrows on the links indicate the direction of the energy flows. The links between components are guided by the points on the correlation matrix where functions/components interact.

In certain instances, the energy-based relationships are simple to analyse. For example, the link between the tachometer and the computer is an electrical voltage, generated by the tachometer, which represents the motor speed, and is passed to the computer via the analogue to digital interface. A more complex energy transfer relationship based upon the systems inertia, torque, displacement, velocity and acceleration exists inside the motor drive system boundary. The equations, which govern the behaviour of the drive

system, (derived in section 8.3.3) result in the acceleration produced by the motor. From the acceleration the load displacement and velocity are derived, being fed forward, whilst the inertia and load torque are summed and fedback. The inertia and load torque's are treated in this way because they have a direct bearing on the acceleration produced by the motor. This form of energy flow within the ACP diagram is consistent with block diagram control theory (Ogata[70]).

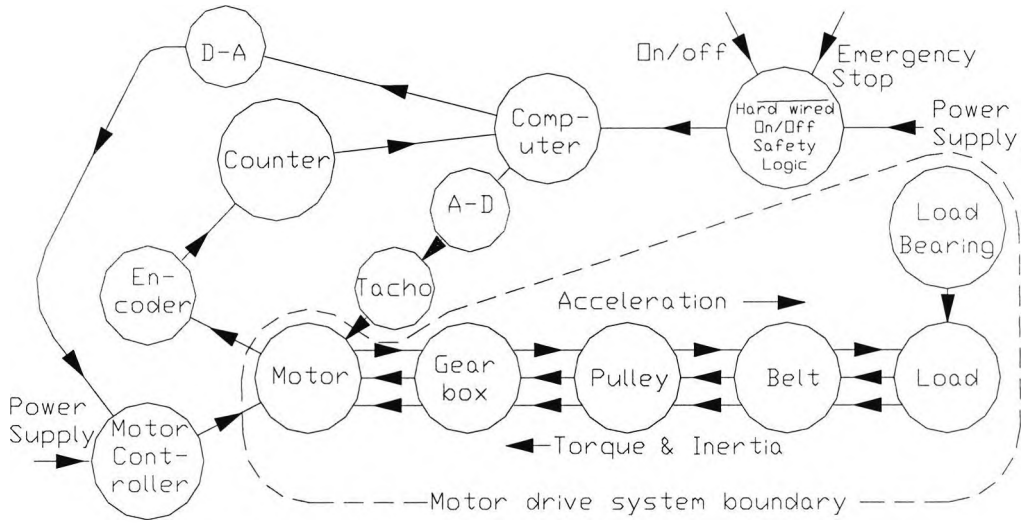


Figure 8-2 ACP diagram of complete drive axis drive system

8.3.3 Behavioural model

Once the system model has been generated, the behavioural model has to be created. The behavioural model is based upon the mathematical equations that govern the behaviour of each component. In this section, the underlying theory that controls each component, is discussed and linked with the other components to produce a detailed ACP model of the motor drive system.

8.3.3.1 Motor behaviour

The type of motor , selected in chapter 7, for the drive system was a brushless DC motor, this form of motor can be modelled quite adequately using equations based upon a standard permanent magnet DC motor (Kenjo(93), Miller(89)).

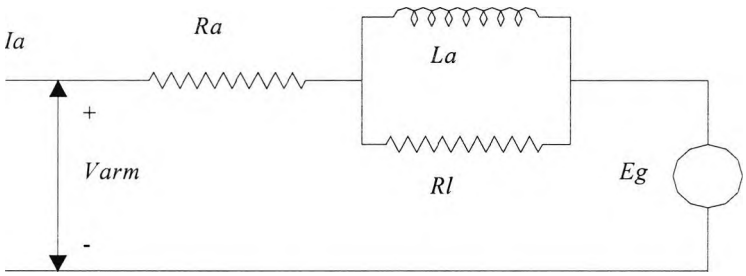


Figure 8-3 Armature circuit for a DC motor

A armature circuit model of a D.C. Motor is shown in figure 8.3, where:-

Ra is the total armature resistance (Ohms, Ω)

Rl is the resistance due to magnetic losses (Ohms, Ω)

La is the armature inductance (Henries , mH)

Ia is the armature current (Amps, A)

Eg is the back EMF produced by the motor, and is proportional to the motor speed

Varm is the voltage applied across the armature

The losses, *Rl*, can be ignored as there are considerably smaller than *Ra.*(Klafter[89],Electro-craft[77]).

The term *Eg* is proportional to the angular velocity ,ω (rads/sec), by a constant, *Ke* (known as the back EMF constant), so that:-

$$Eg = Ke\ \omega(t) \tag{8.1}$$

Applying elementary circuit analysis to figure 8-3, and combining it with Eq (8.1), gives:-

$$Varm = Ra\ Ia + La\frac{dIa}{dt} + Ke\omega(t) \qquad \text{(Klafter[89])(8.2)}$$

For D.C. motors the torque current relationship is proportional, so that:-

$$Tg = Kt\ Ia \tag{8.3}$$

where:-

Kt = Torque constant

Tg = Torque (Nm) generated by the motor

This generated torque is required to overcome four forms of torque (Pacific-Scientific [93]) present in the system:-

- Acceleration Torque (Ta) essential, to accelerate the motor and load inertias.

- Viscous damping torque(Tv) torque proportional to angular velocity, normally very small for most motors.
- Friction torque (Tf), Non linear torque required to overcome the coulomb friction present in the system.
- Gravitational torque (Tgr,), necessary to move loads perpendicular to gravity.

This gives:-

$$Tg = Tf + Tv + Tgr + Tj \tag{8.4}$$

Where :-

$Tv = Bm\ \omega(t)$, where Bm is coefficient of viscous friction and $\omega(t)$ is the angular velocity (rad/S).

$Tj = Jt\ \alpha(t)$, where Jt is the sum of the motor (Jm) and load (Jl) inertias (Nm/S^2), respectively and $\alpha(t)$ is the angular acceleration (rad/S^2)

Tgr and Tf are also combined to give Tl. They are not considered further in this analysis as they are loads external to the system (Klafter(89)).

Substituting these definitions into Eq 8.4 gives:-

$$Tg = Jt\alpha(t) + Bm\omega(t) \tag{8.5}$$

Rearranging Eq. 8.3 gives

$$Ia = Tg/Kt \tag{8.6}$$

Transforming Eqs 8.2, 8.5, and 8.6 into the Laplace domain

$$Varm(s) = RaIa(s) + LaSIa(s) + Ke\omega(s)$$

$$\therefore\ Varm(s) = (Ra + LaS)\ Ia(s) + Ke\omega(s) \tag{8.7}$$

$$Tg(s) = Bm\omega(s) + JtS\omega(s)$$

$$\therefore\ Tg(s) = (Bm + JtS)\omega(s) \tag{8.8}$$

$$Ia(s) = Tg(s)/Kt \tag{8.9}$$

Combining Eqs 8.7, 8.8 and 8.9 gives

$$Varm(s) = \frac{(Ra + LaS)\ (Bm + JtS)\omega(s)}{Kt} + Ke\omega(s)$$

$$Varm(s) = \frac{(Ra + LaS)\ (Bm + JtS)\omega(s) + KtKe\omega(s)}{Kt} \tag{8.10}$$

Transforming Eq 8.10 into standard output/ input form gives:-

$$\frac{\omega(s)}{Varm(s)} = \frac{Kt}{(Ra + LaS)(Bm + JtS) + KtKe} \qquad (8.11)$$

Eq 8.11 is the Closed Loop Transfer Function (CLTF) of the system and it is of the standard block diagram reduction form C/R = K / (G+ HK), (Ogata(70)) where :-

C = system output R = system input K = system gain

1/G = open loop forward transfer function H = feedback gain

Which equates to the following block diagram (figure 8.4)

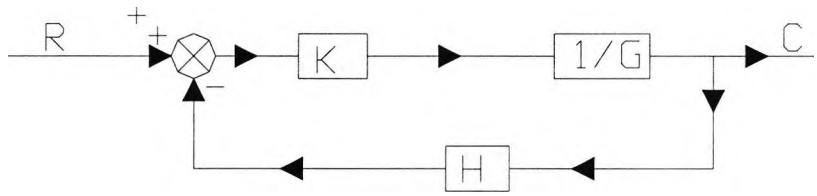


Figure 8-4 Standard CLTF block diagram

Converting equation 11 to the above form gives the block diagram, shown in figure 8.5. Note that *Tl* has now been included as an external input and the angular acceleration of the motor has been tapped of the second major block in the diagram.

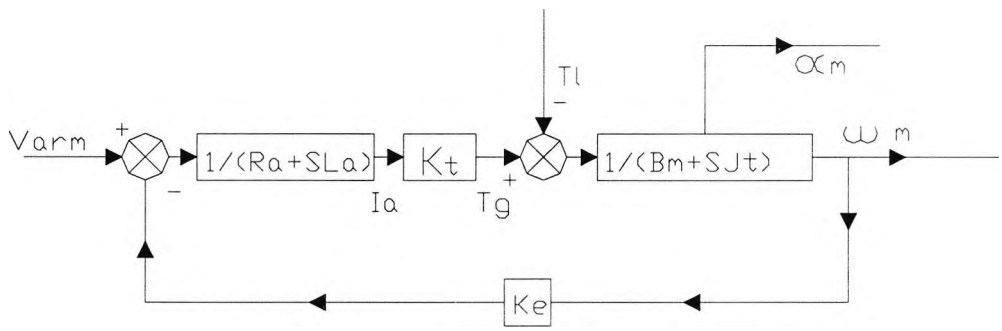


Figure 8-5 CLTF of an electric motor

Within the second block of figure 8.5, a further block diagram is buried which, shows the relationship between the total torque (Ttot) , αm and ωm, this can be seen in figure 8.6 . The angular acceleration is required so that the acceleration of the motor model does not exceed the maximum acceleration of the motor quoted in the specification.

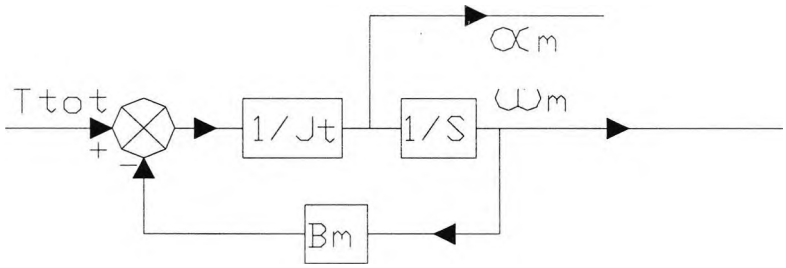


Figure 8-6 Mechanical transfer function of a motor

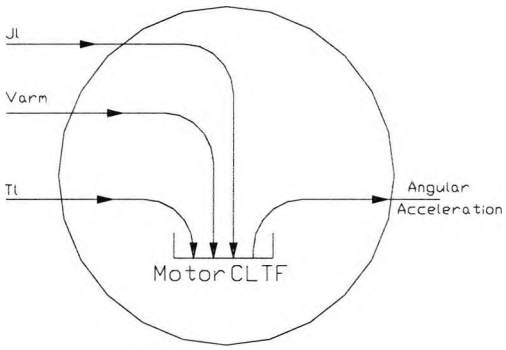


Figure 8-7 ACP diagram of motor

Figure 8.5 (Motor CLTF) can be considered to be an ACP diagram in its own right, with inputs of *Varm, Jl,* (as *Jt* = *Jl* + *Ja*) and *Tl,* and an output of αm. The detailed ACP diagram for the motor is shown in figure 8.7, the pool contains the CLTF of the motor and all the motor constants.

The formal ACP definition of this Motor Activity is: -

ACTIVITY: Motor (INPUTS: *Varm, Jl, Tl*
OUTPUTS: αm)

POOL: Motor CLTF (INPUTS: *Varm, Jl, Tl*
OUTPUTS: αm
VARIABLES: *Ra, La, Ke, Kt, Ja, Bm*)
BEHAVIOUR{ αm(s) = S(*Varm* * CLTF)}

Note that the functionality of a motor activity is described by an equation in the Laplace domain, as opposed to the logical relationship employed in the example in section 5.7.1. This is one of the great strengths of ACP diagrams, they enable a mixture of different modelling methods to be combined within the overall structure of the system model. This enables the system/design engineer to employ the appropriate representation tools for a particular part of the system, the representation method being encapsulated within the ACP diagram. ACP diagrams can, therefore, be considered to be a meta-modelling tool, a tool which exists above the other representation systems used in the model and encapsulates them.

8.3.3.2 Gear box behaviour

The gear box is connected to the motor and it is used to reduce the output acceleration, velocity and displacement of the motor, increase the motor torque and match the load inertia to the motor inertia. The basic layout of a motor gearbox load assembly can be seen in figure 8.8

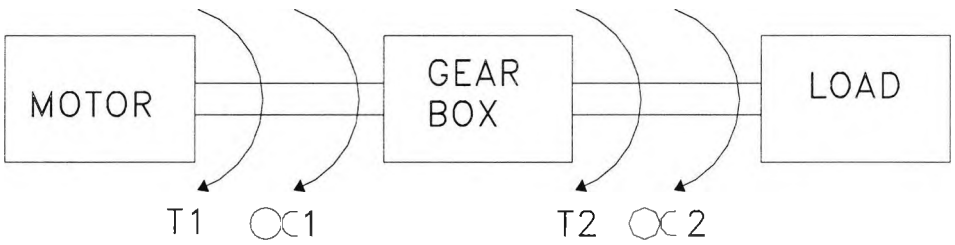


Figure 8-8 Schematic of motor, gearbox and load

The relationships which govern torque and acceleration are given below (Klafter[89]).

$$T1/T2 = \alpha 2/\alpha 1 = 1/GR \qquad (8.12)$$

Where

$T1$ = motor output torque (NM) $\alpha 1$ = motor output acceleration (rads/s)

$T2$ = gearbox output torque (NM) $\alpha 2$ = gearbox output acceleration (rads/s)

GR = gear box ratio

In terms of the system already described, $T1$ can be considered to be the torque Tm which, is seen by the motor.

$$Tm = Tl + Tal \qquad (8.13)$$

where

Tl = torque seen by the motor which is used to accelerate the load against gravity (NM)

Tal = torque seen by the motor which is used to accelerate the system inertias (NM)

Applying the relationships in equation 8.12

$$Tl = Tl2/\mathrm{GR} \tag{8.14}$$

where Tl2 = torque due to the load seen at T2 (NM)

Taking into account the affect of the inertial loads

$$Tal = Tgbj + Tj2/\mathrm{GR} \tag{8.15}$$

where

Tgbj = torque used to accelerate the gearbox inertia on the driver side (NM)

Tj2 = torque used to accelerate the sum of the system inertias at T2 (NM)

By applying the common identity $T = J\alpha$ (Klafter[89]) to equation 8.15 gives.

$$Tal = Jgb\ \alpha 1 + Jl2\ \alpha 2/\mathrm{GR} \tag{8.16}$$

where

Jgb = the gearbox inertia on the driven side (NMS^2)

Jl2 = sum of the system inertias at the output side of the gearbox (NMS^2)

combining the relationships from equation 12 to equation 16 gives

$$Tal = \alpha 1(Jgb + Jl2/\mathrm{GR}^2) \tag{8.17}$$

The analysis so far has assumed an ideal system with an efficiency of 100 %.

Efficiency(η) is a measure of output power to input power and it is related to torque in such a way that.

$$\eta = T2/(T1 * \mathrm{GR}) \tag{8.18}$$

Equation 18 combined with equations 8.17 ,8.14 and 8.13 gives

$$Tm = Tl2/(\eta \mathrm{GR}) + \alpha 1(Jgb + Jl2/(\eta \mathrm{GR}^2)) \tag{8.19}$$

The torque load, *Tl* seen by the motor , is given by the first half of equation 8.19, so that.

$$Tl = Tl2/(\eta \mathrm{GR}) \tag{8.21}$$

The inertia of the system *Jl,* seen by the motor, is given by the second half of equation 19. So that

$$Jl = Jgb + Jl2/(\eta \mathrm{GR}^2) \tag{8.22}$$

The angular acceleration, $\alpha 1$is ignored at this stage because it is found from within the motor CLTF.

The angular acceleration $\alpha 2$ is found by transposing the relationships of equation 8.12 to give.

$$\alpha 2 = \alpha 1 / GR \tag{8.23}$$

The efficiency η used in equations 8.19, 8.20 and 8.21, is the product of the efficiencies of other components and the gearbox efficiency, so that.

$$\eta \text{ or } Etot = Gbe * E2 \tag{8.24}$$

where

Gbe = gear box efficiency

E2 = total of the efficiencies of components on the output of side of the gearbox.

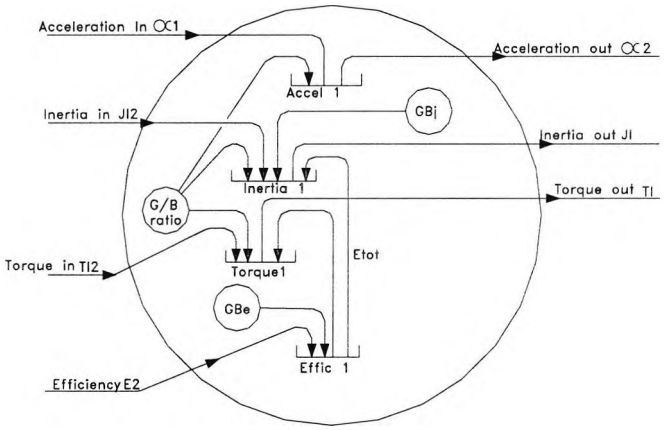


Figure 8-9 ACP diagram for Motor

The ACP diagram for the motor is shown in figure 8.9, and its behaviour is captured below in psuedo code:

ACTIVITY: Gearbox (INPUTS: $\alpha 1(s)$, TI2, JI2, E2
OUTPUTS: $\alpha 2(s)$, TI, JI)

POOL: Accel 1 (INPUTS: $\alpha 1(s)$,
OUTPUTS: $\alpha 2(s)$
VARIABLES: GR)
BEHAVIOUR{ $\alpha 2(s) = \alpha 1(s)/GR$}

POOL: Inertia 1 (INPUTS: *JI2*, Etot
OUTPUTS: *JI*
VARIABLES: GR, *Jgb*)
BEHAVIOUR{ $JI = Jgb + JI2 /(Etot * GR^2)$}

POOL: Torque 1 (INPUTS: *Tl2*, Etot
OUTPUTS: *Tl*
VARIABLES: GR)
BEHAVIOUR{ *Tl*= *Tl2*/(Etot* GR)}

POOL: Effic 1 (INPUTS: E2
OUTPUTS: Etot
VARIABLES: Gbe)
BEHAVIOUR{ Etot = Gbe * E2 }

8.3.3.3 Pulley and belt behaviour

The pulley and belt behaviour is based around 2 activities and 6 pools (see figure 8.10) which encapsulate, the increase in inertia, the alteration of torque to force, changes in efficiency and the conversion of angular acceleration to linear acceleration.

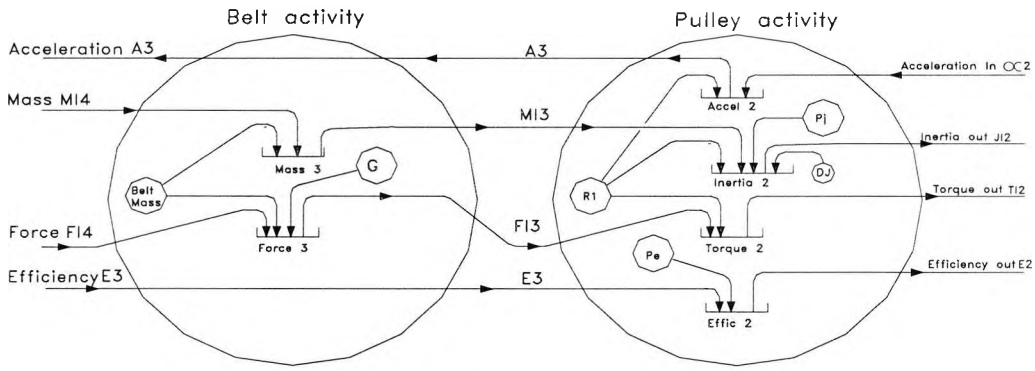


Figure 8-10 ACP diagram for Gear box

Acceleration is converted from angular to linear using the relationship

$$a = \alpha * r \qquad \text{(Hannah[89])} \qquad (8.25)$$

where,

a = linear acceleration(M/S^2) , α = angular acceleration($rads/S^2$) , r = pulley radius (M)

Force is derived from the relationship

$$F = T/r \qquad \text{(Hannah[89])} \qquad (8.26)$$

where,

F = force (N), T = torque (NM), r = pulley radius (M)

The inertia of a load acting at distance r, from a centre line is given by

$$J = M * r^2 \qquad \text{(Hannah[89])} \qquad (8.27)$$

The increase in force required to lift a load due to gravity is derived from the relationship

$$F = mg \sin\theta \quad \text{(Hannah[89])} \tag{8.28}$$

Where

m = mass of the object (Kg), g = acceleration due to gravity (9.81 M/S^2),

θ = angle, between the line of movement of the load and the ground (radians),

These equations are incorporated into pools shown in figure 8.10 in the following manner:

ACTIVITY: Pulley (INPUTS: *A3(s),Tl2,Jl2,E2*
OUTPUTS: $\alpha 2(s)$, *Fl3,Ml3,E3*)

POOL: Accel 2 (INPUTS: *A3(s)*
OUTPUTS: $\alpha 2(s)$
VARIABLES: *R1*= pulley radius)
BEHAVIOUR{ $\alpha 2(s) = A3(s)/R1$ }

POOL: Torque 2 (INPUTS: *Fl3*
OUTPUTS: *Tl2*
VARIABLES: *R1*)
BEHAVIOUR{ *Tl2= Fl3 * R1*}

POOL: Inertia 2 (INPUTS: *Ml3*
OUTPUTS: *Jl2*
VARIABLES: *DJ* = drive shaft inertia, *R1,PJ* = pulley inertia)
BEHAVIOUR{ $Jl2 = Ml3*R1^2 + DJ + PJ$}

POOL: Effic 2 (INPUTS: *Etot 3*
OUTPUTS: *Etot 2*
VARIABLES: *Pe*)
BEHAVIOUR{ *Etot 2 = Etot 3 * Pe*}

ACTIVITY: Belt (INPUTS: *A3(s), Etot 3, Ml4, Fl4*
OUTPUTS: *A3(s), Etot 3, Ml3, Fl3*)

POOL: Force 3 (INPUTS: *Fl4*
OUTPUTS: *Fl3*
VARIABLES: *Belt mass, g= 9.81,* θ)
BEHAVIOUR{ $Fl3 = Fl4 + (Belt\ mass * g * \sin\theta)$}

POOL: Mass 3 (INPUTS: *Ml4*
OUTPUTS: *Ml3*
VARIABLES: *Belt mass*)
BEHAVIOUR{ *Ml3 = Ml4 + Belt mass*}

8.3.3.4 Load and Bearing behaviour

The load and bearing behaviour is also contained within two separate activity diagrams. The load activity contains the mass and force, (dependant upon the angle θ) of the load and the calculation for first and second order integrals of the acceleration. The bearing activity contains the bearing efficiency. The ACP diagrams for these activities are seen in figure 8.11.

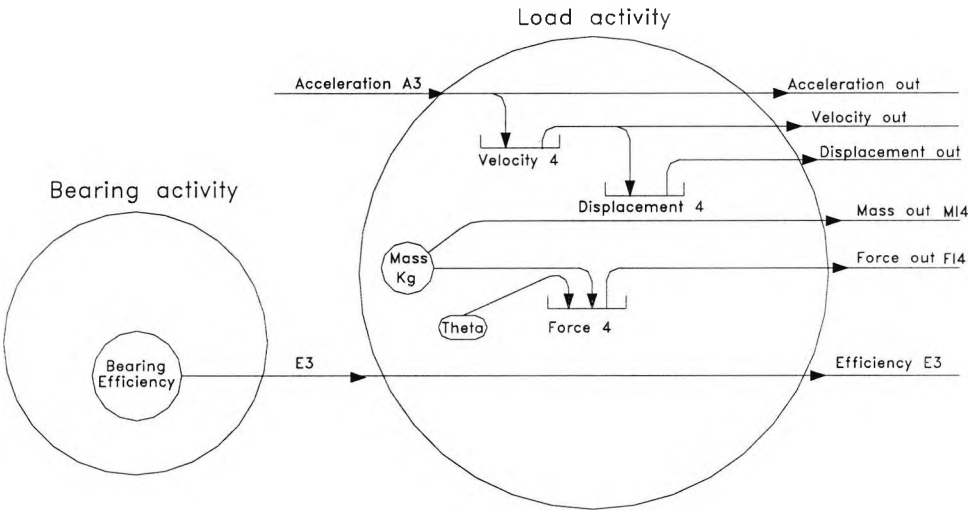


Figure 8-11 ACP diagram for load and bearing

The behaviour of this part of the system is described by

ACTIVITY: Load (INPUTS: *E3,A3*(s)
OUTPUTS:*E3, Tl4, Ml4*, Acceleration(s), Velocity(s), Displacement(s)
BEHAVIOUR { Acceleration(s) = *A3*(s) }

POOL: Velocity 4 (INPUTS: *A3*(s)
OUTPUTS: Velocity
VARIABLES:)
BEHAVIOUR{ Velocity(s) = *A3*(s)/S }

POOL: Displacement 4 (INPUTS: Velocity(s)
OUTPUTS: Displacement(s)
VARIABLES:)
BEHAVIOUR{ Displacement(s) = Velocity(s)/S }

POOL: Mass 4 (INPUTS:
OUTPUTS: *Ml4*
VARIABLES: *Ml4*)
BEHAVIOUR{ }

POOL: Force 4 (INPUTS:
OUTPUTS: *Fl4*
VARIABLES: g, θ, *Ml4*)
BEHAVIOUR{ $Fl4 = Ml4 * g * sin\theta$}

ACTIVITY:Bearing (INPUTS:
OUTPUTS:Be = bearing efficiency)

The ACP diagrams and behavioural descriptions of the system provide the basis for the computer model of the system.

8.4 Development of the VISSIM model for component selection

The concept behind using dynamic software models for component selection is to create a software model of the system in which all the important variables are identified. Combinations of components, which are represented as values for the variables, are applied one at a time to the model in a systematic manner. An input signal or operating procedure is applied to the model for each separate combination of variables, various outputs from the system are then compared to the system specification. If the combination passes the specification it is logged in a file which contains all the successful combinations.

From this brief description five separate parts to the process can be identified:-

- The creation of the component databases
- The creation of the component combination file
- The creation of the system model,
- Combination selection and logging
- The creation of the test input signal

The first four of these process are discussed in this section of the chapter, the fifth, creation of the test input signal will be discussed in section 8.5 .

8.4.1 Creating the component databases

The initial choice of which, motor, gearbox and pulley manufacturers to select, was based upon selecting ranges of components which were economical and the sales literature contained values for the all relevant variables required to model the system accurately. The component specifications can be seen in tables 8.2 to 8.4.

Serial number	ID	Tcont NM	Tpeak Nm	Vnom rads/S	Pnom KW	Jm Kg m^2	Acc Max rads/S^2	KT Nm/ A	KE V/rad/S	Ra Ω	La H	Irms A	Imax A
FAS TO M4 030	0	1.3	5.5	314	0.31	90e-6	54.4e3	0.7	0.337	21.6	51.3e-3	6	8.5
FAS TO M4 060	1	1.3	5.5	628	0.62	90e-6	61.1e3	0.39	0.227	6.05	16e-3	6	8.5
FAS TO M8 030	2	2.4	9	314	0.58	140e-6	39.3e3	0.78	0.486	8.9	28.3e-3	12	18
FAS TO M6 060	3	2.3	9	628	0.99	140e-6	41.4e3	0.41	0.260	2.82	7.85e-3	12	18
FAS T1 M2 030	4	2.7	9	314	0.61	280e-6	32.1e3	0.77	0.517	5.6	23e-3	12	18
FAS T1 M2 060	5	2.7	9	628	0.98	280e-6	20.4e3	0.39	0.272	1.45	6.2e-3	12	18

Table 8-2 Motor specifications for Vickers FAS T range

Serial No	ID	D1 M	D2 M	W1 M	W2 M	Density Kg/M^3
30-5M-09	0	0.0475	0.035	0.008	0.0145	7000
32-5M-09	1	0.0509	0.038	0.008	0.0145	7000
36-5M-09	2	0.0573	0.038	0.008	0.0145	7000
40-5M-09	3	0.0636	0.038	0.008	0.0145	7000
44-5M-09	4	0.0700	0.038	0.011	0.0145	2700
48-5M-09	5	0.0764	0.038	0.011	0.0145	2700

Table 8-3 Pulley specifications

Serial No.	ID	GR	JGb Kg M^2	Gbe	Gmax rads/S
GPL 90	0	12.25	0.16e-3	0.82	314
GPL 90	1	14	0.16e-3	0.82	314
GPL 90	2	16	0.16e-3	0.82	314
GPL 90	3	20	0.16e-3	0.82	314
GPL 90	4	24	0.16e-3	0.82	314
GPL 90	5	25	0.16e-3	0.82	314

Table 8-4 Stegman Planetary gear box specifications

The manufacturers selected were

- Stegman: Planetary gearboxes
- Vickers: FAS T range of brushless motors
- Davall: Toothed pulleys and belts

A basic analysis for torque and velocity requirements was carried out to ensure the right range of components were selected. In this instance 6 motors, 6 gearboxes and 6 pulleys were selected.

The ID is the number used to identify the component within the VISSIM model, for each set of components. For the motor, no value of back EMF constant, KE ,was given so it was derived using the following formula:

$$KE = \frac{180}{Vnom(((180/Ra)\ Kt)/\ (((180/Ra)\ Kt)-Tcont))} \qquad (8.28)$$

This formula is based upon finding the maximum theoretical torque which can be developed at zero velocity and regressing it back to determine the maximum velocity under no load at 180 volts, effectively calculating the gradient of the extended torque velocity curve.

There is no value for the coefficient of viscous friction Bm for the motor, firstly it is generally accepted that for brushless motors this value is very small and can be ignored (Electro-craft 77, Kenjo 93) and secondly no value for Bm was given in the motor specification.

There is no value for the inertia of the pulley because it can be derived from its dimensions and density using the formula from Hannah (89).

$$J = m\ R^2/\ 2 \qquad (8.30)$$

where

J = Inertia (KgM^2), m = mass (Kg), R = D/2 = radius of a rotating disc (M).

The pulley dimensions are shown in figure 8.12.

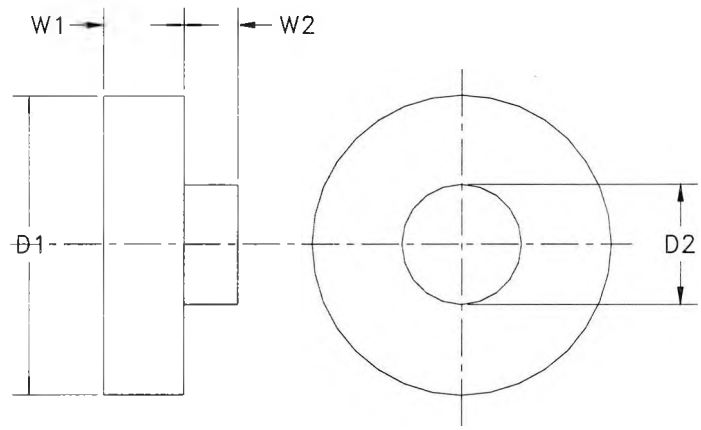


Figure 8-12 Pulley dimension notation

The information contained in tables 8-2 to 8.4 was written to a number of text files with a .MAP extension. A MAP file is a VISSIM file which allows certain information to be output depending upon an input. For example, if a 1 was input into the .MAP file containing the pulley information, all the data relating to pulley type, 32-5M-09, would be retrieved, providing the proper output variables are set in place.

8.4.2 Creating the component combination file

The component combination file contains every combination of motor, gearbox and pulley that are to be considered for the system. In this instance, 6 motors, 6 gearboxes and 6 pulleys were selected, this gives 216 different combinations (6^3=216). It also, is represented as a .MAP file.

The file is created by employing three ramp signals with gradients of 0.0277 (6/216), 0.1666 (6/36) and 1 (6/6) respectively. The ramps are produced using RESET INTEGRATOR blocks which are set back to an initial of zero whenever the output reaches 6. The output from each ramp signal is quantisized using an INT block that rounds the input down to the nearest integer. The output signal and block diagram for the MOTOR ID can be seen in figure 8-13. The sloped line indicates the normal output of the integrator block whilst the stepped, indicates the quantisized output.

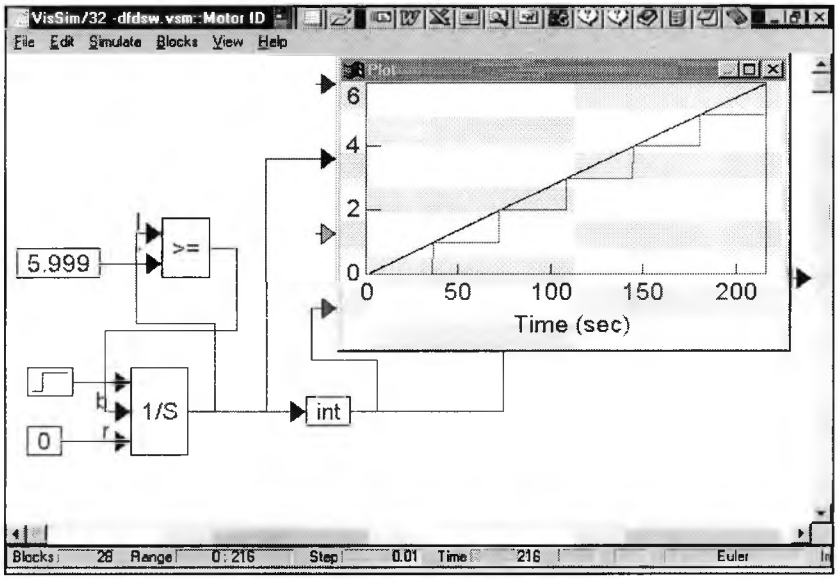


Figure 8-13 Effect of INT block on a ramped signal

Figure 8.14 shows the output of the MOTOR ID and the GEARBOX ID. From the plot shown in the figure it can be seen that all combinations of motor and gearbox are displayed, the long steps indicate the MOTOR ID whilst the series of 6 shorter steps indicate the GEARBOX ID. The pulley combinations are not displayed on this plot because it makes the plot unreadable. This information is logged at one second intervals for 216 seconds (virtual time), to create the file SWEEP.MAP which contains all the component combinations.

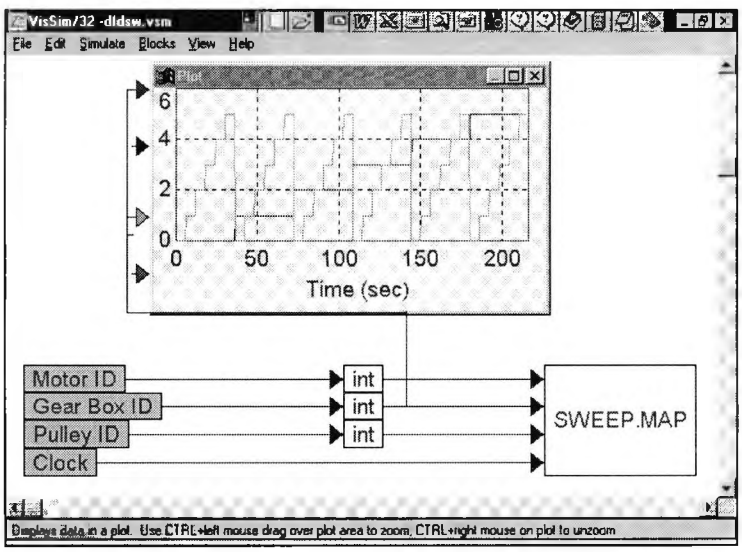


Figure 8-14 Combination logging to SWEEP.MAP

A section of text from the file SWEEP.MAP can seen in figure 8-15. The first column gives the combination ID, the second, third and fourth columns show the motor, gearbox and pulley ID numbers respectively. The Fifth and final column again shows the combination ID.

The reason for this repetition is that the ID in the first column is only used as a pointer to a particular row and is lost when the successful combinations are logged to a file, this ensures that the identity of the combination is kept. The various combinations are selected by applying a quantisized ramp signal to the .MAP file input. The output of the SWEEP.MAP file is input into the corresponding database MAP files containing the component information. The information contained in these files is then input into the component model.

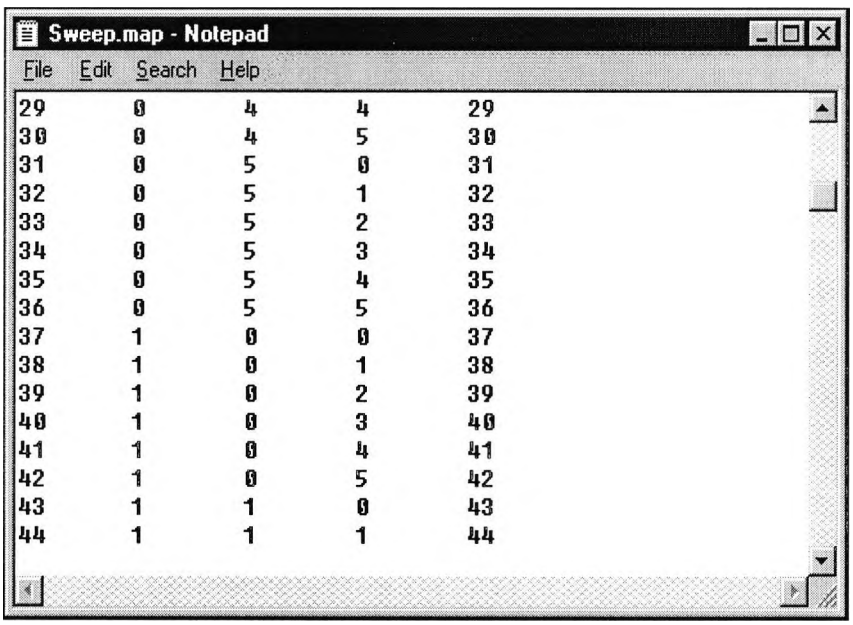


Figure 8-15 Section of SWEEP.MAP

8.4.3 Creating the VISSIM model of the drive system

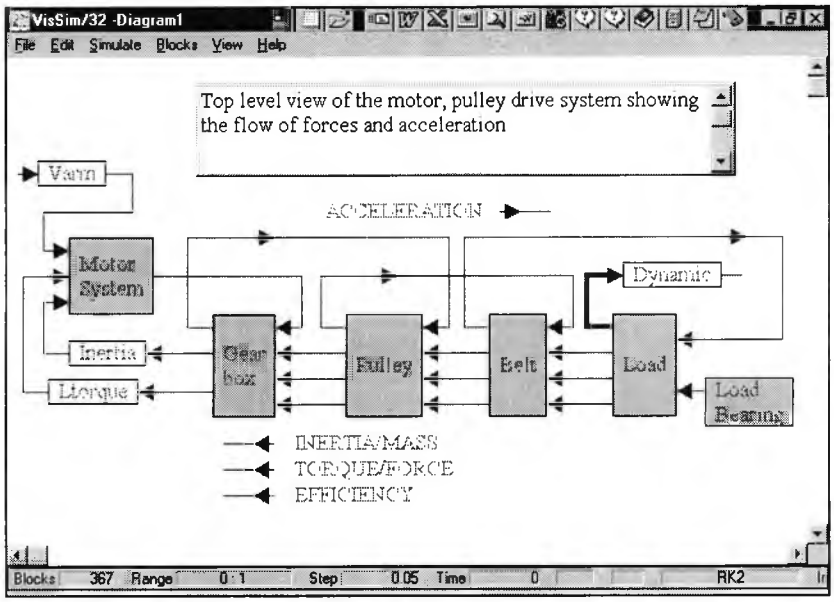


Figure 8-16 VISSIM model of drive system

The VISSIM model of the drive system (see figure 8-16) is very similar to the ACP diagram shown in figure 8-2. It is comprised of 6 major blocks which relate directly to the Activities and their behaviour, detailed in section 8.3.3. The input signal to the system is Varm, which is the input voltage. The output of the system is the loads, displacement velocity and acceleration, this information is contained in the Vector block called DYNAMIC. As in the ACP diagram of figure 8.3, the motor acceleration is passed from left to right, whilst the torque/force, inertia/mass, and efficiency are passed from right to left, applying load to the motor. Each of the major blocks in figure 8-16 will be expanded upon, and the behaviour described with respect to the representation of the Activities descriptions of section 8.3.3, within the VISSIM model.

8.4.3.1 Motor model

The motor model is broken down into a number of sub blocks (see figure 8-17), the arrangement of these blocks being very similar to the motor block diagram in figure 8-5. The combination of blocks, encapsulates the behaviour of the motor model described by equation 8.11.

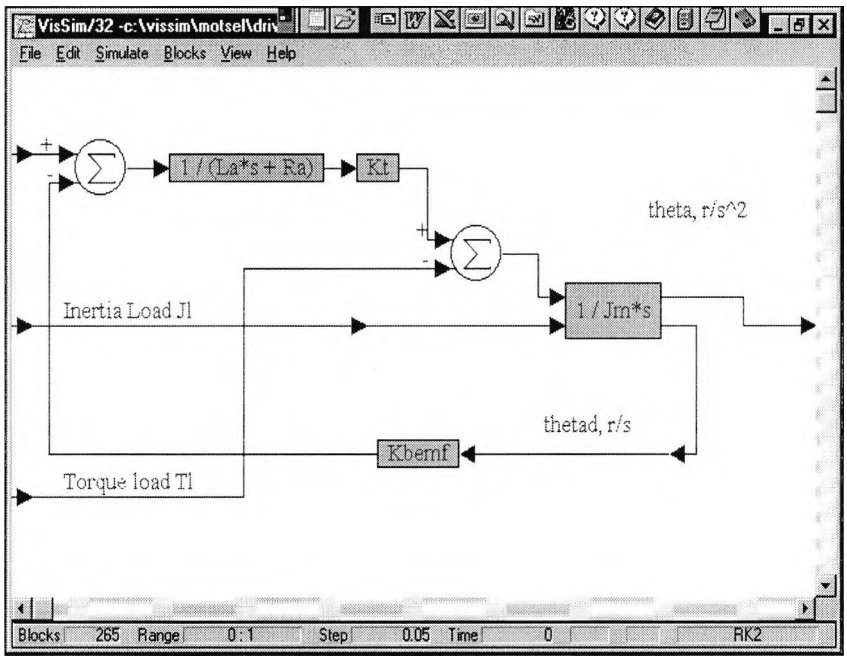


Figure 8-17 VISSIM model of an electric motor

The armature transfer function is shown in figure 8-18. If block diagram reduction techniques are applied to the model, it is seen that it represents the equation shown on the first block label, from the left, in figure 8.18. Variables of Ra and La (Armature resistance and Inductance) are entered via GLOBAL VARIABLE blocks (indicated by the labels Ra and La). The integrator block, signified by 1/S, is a RESET INTEGRATOR block with an input variable RESP, this stands for reset spike. Whenever this variable is high, the integrator is set back to zero. This functionality is required when a combination has been tested and the contents of the integrator have to be cleared for the next combination. The RESP variable is used throughout the model for this reason and it will not be discussed further.

Two other blocks are in common usage throughout the model, the LIMIT block and the ABS PEAK block, both were created using standard VISSIM elements and logic. The

LIMIT block, is used to limit the current drawn by the motor to the maximum limit of the power supply which is being used (Imax). The ABS PEAK block is employed to log the absolute peak value of a variable. It is used in this case to determine the peak current drawn by the motor, for observation purposes.

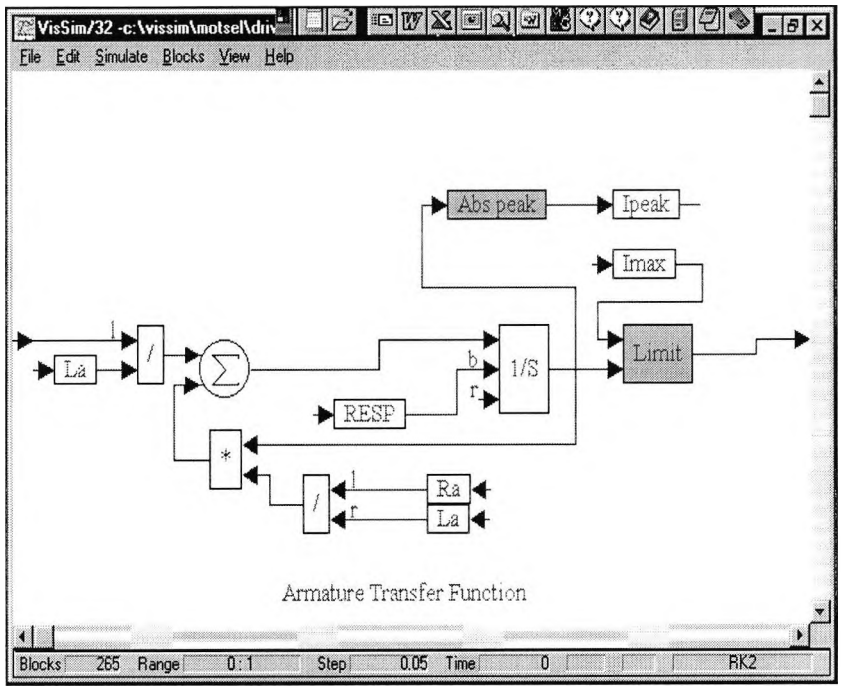


Figure 8-18 Electrical transfer function (1/(LaS + Ra))

The mechanical transfer function of the motor is contained within the third block in the forward direction. In this instance, the coefficient of Viscous friction (Bm) is ignored and combining the definition of *Jt* with equation 8.8 the mechanical transfer function is reduced to

$$\omega(s)/Tg(s) = 1/(Jl + Jm)S \qquad (8.31)$$

The VISSIM representation of this equation is shown in figure 8.19. Again a limit is put upon the output, in this case acceleration. The integrator changes the system output from acceleration to velocity, so that it can be fed back to the motor as back EMF.

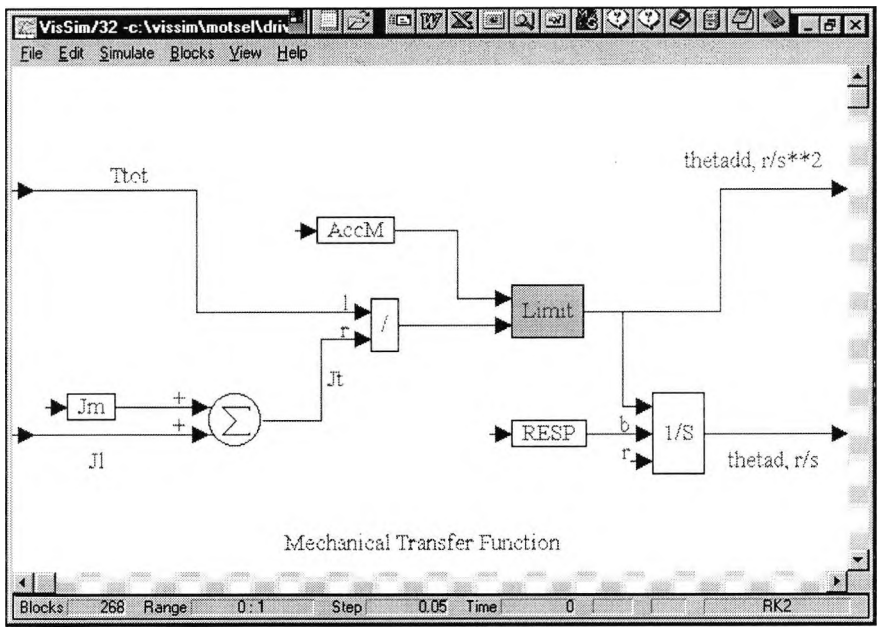


Figure 8-19 Mechanical transfer function (1/(Jl + JmS))

The Kbemf and Kt blocks, within figure 8-17, contain a multiplier and the appropriate variable label (Kt and Ke) for the block, which act as system gains.

8.4.3.2 Gear box and Pulley model

The gearbox model and pulley models (see figures 8-20 and 8.21) are identical to their ACP representation except the equations that describe the behaviour in their various Pools are not contained within compound blocks which would represent pools. This is because a large number of separate diagrams would be required to show the VISSIM representation of each Pool. To aid the reader in identifying the equations that represent each Pool, the name of the Pool is above or below its mathematical model this convention is used throughout this model, unless a diagram is too complex, in which case it is functionally decomposed, as in the Motor Activity.

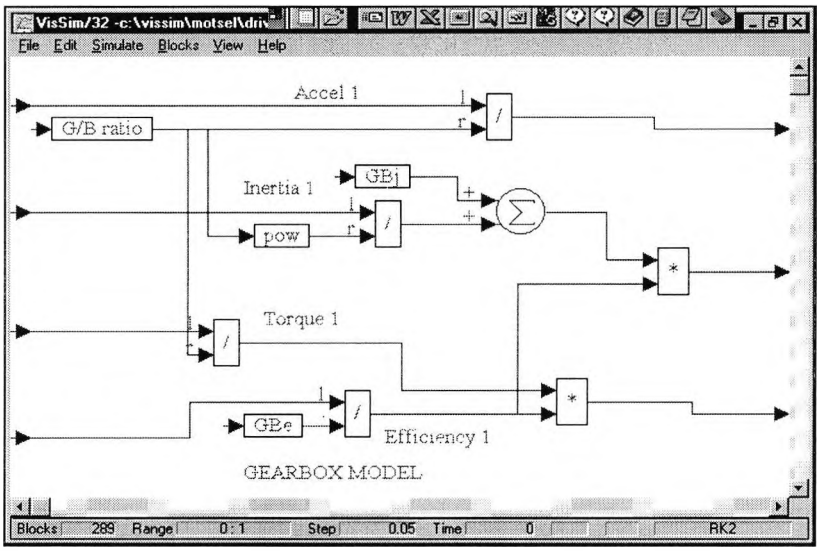


Figure 8-20 Gearbox model

Within the gearbox and pulley models, the signals sent along each line indicate acceleration, inertia, torque and efficiency, from top to bottom respectively.

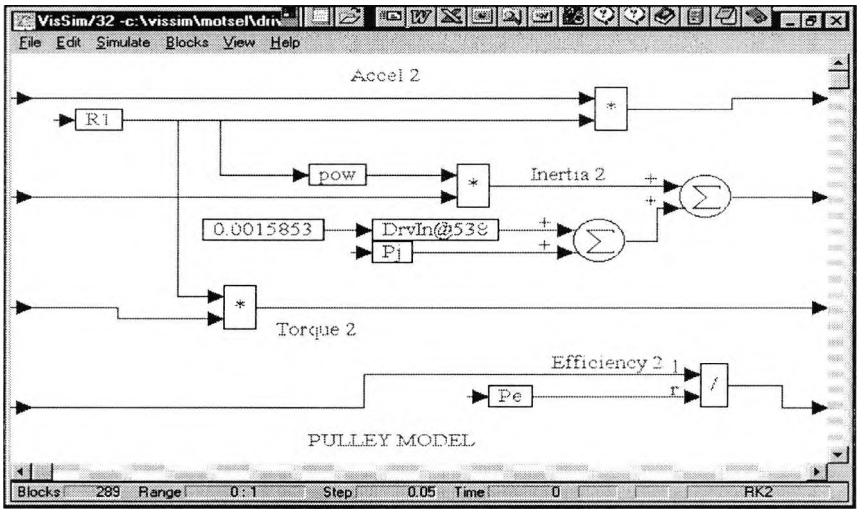


Figure 8-21 Pulley model

8.4.3.3 Belt, Load and Pulley model

The belt, load and bearing models are also very similar to their ACP diagrams, as can be seen in figures 8-22 and 8-23 which show the VISSIM models for the Belt and Load. The bearing model is not shown as it is just contains a constant value which represents the bearing efficiency.

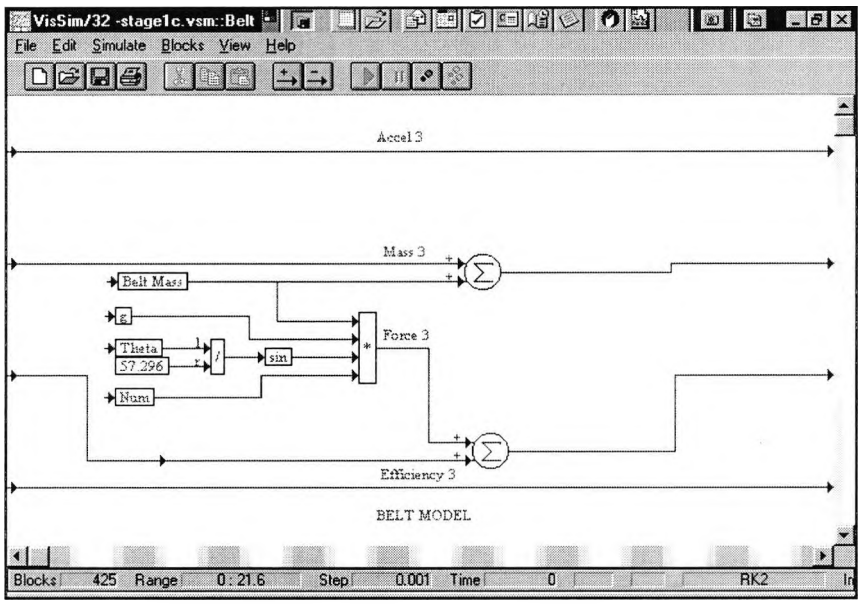


Figure 8-22 Belt model

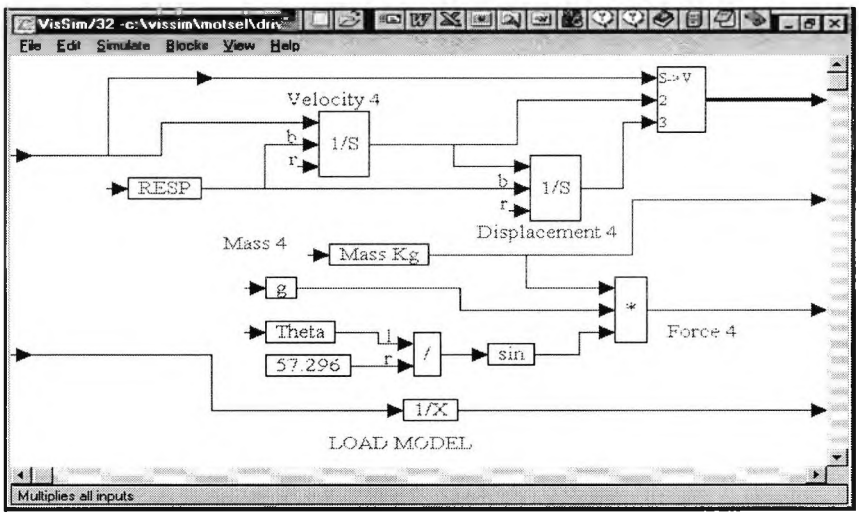


Figure 8-23 Load model

8.4.4 Combination selection and logging

Combination selection and logging is carried out using a series of 4 major blocks.

- OUTPUT ANALYSIS block: - contains the mathematical equations used to derive values such as power, maximum torque, and RMS torque. These values are input into variables and fed into the DECISION block.
- DECISION block: - Compares the values obtained from the OUTPUT ANALYSIS block with the specification variables. If all the variables from the Output Analysis are less than the specification block, a signal is created which is passed to the DATA FILTER. If any of the specification is not met, the combination ID and the values of the OUTPUT ANALYSIS block are logged into a .DAT file that contains all failure information.
- DATA FILTER: - The signal from the DECISION block for a successful combination is used to put the successful combination information on hold, until it is ready to read the input.
- COMBINATION FILE: - This is a .MAP file which logs all the successful outcomes to the test input.

Each of these will now be discussed in turn.

8.4.4.1 Output analysis

The function of the OUTPUT ANALYSIS block (see figure 8-24), as mentioned previously, is to derive values which can be compared against the specification. The values required are derived from the motor acceleration, motor torque and the velocity and displacement of the load. The output variables derived from these inputs are:-

- Torque absolute peak (Tpeak)
- Torque Root Mean Square, (Trms) value, which can be taken as the nominal value for the torque.
- Angular acceleration absolute peak (Apeak).
- Angular velocity absolute peak (Vpeak), limited by the maximum angular velocity of the gearbox to 314 rads/s.
- Linear velocity of the load, absolute peak (Vlpeak).

- Linear displacement of the load, (ActDistance).
- RMS power of the drive (Prms), derived from torque and velocity, it gives the nominal power output.

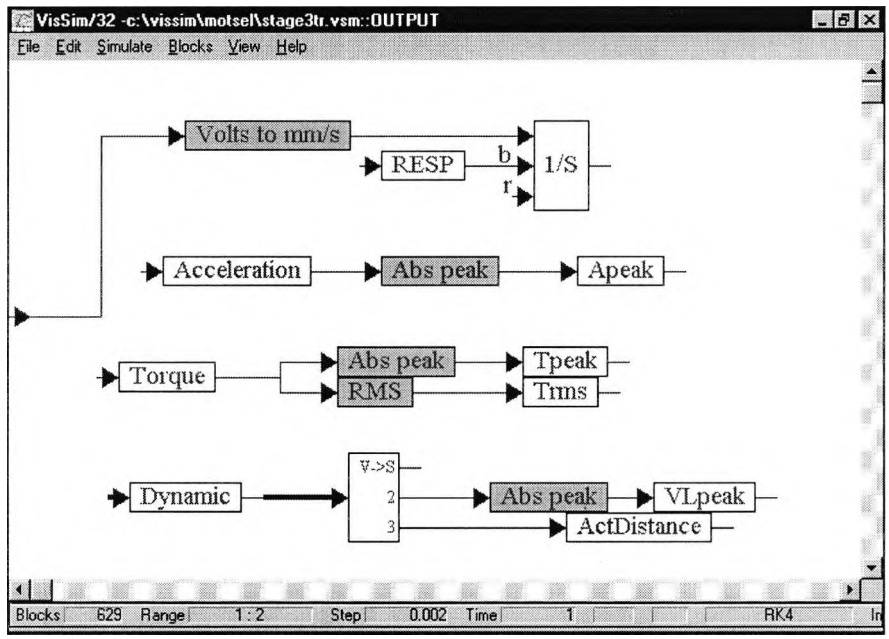


Figure 8-24 OUTPUT ANALYSIS block

The block "VOLTS TO MM/S", is used to convert the motor input voltage from volts to mm/s, this signal is then integrated to give displacement. Its major function is testing of the input signal.

8.4.4.2 Decision and Data Filter Blocks

The Decision block takes the values generated by the Output Analysis block and compares the values to the motor, gearbox and pulley specifications contained in the component input file for a specific component. This process is shown in figure 8-25.

The Output Analysis variables are compared to the specification values using a boolean, LESS THAN OR EQUAL block. This ensures that when the variable exceeds the value set by component specification, the output line from the block goes low, this indicates a failure. The output from all the comparison blocks are compared in an AND block, whenever a failure is detected the output from this block goes low. A variable known as Res, is used to hold all the drive data at every second with a delay of - 0.05 seconds, this

drive data is sampled at 1-second intervals so that it can be logged. The main function of this file is to make it possible to examine why certain combinations failed.

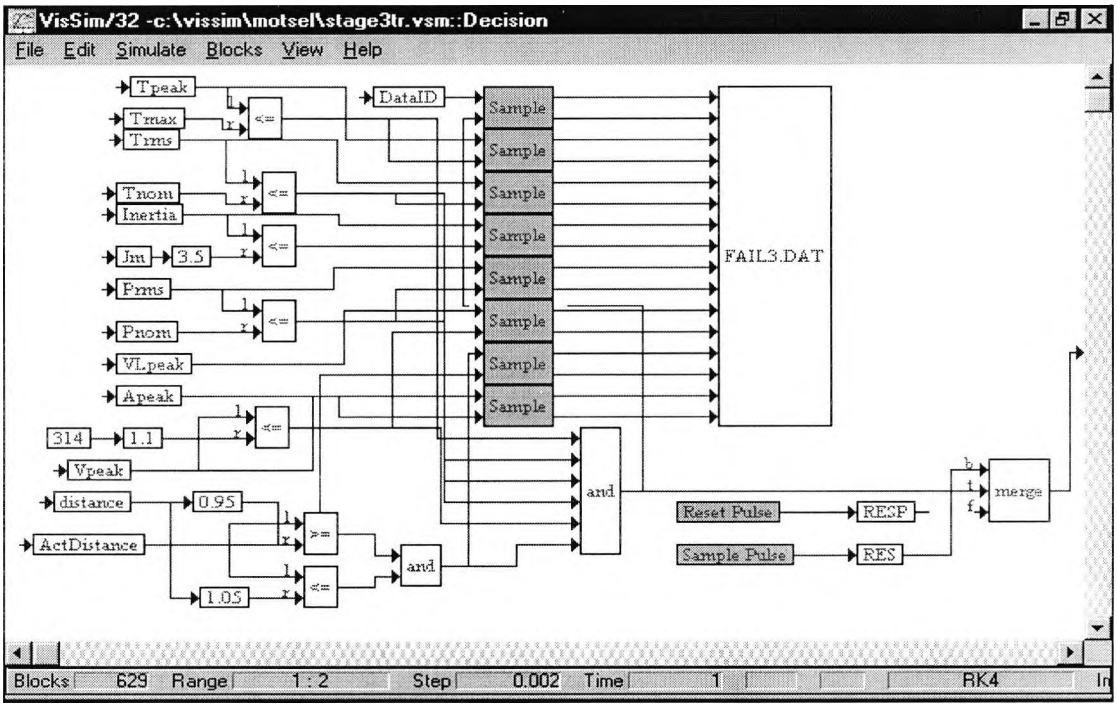


Figure 8-25 DECISION and DATA FILTER blocks

If none of the specifications are exceeded, a signal is output to the Data Filter block which latches the data in the same way the output variable data is logged, except in this instance only successful combinations are logged.

8.4.4.3 Combination File

The Combination file is set up as .MAP file because it is used for successive selection criteria. It contains an ID counter, which is used for selection purposes later on, the combination data, (motor ID, Gearbox ID and Pulley ID), followed by the combinations success ID no and its original ID number. A section of typical output from a selection combination file can be seen in figure 8-26. From this figure it can be seen that combinations 1 to 4 , 7 to 12, and 14 to 18, are acceptable, whilst combinations 5, 6, 13, and 19-21, etc. were not. Successful combination 10 consists of Motor 0, Gearbox 1 and Pulley 5 and relates to combination 12 in the original data file, see appendix B.

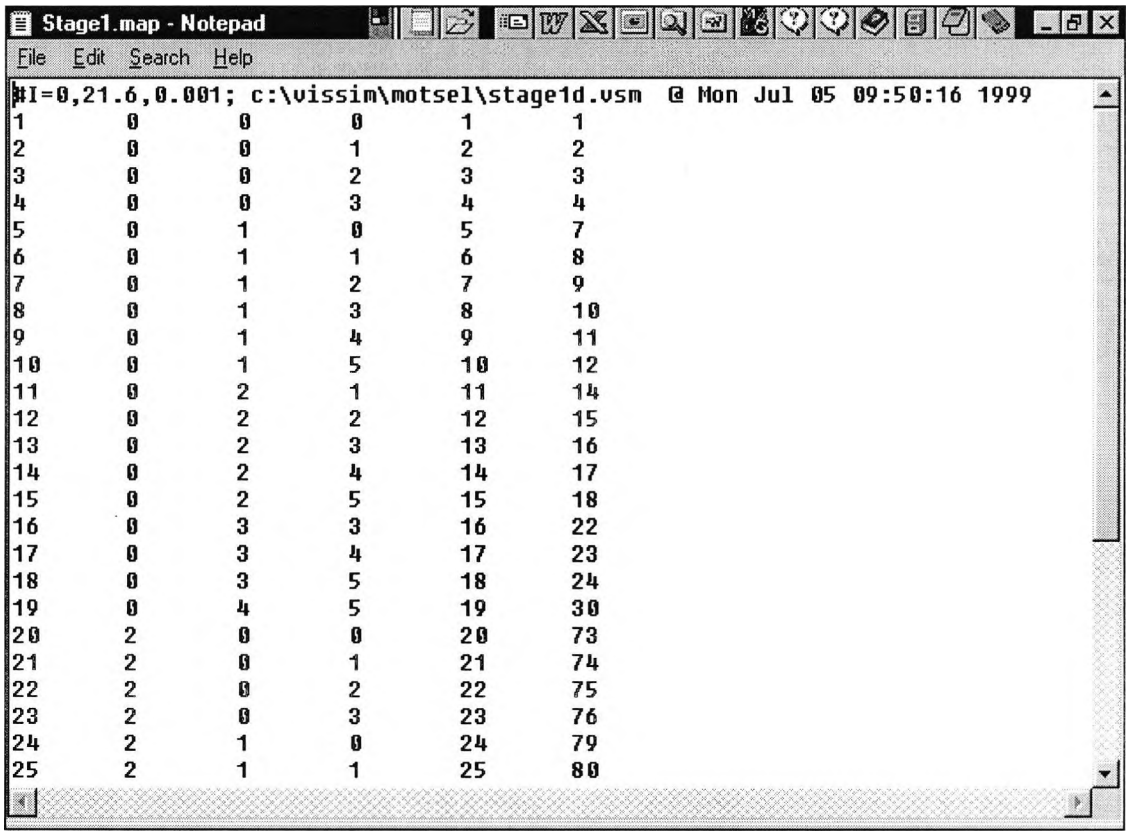


Figure 8-26 Section from a successful combination file

8.5 Computerised component selection

An example of the complete VISSIM diagram for component selection can be seen in figure 8-27. This diagram contains a number of blocks which have not been discussed yet. The most important part is the input signal, in this instance, 180 VOLT STEP INPUT block, this block is discussed in section 8.5.1.

Environment details are entered in the blocks in the top left-hand corner. These cover details such as the number of pulleys, angle of the drive system between the drive system and the ground, mass of the load and the expected simulation time.

In the area of the SWEEP.DAT and STAGE1.MAP files, DISPLAY blocks are used to show the combination being tested, the combination ID number and the number of successful combinations.

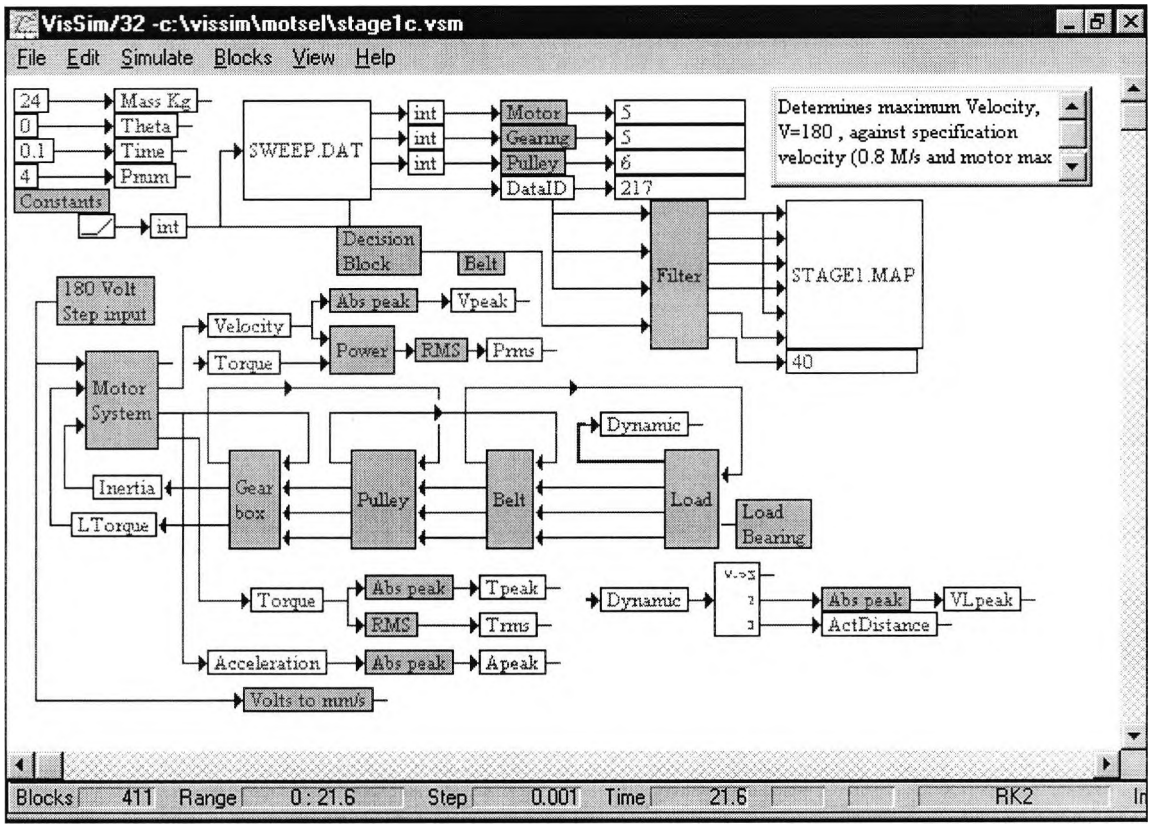


Figure 8-27 VISSIM model for computerised component selection

8.5.1 Input conditions

The input conditions, used to test whether the system reaches its test specification, are separated into 3 stages. Initially tests were combined and applied to every combination. Because this exercise was originally carried out on a 40 MHz, 386 PC, it took approximately 10 hours to complete, therefore, a method of improving the search of the design space was required. It was decided to stage the tests and remove any combination, which failed that particular stage. This reduced the analysis time from 10 hours to approximately an hour when using a fast 386 machine and approximately 10 minutes when using a 75 MHz, 486.

The first stage is a fast ramp input voltage, which rises from 0 volts to the nominal voltage of 180 volts in 0.02 seconds. This signal is used to test whether the load can reach the required speed of 500 mm/s with the torque limitations of the motor. A ramp is applied because if a step input of 180 volts is used no solutions are found. This is due to the very

high acceleration which produces a torque far in excess of the capabilities of any of the motors under consideration.

The second stage, is a triangular velocity curve (250 mm displacement in 1.98 seconds). A triangular velocity move is based upon a constant positive acceleration for the first half of the move distance, followed by negative acceleration for the second half of the move. This test is used to examine whether the load can be accelerated at the required rate of 250 mm/s^2 without exceeding power and torque limitations. The final position of the load is compared to its target position of 250 mm, this is used as a further selection criteria.

The final stage of the analysis is based upon a trapezoidal velocity load move of 500 mm in 1 second. A trapezoidal velocity move divides the velocity curve into three phases, a constant acceleration for a third of the move time, constant velocity for the second third, and constant deceleration for the final third. This test was chosen because it is a standard move used in robotics and reflects a truer representation of the requirement to move the load 500 mm in 1 second.

Although the load acceleration and velocities are constant for each combination, the motor control signal differs, as it is dependent upon the gearbox ratio and pulley diameters. If the required motor accelerations and velocities cannot be reached or exceed the motor specification the combination fails.

8.5.2 Results

The above analysis was carried out for the X and Y axes of the robot, each axis being analysed separately. The Y axis differs from the X axis in two ways. Firstly the Y axis is carrying a constant load against gravity and therefore has torque load requirements dependant upon the axis angular orientation with the ground. And secondly, only two pulleys and one belt are used, this affects the inertia seen by the motor.

8.5.2.1 Component Selection results

A summary of the number of combinations examined and those that were successful for each stage is given in table 8.5. Copies of the FAIL (containing failure information) and STAGE (containing successful combinations) files can be seen in Appendix B

Stage No. and axis ID	Number of variations tested	Number of successful variations	Reasons for failure deduced from the appropriate failure file.
1 X	216	37	Combination of Torque requirements not met and load velocity is not between 0.5-1 M/s when the gearbox velocity exceeds 314 rad/s (3000 RPM), FAIL1X.DAT.
2 X	37	18	Inertia matching criteria exceeded FAIL2X.DAT.
3 X	18	2	Combination of gearbox velocity and load displacement criteria not met, FAIL3X.DAT .
1 Y	216	36	Combination of Torque requirements not met and load velocity is not between 0.5-1 M/s when the gearbox velocity exceeds 314 rad/s (3000 RPM), FAIL1Y.DAT.
2 Y	36	25	Inertia matching criteria exceeded, FAIL2Y.DAT
3 Y	25	2	Combination of gearbox velocity and load displacement criteria not met, FAIL3Y.DAT .

Table 8-5 VISSIM model for computerised component selection

The successful component combinations for the stage 1 analysis, for both the X and Y-axis were identical apart from combination 12, which failed in Y-axis analysis. This is because it produced a load velocity in excess of 1 M/s. The successful combinations were based upon the FAS T 0 M4 030 (0) and FAS T 0 M8 030 (2), motors.

The stage 2 analysis produced differing results, with the X-axis analysis producing 18 successful variations whilst the Y-axis produced 25. The Y axis analysis included 6 combinations based upon the FAS T 0 M4 030 (0) motor. Upon comparison of the FAILX2.Dat and FAILY2.Dat files, it was noted that the load torque on the Y axis was offset by the lower inertia of the load, giving a lower total load than the X axis.

The third stage analysis reduced both sets of axis combinations to the same two combinations, this was due to the higher velocity required by the 0.5 M trapezoidal velocity move in 1 second. Linear velocities of approximately 0.7 - 0.8 M/s were observed with angular velocities ranging from 302 rad/s to 517 rad/s. Any motor angular velocity in excess of 314 rad/s exceeded the gearbox angular velocity limitations.

The successful combinations equated to physical components can be seen in table 8.6.

Combination code	Motor	Gearbox	Pulley
2 0 3	FAS T 0 M8 03	GPL 90 12.25:1 gear box ratio	40-5M-09 63.66 mm outer dia
2 1 4	FAS T 0 M8 03	GPL 90 14:1 gear box ratio	44-5M-09 70.03 mm outer dia

Table 8-6 VISSIM mode for computerised component selection

The motor / gearbox / pulley combination finally selected for both axes was the 2 0 3 combination, this was due to the slight increase in cost of the 14:1 gear box compared to the 12.25:1 gearbox.

8.6 Summary and conclusions

This chapter has covered the embodiment design of the drive aspects of the Gamma radiography robot. This process involved the novel application of ACP models, derived from EQFD correlation chains, and VISSIM to carry out the automated selection of components.

The first stage covered the generation of the correlation chain from within the EQFD 'House of Quality'. This was relatively trouble-free as the correlation chain is reflected in the physical relationships that exist between the components. However, it can be problematic to identify the positive or negative effects of different components upon each other. As a positive consequence of this, the correlation chain analysis forces the design engineer to consider these effects and gain an understanding of the system.

The ACP system model can be developed, directly from the correlation chain. Because of its hierarchical nature, detail can be hidden within the diagram until the necessary level is reached. As indicated in this chapter, ACP, diagrams can be considered to be a meta-modelling language, allowing different model representation systems to be employed at the lowest level of abstraction (the POOL). This is very useful, as it allows the user to employ whatever modelling system they choose or is appropriate. In this instance, pseudo code and mathematical equations were used as the low-level modelling systems. The writing of these behavioural models proved to be very laborious and time consuming and, in certain

instances (the load bearing, load and pulley Activities), unjustifiable on the basis of there simplicity alone.

The energy flows within the system were more difficult to identify, as they did not fully reflect the positive and negative relationships detailed in the correlation chains. This is because more than one form of energy flow can be travelling in different directions through the same component. A problem which could possibly be solved by the use of square correlation chains (Edney[99] , Atherton[97]), or the designer taking a different analysis approach.

The development of the VISSIM model from the ACP diagrams was straightforward, the ACP diagrams and behavioural model providing a complete paper based model of the system. The VISSIM model provided a virtual verification of the ACP model, and ensured that any logic developed within the ACP model would actually work.

The extension of the VISSIM model for component selection was considered to be a natural extension of the use of the model, and, once the general strategy was developed it proved possible to implement. The results from the VISSIM model for component selection were identical to the results obtained by hand when the original design calculations for the Gamma radiography robot were completed. However, the VISSIM analysis, including the building the of model, was considerably faster than when the calculations were completed by hand. This was due to automated search of the solution space by the VISSIM model compared to the time taken carrying out several sets of calculations by hand, with different component combinations.

It would have been possible to carry out a similar analysis using an Excel spreadsheet or C program, however three problems exist. Firstly, the time domain solution to the Laplace transform which represents the motor, changes depending upon the type of input. Although a solution could be found to these equations they would have had to have been verified by the author using a package such as VISSIM, thus negating the need for the Excel model. Secondly, if an Excel spreadsheet were created, which, just used torque/acceleration/inertia relationships, it would not have allowed the program to identify combinations in which the load did not reach or overshot its target position. Finally the generation of an Excel model

and supporting macro, is likely to have been as time consuming as the generation of the VISSIM program used for the analysis, presented in this chapter.

Overall the use of the correlation chain, ACP diagrams and VISSIM for automated component selection was very successful. It has shown a detailed example of how computer tools can be successfully integrated with a function based design methodology and many of the benefits realised by the designer, such as: -

- Increased understanding of the system.
- Time saving efficiencies due to automation and staged selection criteria,
- Thorough search of the solution space.
- Use of the FAIL and STAGE files leads to an automatically generated record of the decisions made during the component selection process.

The only disadvantage of using VISSIM modelling for component selection is that it is possible for the designer to get too involved in the modelling task and generate very detailed system models when they are not necessary.

In conclusion, the above disadvantage is far outweighed by the benefits and this chapter proves the first part of hypothesis two which states that:-

- The integration of computer based models for component selection is feasible and beneficial to the design process.

The next stage of the design process is the expansion of the X axis model to include all the system components. Once this has been created it is possible to build a VISSIM program, using many of the techniques developed here, which is capable of carrying out automated fault analysis. This process is described in the next chapter.

9 Software models for fault analysis

9.1 Introduction

The objective of this chapter is to show, by example, the benefits of combining computerised failure analysis with EQFD. This objective will be achieved by developing the VISSIM system model, introduced in chapter 8, so that it encapsulates the complete system behaviour (including failure behaviour) of all components. Systematically selecting failure combinations and comparing the system output to a failure specification, which are logged if not met, produce reduced fault trees. These are used to provide the drivers for a 2nd stage EQFD chart, from which solutions to the failures identified in the fault tree are produced. The VISSIM model is then updated and the computerised failure analysis repeated. This process is repeated until an acceptable level of safety has been achieved, this is generally defined as the removal of all single point failures which can cause system failure (HSE[93]).

To show the benefits of applying structured design methods and tools to the process, four models were developed, all based upon the X axis of the gamma radiography robot.

- Model 1:- First generation model with few safety features
- Model 2:- Model incorporating safety features identified by CLASH/EQFD analysis, detailed in chapter 7.
- Model 3:- Second generation model of model 2, with safety features derived using an unstructured intuitive approach to design.
- Model 4:- Second generation model of model 2, with safety features derived using a structured approach to design.

The effectiveness of the methods will be measured by the number of failures in each model and the speed of analysis. Examples of the VISSIM models created for this analysis can be seen on supplementary disks A and B, their location being described in Appendix C.

This chapter details the research illustrating how computer tools can be integrated with EQFD to carry out automated failure analysis. It also provides a insight into the benefits of combining structured design tools with failure analysis tools.

9.1 VISSIM models for failure analysis

The VISSIM model for failure analysis is based upon upgrading the VISSIM model developed in chapter 8 to include all the elements shown in the ACP diagram of figure 8-2. In a similar manner to the VISSIM model for component selection several extra files are required. A schematic of the major components used in the analysis is shown in figure 9-1.

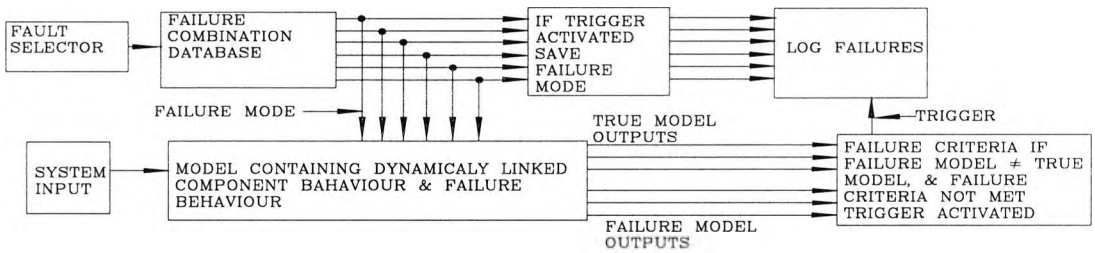


Figure 9-1 Schematic of VISSIM model for failure anlayis

The Fault Selector is a ramp input which accesses a .MAP file that contains all the failure combinations for a particular analysis stage. This operates in exactly the same way that the component combination selector did, except the information contained in the file relates to behavioural states for the various components. The component state information derived from this file is input into the various components and an operating sequence which tests the entire system functionality, is applied to the model. The output from the system is compared to the expected output of a perfect model contained within the VISSIM program. If there is a deviation between the behaviour of the perfect model and the model containing

failure behaviour, which is not detected and acted upon by leaving the system in a safe state, the failure is logged.

9.1.1 Upgrading the VISSIM model

The first stage in the development of the VISSIM failure analysis model, is to upgrade the model to include all the elements indicated in the ACP system diagram, (figure 8-2) plus additional components, that reflect the complete physical embodiment of the system. Because of the increase in the number of components, this model is much more complex than previous models as can be seen in figure 9-2.

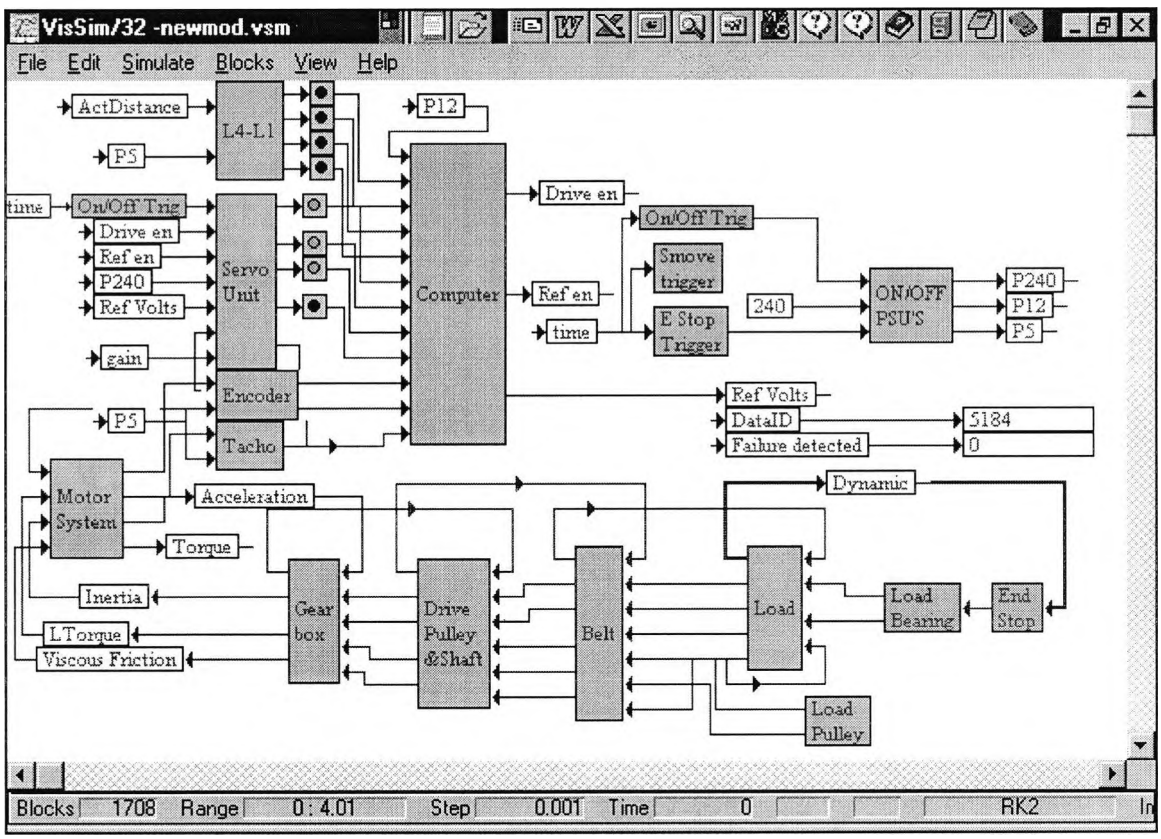


Figure 9-2 VISSIM model X axis drive system model

Formal ACP diagrams were not created for these models as they were found unnecessary. This is because the VISSIM environment provided a flexible and effective drawing tool which can capture all the relevant ACP diagram functionality. The model in figure 9-2 contains the basic elements of motor, gearbox, pulley, load and load bearing, plus additional elements, which give the model its full functionality. Each element and its basic functionality will now be described.

The COMPUTER block is the main controller of the system, it takes in a number of inputs such as the limit switches, tachometer and encoder signals, and outputs a control voltage plus other signals which control the servo system. The computer contains, a COUNTER, analogue to digital card, digital to analogue card and Input Output ports. They are located at the first level of system hierarchy (see figure 9-3) in the computer, because this is where these components would physically reside. This is in contradiction to the system ACP diagram of figure 8-2, but gives a better representation of the system.

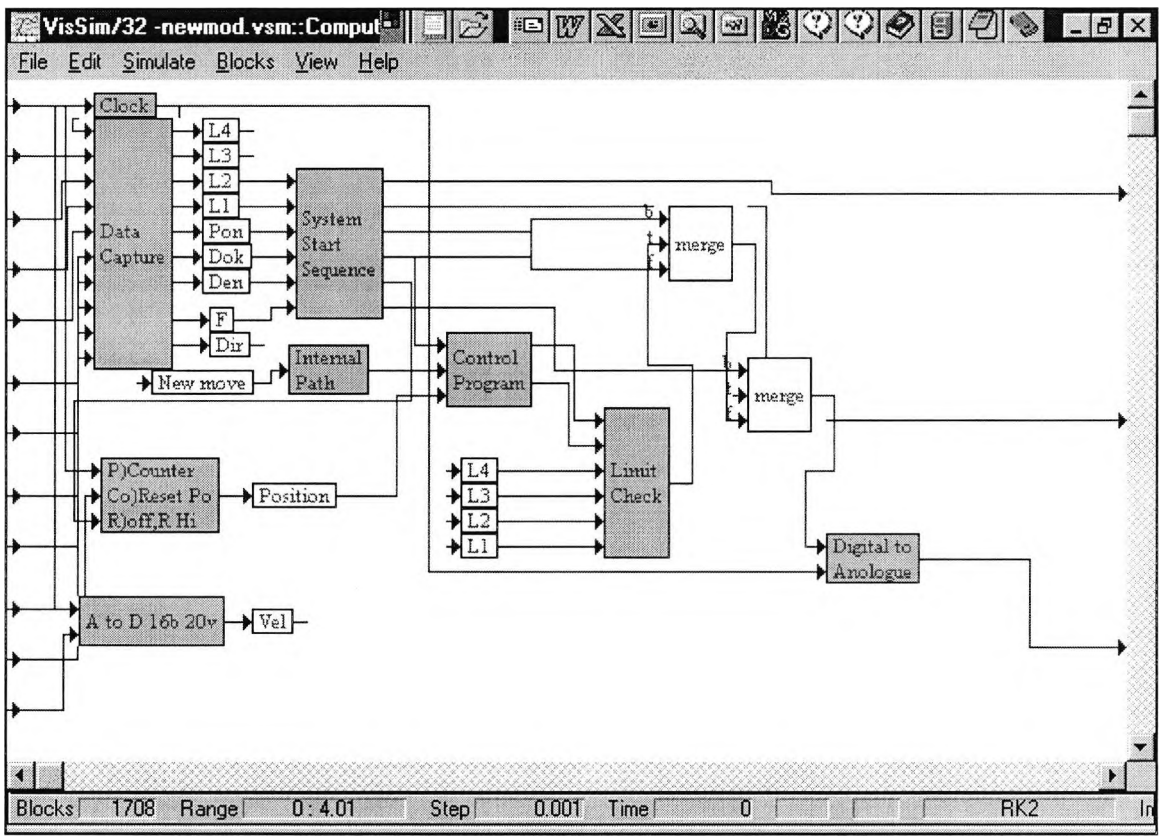


Figure 9-3 Functional decomposition of the COMPUTER block

The system control is encapsulated in a number blocks at this level. The SYSTEM START SEQUENCE block turns the servo motor on according to a specific sequence of events and initialises the axis. The INTERNAL PATH block contains an axis displacement profile. This is fed to the CONTROL PROGRAM block which generates the correct voltage to achieve the displacement. The LIMIT CHECK block contains the checks that control the velocity depending upon which limit switches are set. For example, when limit switch two (L2) is on, (the load is within 5 mm of the axis end) the velocity command

signal is ignored and overridden by a 1 volt velocity signal. This slows the load down until the correct position, or end stop is reached.

The block labelled LI-L4 represents the behaviour of the limit switches. These send out a positive signal whenever the load is within a given position of the end stops. The END STOP block prevents movement of the load by simulating a large decrease in efficiency and an increase in viscous friction. The ENCODER and TACHOMETER are used to provide control information relating to position and velocity of the motor. The ENCODER is a quadrate encoder which sends out two square wave signals out of phase by 90 degrees, this enables changes in motor direction to be detected as well as displacement. The TACHOMETER produces an output voltage proportional to the angular velocity of the motor, this signal is fed to the SERVO UNIT and the COMPUTER. The LOAD PULLEY block is included in the diagram because a failure is associated with its behaviour, as discussed in section 9.2.2.

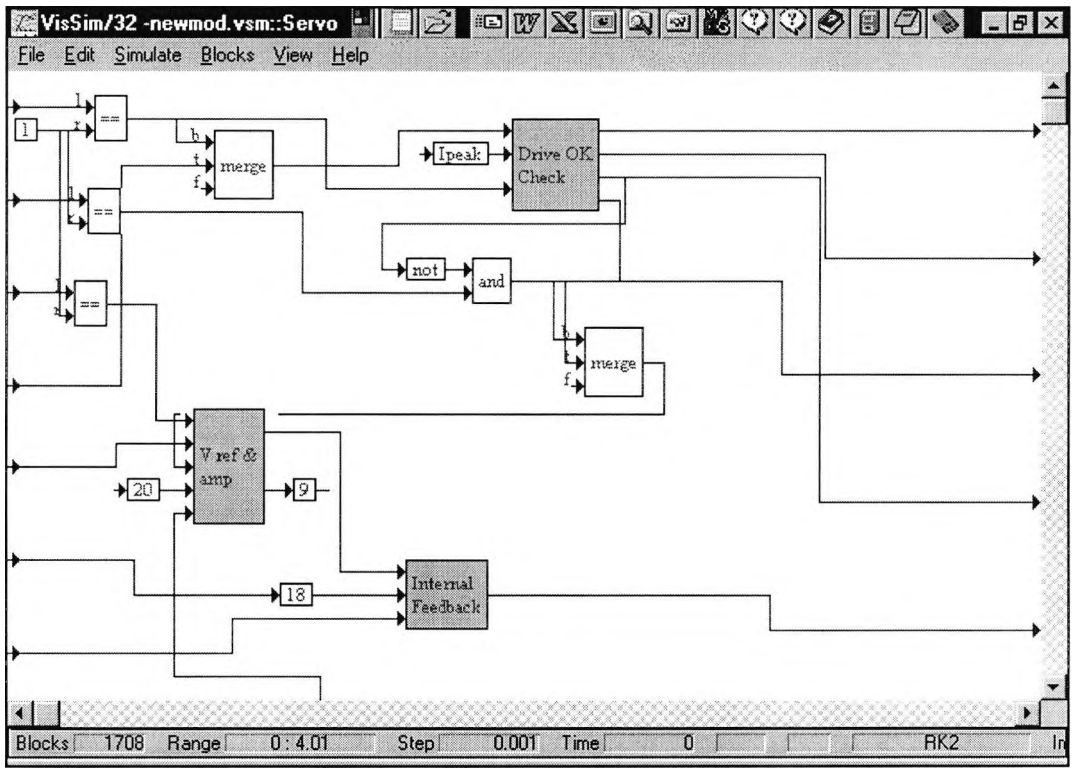


Figure 9-4 SERVO UNIT block

The SERVO UNIT (see figure 9-4) contains the behaviour described in its operation manual, which allows the motor to be checked and started up. It also contains the servo amplifier and control summing junctions, for the internal velocity control of the motor.

Finally, the ON/OFF PSU block, contains a hardwired relay circuit for the start, end and emergency stop functionality. It also contains the 3 power supply units required by the system , a 5 VOLT PSU for the limit switches, a 12 VOLT PSU for the computer and 240 VOLT PSU to drive the servo unit and motor combination.

The various lights at the output side of the SERVO UNIT and the L1-L4 blocks are used to indicate the state of the device for diagnostic purposes.

9.1.2 Failure modelling within the VISSIM model

The next stage in the analysis is the completion of a behavioural analysis for each component in the system. The object of this exercise is to identify the perfect and failed behaviour of each component, so that it can be recreated within the VISSIM model. It also provides information, to the designer, as to the number of potential failure modes present in the system.

9.1.2.1 Failure Mode Analysis

The initial failure analysis of the first generation model is shown in table 9-1. The blocks that contains the components, are enclosed in brackets.

Component	**Function mode**	**Function behaviour**
Switch	OK	
(ON/OFF PSU)	O/C (open circuit)	No signal
3 modes	S/C (short circuit)	Continuous signal
latched Emergency	OK	
Stop , E/Stop	O/C	No signal
(ON/OFF PSU)	S/C	Continuous latched signal
4 modes	Latch Fail	Momentary signal when E/stop pressed
Power Supply Unit	5 V PSU (ok)	Output = 5 V
PSU,	5 V PSU (nok)	Output = 0 V

Table 9-1 failure analysis for 1st generation model (continued overleaf)

(ON/OFF PSU)	12 V PSU (ok)	Output = 12 V
	12 V PSU (nok)	Output = 0 V
	240 V PSU (ok)	Output =240 V
2^3 modes	240 V PSU (nok)	Output = 0 V
Limit switches	L1 (ok)	5 V out when limit reached
(L1-L4)	L2 (ok)	5 V out when limit reached
	L3 (ok)	5 V out when limit reached
	L4 (ok)	5 V out when limit reached
	L1 (Non Contact)	No signal when limit reached
	L2 (Non Contact)	No signal when limit reached
	L3 (Non Contact)	No signal when limit reached
2^4 modes	L4 (Non Contact)	No signal when limit reached
Vref & Amp	Signal OK	Correct gain
(SERVO UNIT)	Noisy signal	Noise on output
2 modes		
Gain	Signal OK	Correct gain
(SERVO UNIT)	S/C	10 V out
3 modes	O/C	0 V out
Motor	OK	Correct operation
Encoder	OK	Correct operation
	No signal	No output (counter stationary)
3 modes	Bad connection	Intermittent Output
Tachometer	OK	Correct operation
	No signal	No output
3 modes	Bad connection	Intermittent Output
Gearbox	OK	Output torque OK
2 modes	Jams	Torque out is very high & increase in viscous friction, efficiency decreases
Belt	OK	normal load carry functionality
	1 Belt breaks	decrease tensile strength of belt by 2
3 modes	2 Belts break	No load transmission

Table 9-1 Failure analysis for 1st generation model (continued)

Bearing 2 modes	OK	Output torque OK
	Jams	Torque out is very high & increase in viscous friction, efficiency decreases
Drive pulley 2 modes	OK	Output torque OK
	Jams	Torque out is very high & increase in viscous friction, efficiency decreases
Load Pulley 2 modes	OK	Output torque OK
	Jams	Sawtooth efficiency profile

Table 9-1 Failure analysis for 1st generation model (continued)

The analysis for the limit switches was simplified, this is due to the large number of possible failure combinations which can be derived if all the limit switch failure modes were taken into account. The modes ignored were Short Circuit (S/C) and Open Circuit (O/C) failures. If these were included it would have increased the total number failure combinations by a factor of 16. Failure of the Motor was ignored, because any motor failure mode would be a right side failure. For instance, decreases in efficiency would lead to the motor stopping faster or not moving at all. In addition, efficiency losses would also be seen by the servo unit as high current, which is limited by the current controller. When the limit is reached, the servo unit shuts the motor down. Failure within the computer was also ignored as the effect of, for instance, a line high failure in the data capture board is already represented within the model by a short circuit in a component.

The total number of failure combinations, which need to be produced in the failure selection file, is calculated by multiplying the sum of all the possible behavioural modes for all components together. The number of behaviour modes is indicated for each component, in table 9-1. By applying the calculation mentioned above, the total number of failure combinations for this system was found to be 3, 981,312. Initial testing indicated that if all the failure modes were applied to the model it would take approximately 1. 5 years on a 40 Mhz 386 PC or 6-7 months on a Pentium160 MHz .PC. This does not take into account the time necessary for analysis of the output data.. A timescale of this magnitude is unreasonably long and it was decided to rationalise the behavioural analysis by identifying failure modes which:-

- Produced similar behaviour effects

- Were unnecessary due to duplication of component functionality
- Would obviously produce a right side failure

By applying the above criteria to the components, the following simplifications were made:-

- The failure modes for the 240 and 12 V PSU were removed as they would produce a right side failure of the system, as no power would be passed to the computer, or the motor and servo unit.
- The failures associated with the Limit switches can be reduced by removing Limit 3 and limit 4. This is because their functionality is repeated in Limits 1 and 2. Because of this reduction, the simulation time can also be reduced as the load will not have to move through Limits 3 and 4 to test them.
- The effects of the drive pulley jam, gearbox jam, load pulley jam and load bearing jam are identical, so to reduce the number of failure combinations only the load bearing jam will be modelled
- The breaking of 1 belt has an insignificant effect on the load, unless the load on the remaining belt is greater than its load bearing strength. It also complicates the model unnecessarily.
- The On/Off Open Circuit failure was removed because it would not allow any power to be passed to the system, thus leading to a right side failure. In a similar manner the Emergency stop failure S/C is effectively a Normally Closed (NC) switch, that, prevents any power from leaving the PSU block.
- The error in the gain amplifier was changed from being a + 10 V or 0 V output to +/- 10% drift, as it was felt that the gain amplifier was more likely to exhibit this failure mode.
- Finally, it was decided that mechanical failures were unlikely to occur simultaneously, although failure in one component could cause a failure in another. For this reason it was decided to model the load pulley jam and the belt breakage as single state mechanical failures.

These alterations give the failure mode behaviour chart indicated in table 9.2

Component	comp ID	Function mode	F/mode ID	Function behaviour
Switch (2 modes) (ON/OFF PSU)	1	OK	0	Behaviour OK
		S/C (short circuit)	1	Continuos signal
latched Emergency Stop , E/Stop 3 modes	2	OK	0	Behaviour OK
		O/C	1	No signal
		Latch Fail	2	Momentary signal when E/stop pressed
Power Supply Unit PSU, 2 modes	3	5 V PSU (ok)	0	Output = 5 V
		5 V PSU (nok)	1	Output = 0 V
Limit switches (L1-L4) 2^2 modes	4	L1 (ok)	0	5 V out when limit reached
		L2 (ok)	1	5 V out when limit reached
		L1 (Non Contact)	2	No signal when limit reached
		L2 (Non Contact)	3	No signal when limit reached
Vref & Amp (SERVO UNIT) 2 modes	5	Signal OK	0	Correct gain
		Noisy signal	1	Noise on output
Gain 2 modes (SERVO UNIT)	6	Signal OK	0	Correct gain
		Drift	1	varies randomly between +/-2.5 %
Encoder 3 modes	7	OK	0	Correct operation
		No signal	1	No output (counter stationary)
		Bad connection	2	Intermittent Output
Tachometer 3 modes	8	OK	0	Correct operation
		No signal	1	No output
		Bad connection	2	Intermittent Output
mechanical failures 3 modes	9	OK	0	Correct operation
		Belt	1	No load transmission
		Load bearing	2	Increase viscous friction and decreased efficiency

Table 9-2 Reduced failure mode behaviour for 1st generation model

The number of combinations produced by these behaviour variations is 5184, a notable reduction from the original 4 million, but still not an insignificant number. Even though the complexity of the failure behaviour has been reduced, the integrity of the model has been kept by applying logical reduction criteria. Testing of the model showed that the analysis would take approximately 17 hours on an 386 40 MHz PC, originally thought to be a reasonable time scale for the analysis.

9.1.2.2 Modelling perfect and failure behaviour

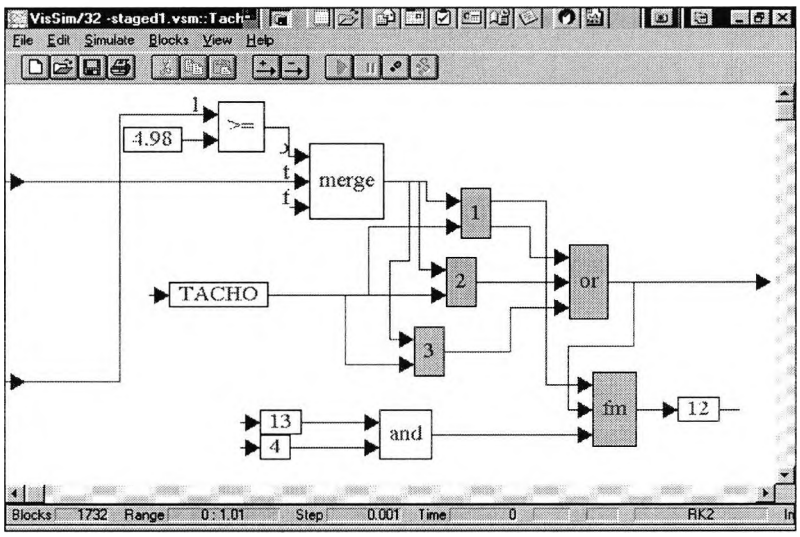


Figure 9-5 TACHOMETER failure model

This section deals with an example for the development of a multiple behavioural model for a particular component, the tachometer. The complete failure model of the tachometer can be seen in figure 9-5. The inputs to the tachometer are a voltage of 5 volts, required to provide power to the system, and the input velocity. The system output is a voltage proportional to the input velocity, this ideal behaviour is encapsulated in the compound block labelled 1 (figure 9-6). Compound block 2 contains no connections between the input and output, whilst compound block 3 (figure 9-7) contains the intermittent behaviour. This is recreated using a RANDOM block, which generates a signal every sample period. When the signal is greater than 0.5, (indicates 50% intermittency) a 1 is output, which allows the generated voltage to pass to the output.

The blocks labelled SS in figures 9-6 and 9-7 are State Selectors. When the ID number relating to a particular behavioural model is input into the state selector it allows the behavioural output relating to the selected state to be output from the model.

This output is passed to an OR block and becomes the tachometer output. The tachometer output signal is compared to the output of the perfect behavioural model of the tachometer in the FM block. The variables input into the FM block allow failure propagation to be passed through the system. If a failure occurs, this line is set low (0 output) and passed onto the next component in the system.

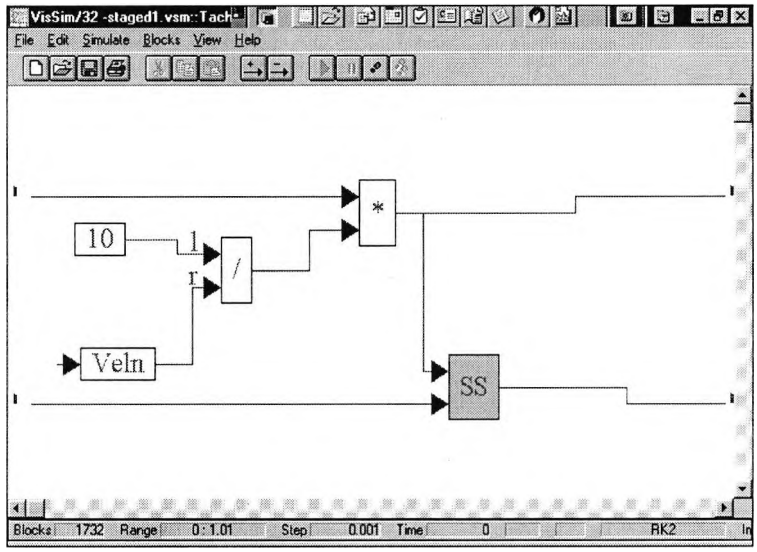


Figure 9-6 TACHOMETER ideal behaviour model

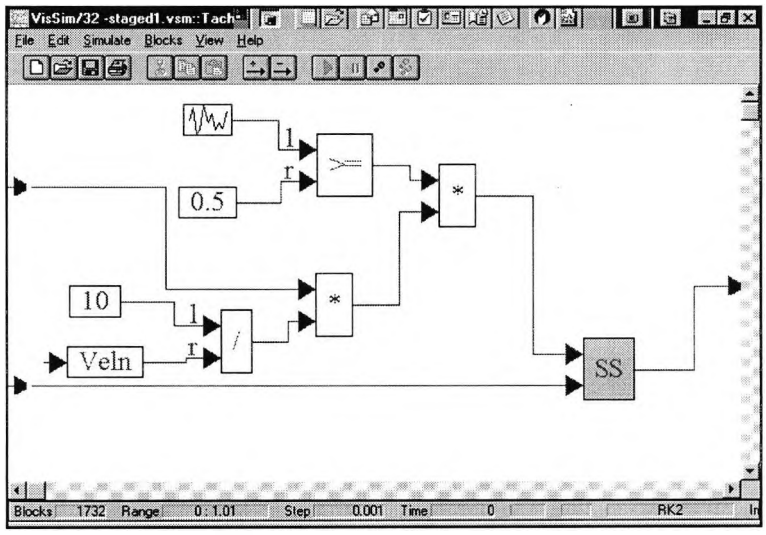


Figure 9-7TACHOMETER intermittent failure model

9.1.2.3 Generating the failure combination file

The failure combination file is produced in a similar manner to the component selection file, as explained in chapter 8. In this case, however, the file contains 9 input lines, each relating to a particular component, with differing amplitudes and rates being applied to the ramp inputs. These rates depend upon the number of variations in component behaviour.

A section from a behaviour or failure combination file can be seen in table 9-3.

#I=0,5184,1; FMEASW4.VSM @ Mon May 26 12:15:17 1997										
0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	1
2	0	1	0	0	0	0	0	0	0	2
3	1	1	0	0	0	0	0	0	0	3
4	0	2	0	0	0	0	0	0	0	4
5	1	2	0	0	0	0	0	0	0	5
6	0	0	1	0	0	0	0	0	0	6
7	1	0	1	0	0	0	0	0	0	7
8	0	1	1	0	0	0	0	0	0	8
9	1	1	1	0	0	0	0	0	0	9
10	0	2	1	0	0	0	0	0	0	10

Table 9-3 section of failure combination file

Each row relates to a failure combination , and each column relates to a component. Values represent a behaviour mode for a particular component. For example, combination 10 relates to failure mode 2 (Latch not OK) in component 2 (E/stop switch) and failure 1 (PSU not OK) in component 3 (5 Volt power supply unit).

9.1.3 Operating sequence

An operating sequences was developed to test as much of the system functionality as possible. This operating sequence differs between the first and second models, because the second and subsequent generation models were more sophisticated and, therefore, required a different operating sequence to test system functionality.

The operating sequence associated with the first model is started by the operator pressing the start button. The operator is modelled as a ramp signal which acts as a time base, when

a particular time has been reached events, such as, pressing the start button, are triggered. The load is positioned 6 mm from one end of the axis and is then moved backwards, allowing the computer to check the limit switches and datum the axis. When the end of axis has been reached (indicated by L1 going high) a trapezoidal velocity move is initiated, at 0.18 seconds the emergency stop button is pressed. This sequence of events tests all of the possible failures in the system as shown in the results of the first model.

The sequence of events for the second, and subsequent generation models slightly differ. The start position of the load is set to the beginning of the axis, the load is moved forward 10 mm and then back 10 mm, returning to the start position, allowing datuming to take place and a check of the limit switches. The velocity move is then initiated, during which the emergency stop is activated The operating sequence was changed because the second and subsequent generation models incorporated a sophisticated, time dependant start sequence that required the position of the load to be at the end of the axis.

9.1.4 Modelling of Failure criteria and data logging

The safety criteria for the system is stated in section 6.3, but will be repeated here. Two wrong side failure criteria (F_{C1} and F_{C2}) were selected. Thus, the two most important actions, on detection of failure, are that the Gamma source (or other NDT payload load) should:

- Stop moving as soon as possible (i.e load velocity = 0 or as near as possible to 0).
- Be detected to be within 10% of its expected position.

Reflecting the impossibility for a system to react immediately to a failure condition, the actual response time was set as 0.06 seconds. Furthermore, for the purposes of simulation, the wrong side failure was defined as:

- F_{C1}:-The velocity of the NDT load is greater than +/- 0.0001 M/s after 0.06 seconds from the time that the difference between the perfect model and the actual model (with failure modes) is detected.

- F_{C2}:- The position of the NDT load, under failure conditions, differs by more than ± 10 % of that expected after 0.06 seconds from the time when that difference is first apparent.

These two criteria are captured in the block, FAILURE CRITERIA FILTER which contains two further compound blocks the FAILURE CRITERIA block (see fiure 9-8) and the FAILURE LOGGING blocks (see figure 9-9)

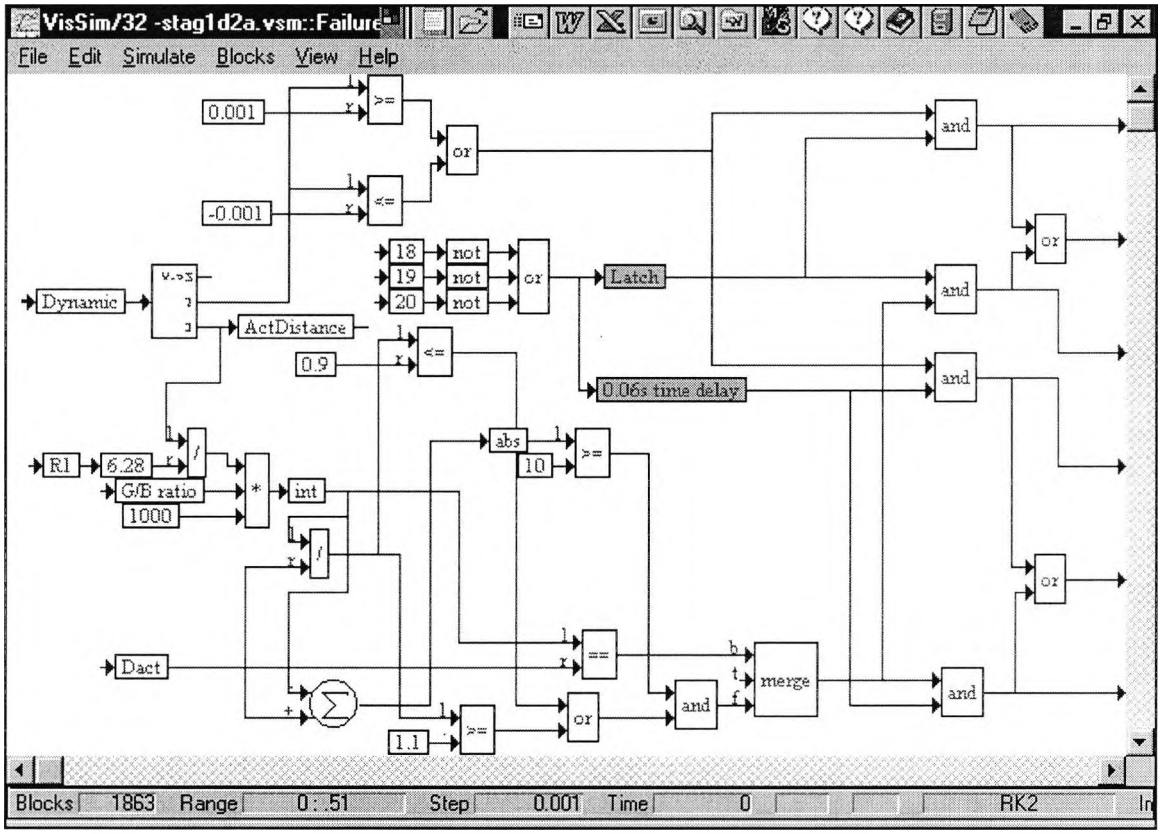


Figure 9-8 FAILURE CRITERIA block

The FAILURE CRITERIA block has four inputs and six outputs. The most important input is the DYNAMIC vector input, which contains displacement, velocity and acceleration information, these are separated using the VECTOR TO SCALAR block. The other inputs are the variables associated with the end of the failure propagation chain (indicated by the numbers 18.19, and 20). If any of these lines are low (0) they are inverted and used to trigger the 0.06 second latch. If the right side failure criteria are not met in this time period, then a signal is output to log the failure.

The top part of Figure 9-8 is used to monitor the velocity, ensuring that it is less than 0.001 mm/s when a failure occurs. This is as close to zero as it possible to measure. The lower right hand corner contains the section of code that deals with monitoring the displacement and ensuring that it is within 10 % of the expected displacement.

The six outputs from the failure criteria block relate to the six files which log the data, the 3 upper outputs allowing the logging of failures as they occur, whilst the lower three deal with failures which occur 0.06 seconds after the failure has been detected. Information logged by the triggering of the top 3 failure criteria is generally ignored, but gives an indication to the designer of failure modes which are detected and leave the system in a safe state.

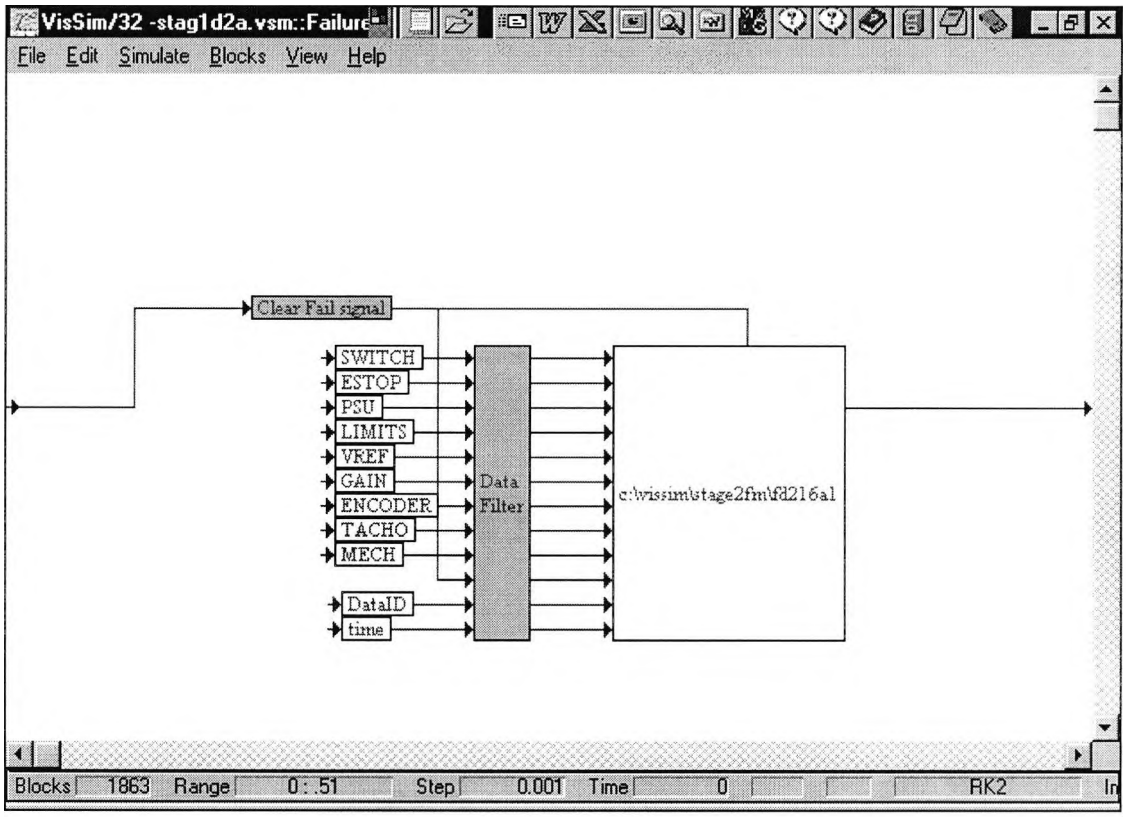


Figure 9-9 FAILURE LOGGING block

Three files are used for each type of failure, files 1 and 4 contain combinations which failed under Fc1, files 3 and 6 contain combinations which failed under Fc2 and files 2 and 5 contain combinations which fail under both criteria. The output signal from the failure criteria block is used to trigger the logging of the failure combination.

9.2 Failure models for fault analysis

The models created for fault analysis were quite large and can be downloaded from the supplementary disks A and B, the location of the files and their function is described in Appendix C. A condensed version of the relevant files produced as a result of running these tests can been in Appendix B, again the full versions are contained on the supplementary disks.

9.2.1 1st Generation Model

In the first attempt at modelling the failure modes of one axis of the robot, 9 components were identified as having failure modes that could possibly lead to failure of the system. Initial, failure simulation runs showed that wrong side failures were correctly detected.

However, it was found that the analysis of the data was considerably more complicated than expected, the failure modes detected were all single point failures, but they were embedded within a very large fault tree of approximately 3000 failure mode combinations This was a result of the fact that if a single point failure is present in a multiple point failure combination, which may not produce a failure if the single point failure is removed, then failure will still occur. The level of failure (i.e. single point or double point) then needs to be filtered out by examining the failure logs.

On account of the processing time, coupled with the observation that output data analysis could not be carried out efficiently, it was decided to identify a different approach to failure reasoning, optimistically producing sensible amounts of data that can be readily analysed.

The approach taken was to start with single point failures. Where the failure mode of the component did not produce a single point failure, all possible two or double point failures (DPFs), excluding failures which produced a single point failure, were selected and the simulation repeated with the modified data set. This dramatically reduces the number of failure simulations that needed to be carried out, and produces an easily manageable data analysis task. This is a variation of a breadth first search, (Jackson[90]), or beam width search (Gen[97]), in which all the combinations at a particular level of a search tree are tested. next level of the search tree are examined. An example of the advantages of this process can be seen figures 9-10 .

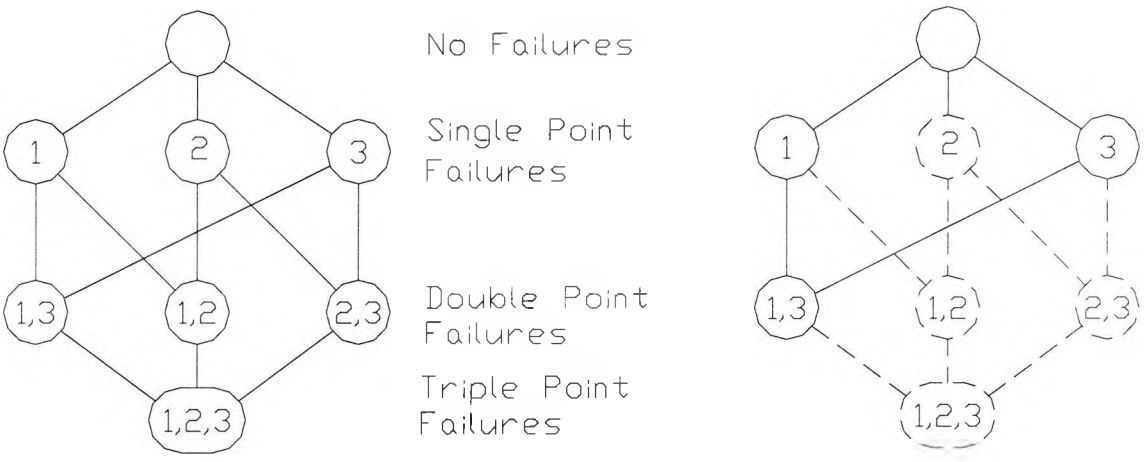


Figure 9-10 The effect of a breadth first search on the failure space

In figure 9-10, three components are shown at 3 levels of possible failure. The number of possible failure combinations for this system is 7. At the first level all single point failures are examined. If for example component 2 produces a system failure. Double point failure combinations 1,2 and 2,3 and the triple point failure 1,2,3 are removed from the analysis, (indicated by the dashed lines in the figure 9-10). This leaves 1 double point failure combination, to be examined (1 and 3). Even with this small example, a 43% reduction in analysis time is achieved, by having to test only 4 failure combinations instead of 7.

Because the number of failure combinations changes at each stage, the VISSIM program used to create the failure combination files was modified so that failure modes could be removed. This was achieved by running the complete failure combination file of 5184 combinations through a selector (set by the user), which filtered out combinations that contain selected failure modes. Figure 9-11 shows part of model SWFILT2D.VSM in which this process is carried out.

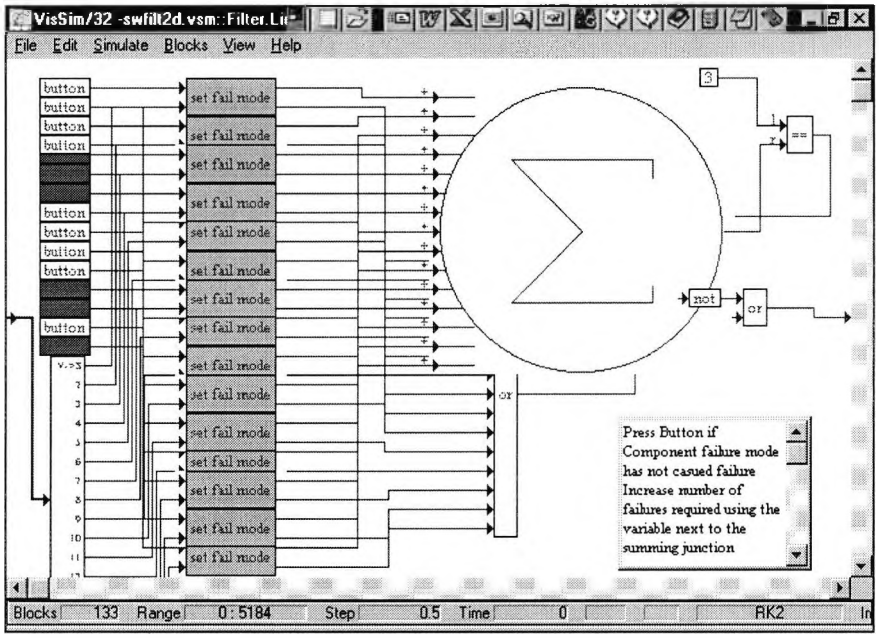


Figure 9-11 Section of SWFILT2D.VSM showing failure mode selection for triple point failures

For the first generation model, a single point, failure combination file was produced with 14 combinations (SPFAIL1.MAP). These combinations were tested using, VISSIM model STAGES1.VSM. This resulted in 14 single point failures for Fc1 (FS114.DAT) and 13 single point failures for Fc2 (FS116). The remaining failure combinations were filtered out using SWFILT2D.VSM. This was used to generate DPFAIL2.MAP, which contained 2 double point failure combinations, that both produced failure under Fc2 (FS126.DAT). The reduced fault trees for both failure criteria can be seen in figures 9-12 and 9-13, the key relating each component and failure mode to a specific number is given in table 9-2.

The reason for failure of each of the failure mode combinations was identified and found to be justified. This provided verification that the model was correct, and showed that some form of proactive failure analysis should have been carried out, as all the single failure modes produced a system failure.

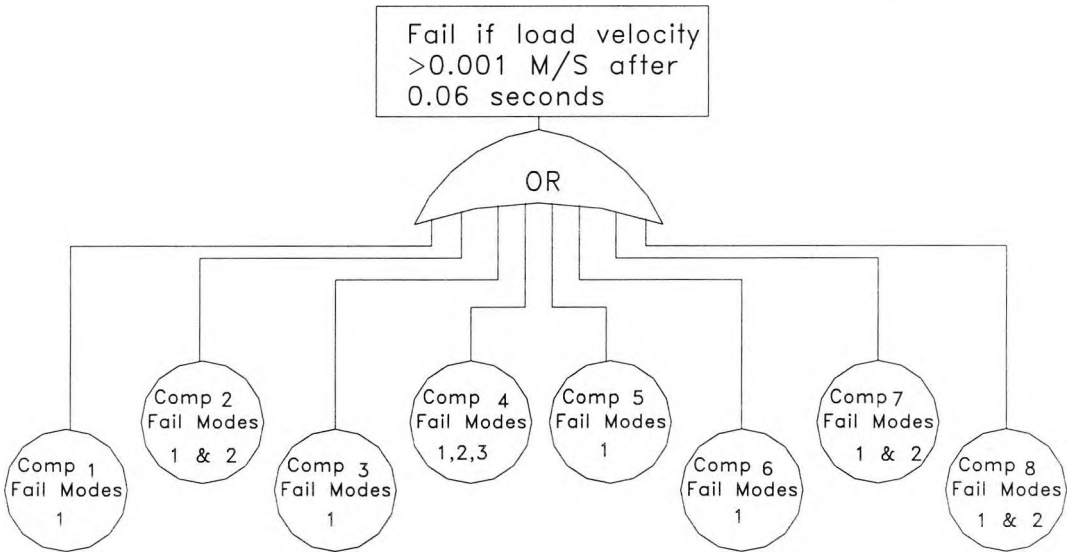


Figure 9-12 RFT for model 1, Fc1

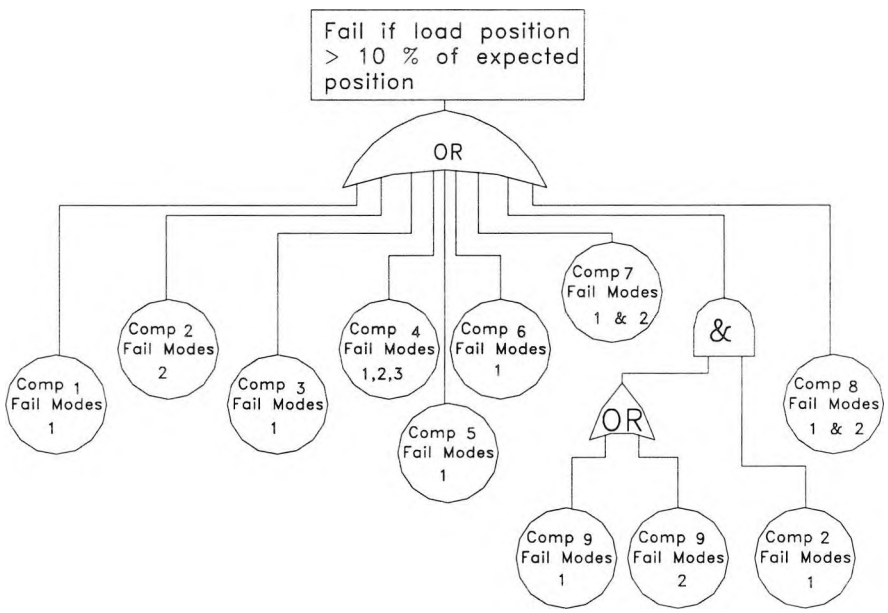


Figure 9-13 RFT for Model 1, Fc2

9.2.2 2nd Generation Model

The second generation model (STAGES2A.VSM) was based upon the inclusion of safety features indicated by the proactive CLASH/EQFD study carried out in chapter 7. The study highlighted the need, for the comparison of the displacement signal with the derived displacement from Velocity signal, used to check the encoder and tachometer. The other safety feature proposed, was the checking of sensors, such as, limit switches and the tachometer against each other, this being an implementation of redundant sensing.

The first requirement was achieved by integrating the velocity signal and comparing it to the load displacement. If the difference between the two signals differed by more than 0.1 radians (equivalent to linear displacement of 1.25 mm) , then a fail signal is pulled high, and the motor stopped. This functionality is contained within the INTERRUPT BASED CONTROLLER block and is shown in figure 9-14.

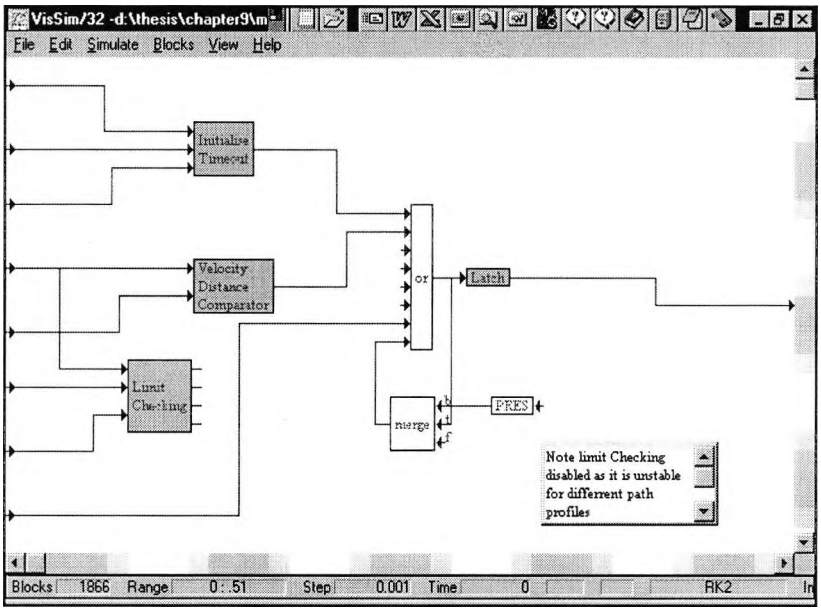


Figure 9-14 Interrupt based failure checking

The second requirement was achieved through the use of a staged start-up and load datuming sequence. This sequences the start-up of the motor controller and checks the output of the limit switches against the displacement sensor and the sensed displacement direction, whilst axis datuming is occurring. A time-out was also added to ensure that the start up sequence was completed in a set time. If any error occurs in the system which causes the motor to slow or fail to complete the sequence in this time, then the time out

triggers the fail signal. This is shown in figure 9-15, which is taken from the contents of the SYSTEM START SEQUENCE block.

Finally, a separate element of computer code was added to ensure that the limit switches were working correctly during the trapezoidal velocity move, by checking load position against limit switch activation (shown in figure 9-14). This final addition was unsuccessful and removed from the model, because the position of the load around the limit switches is dependant upon the velocity of the load and the sampling time of the control system., if the velocity of the load is too fast then the position given by the encoder is greater than the limits set in the computer.

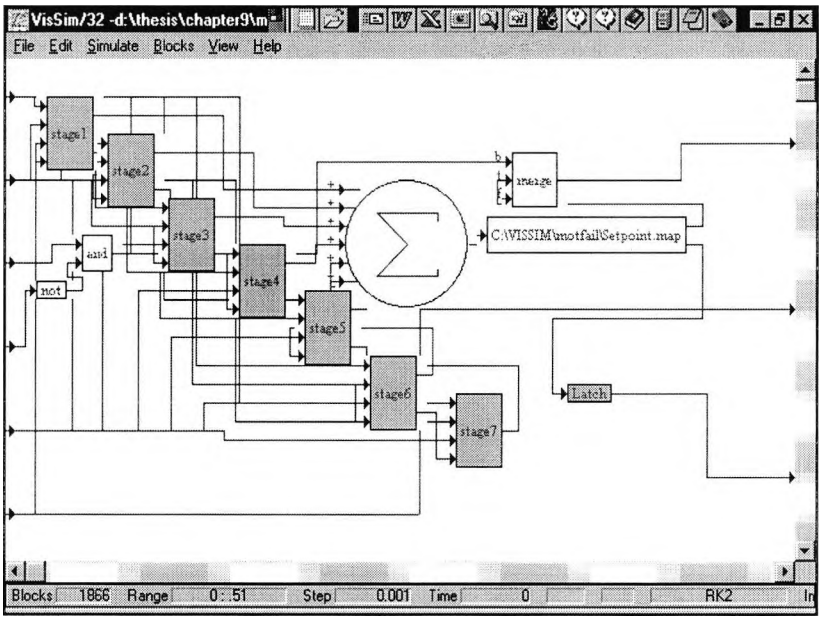
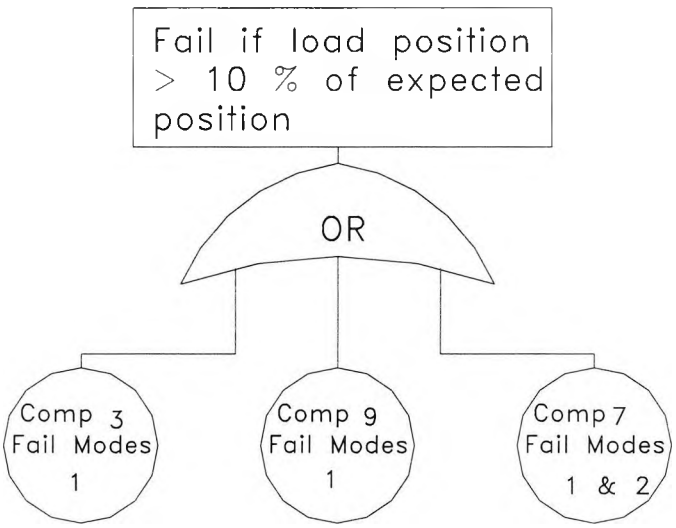


Figure 9-15 VISSIM models for staged startup

This model produced 6, single point failures under Fc1 (FS214.DAT) and 4, single point failures under Fc2 (FS216.DAT). Because the failure modes differed for both failure criteria, a separate series of tests were carried out for each failure criteria. This resulted in 1 double point failure being discovered under Fc1(FS2244.DAT), (checked up to quadruple point failures) and no further failures under Fc2 (Checked up to 6 six point failures). These results, produced the Reduced Fault Tree shown in figures 9-16 and 9-17.



igure 9-16 RFT for model 2, Fc1

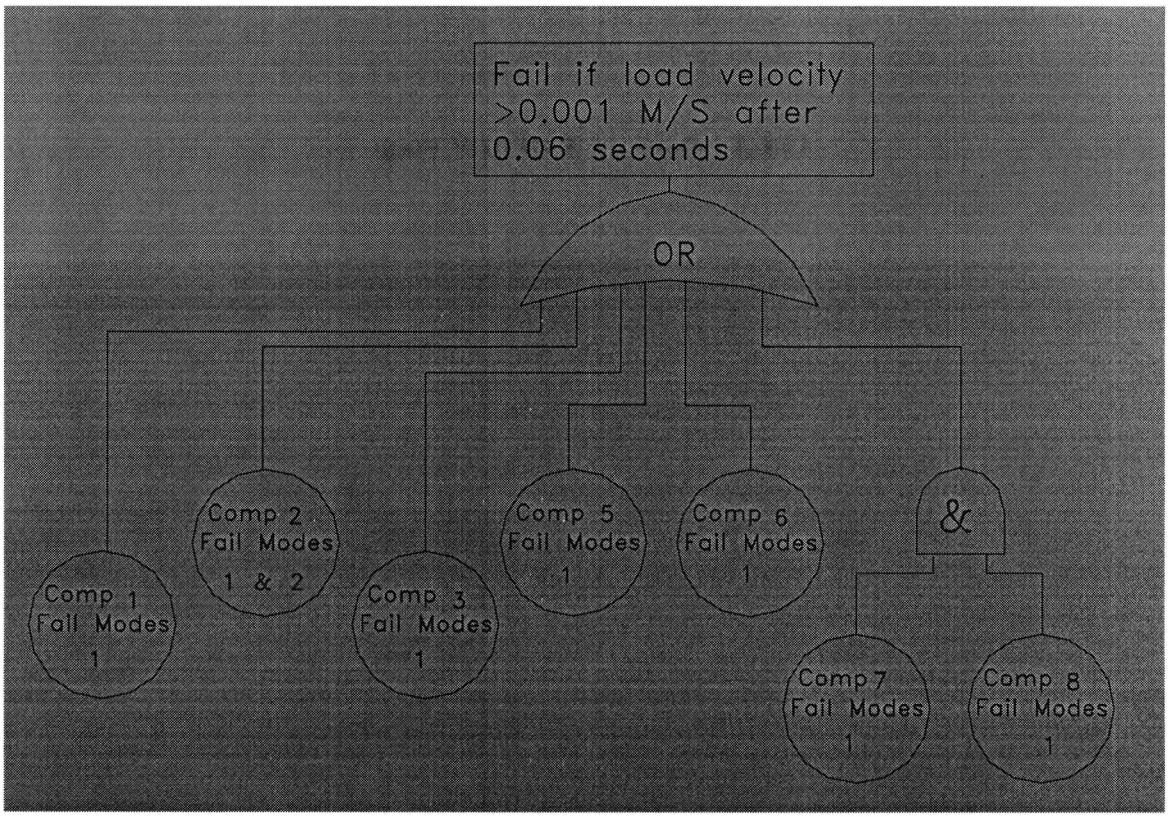


Figure 9-17 RFT for model 2, Fc2

The modifications indicated by the Proactive use of the CLASH/EQFD approach under Fc1, resulted in detection of faults in the limit switches (component 4) and the faults due to

mechanical failure (Belt break and pulley jam). The double point failure of the encoder and tachometer is due to the failure in each to produce an output voltage, therefore the derived displacement signal and the actual sensed displacement are equal and failure is undetected.

For Fc2, velocity less than 0.001 m/s after failure detection, the modifications in the system produced a more dramatic result, with only 4 single point failures. These were in the 5 V PSU, the encoder and the belt break.

Even though the number of failure combinations detected were reduced by approximately 75%, due to the use of a proactive approach to failure analysis, it was felt that this could be improved by the use of a further design stage. To show the effect of applying an structured approach to design for safety, two further models were created. Model 3 shows the effect of intuitively modifying the software, whilst Model 4, shows the effect of using FTA and FMEA within the structure of EQFD.

9.2.3 3rd Generation Model

The 3rd generation model (STAGES3.VSM) was developed using software to determine where the system had failed. The idea was to keep the number of components in the system as low as possible to reduce the number of failure modes, and therefore the number of possible failure combinations. By examination of the RFT of figure 9-17, it was decided to concentrate on removing failures relating to, the 5 V PSU, which interacts with the limit switches, encoder and tachometer, and the encoder failure caused by intermittent signals. The author felt that the other failures, indicated by the 2nd generation model, could not properly be tackled unless a major redesign took place. This will be detailed in section 9.3.2.

The 5-volt PSU was checked for failure by adding an input line to the I/O port interface, which was connected directly to the 5 V PSU output. If this line was pulled low for any reason, a fail signal was generated which would stop the motor.

The encoder was checked for intermittent pulses, by sampling the encoder counter and holding its value for one sample period, this is compared to the value of the encoder counter at the next sample period, if they are the same and the velocity of the load is greater than 0.5 rad/s, then the fail line is sent high and the motor will stop.

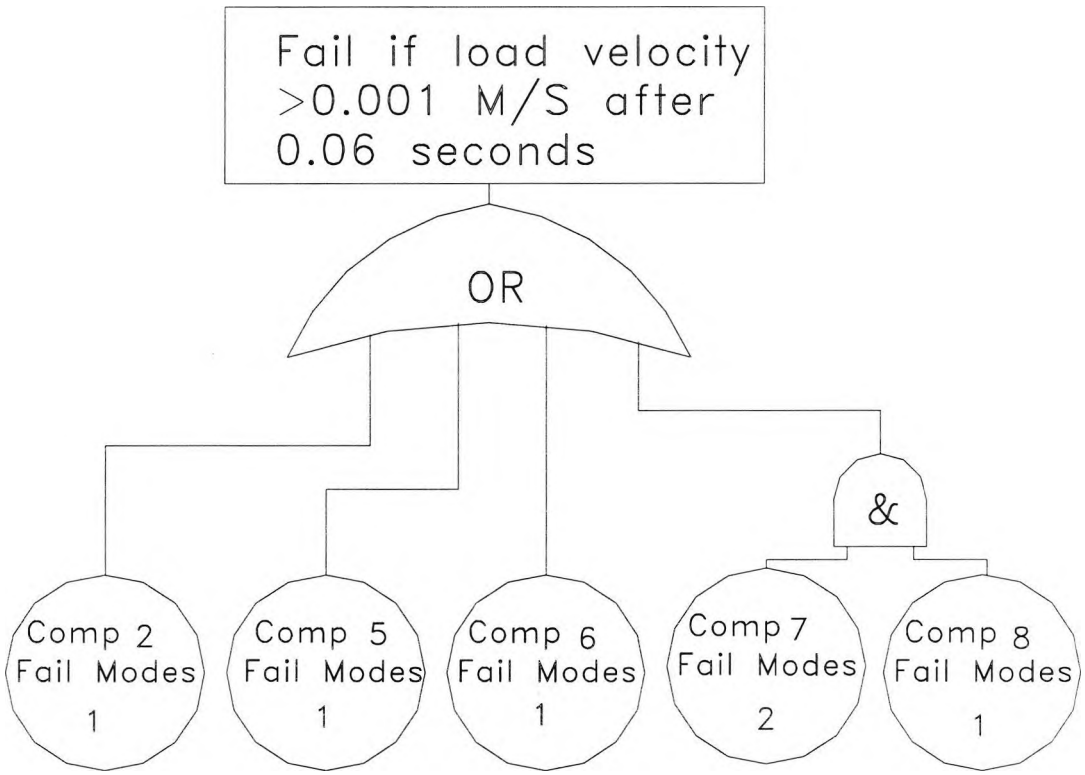


Figure 9-18 RFT for model 3, Fc1

Under Fc1, 3 single point (FS314.DAT) and 1 double point (FS3244.DAT) failures were found, the model checked up to 7 point failure modes. The implementation of the software based fault detection of the encoder and the 5 Volt PSU was successful, changing the emphasis from single point to double point failures. This is a positive improvement, as the probability of both events occurring is the product of the probabilities of the individual events (Klassen[89]).

This change in emphasis from single to double point failures is continued when the RFT for Fc2 is examined. Under this criteria, the system reliability deteriorated so that 5 single point (FS316.DAT) failures were detected and 7 double point (FS3266.DAT) failures. The most prevalent cause of failure was noise from the amplifier interacting with other electronic components to prevent the correct load position being reported to the computer.

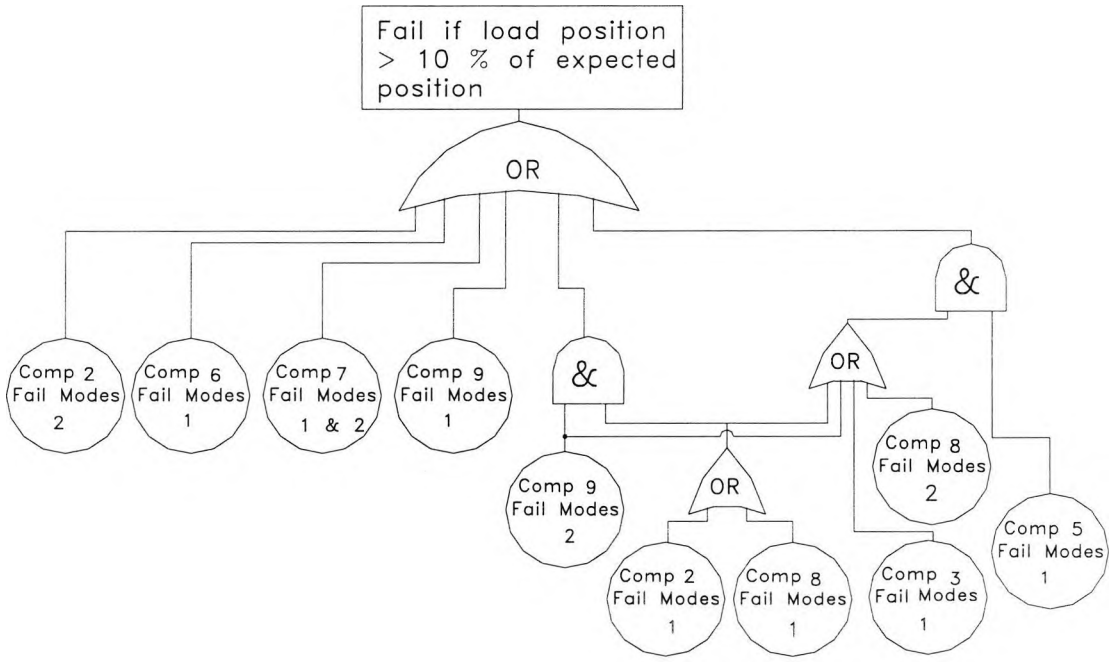


Figure 9-19 RFT for model 2, Fc2

9.2.4 4th Generation Model

On account of the mixed success of applying a software based solution to the problem of fault detection, the author felt it was necessary to develop the design in a more structured and controlled manner. To achieve the improvement in the system reliability, the author applied FTA, FMEA and a second stage EQFD analysis to the fault tree analysis of the 2nd generation model. From the outputs of this analysis, a fourth generation model was designed and generated.

The process starts by carrying out an FMEA of the system incorporating the failure criteria and component failure mode under which the component failed. This is used to define RPNs and possible solutions to solve each possible failure mode.

The next stage is a form of Morphological chart, which maps all the possible solutions to all the possible failure modes for each component. From this, possible solutions that achieve the negation of more than one of the possible failure modes can be found.

A combined, stage 2 EQFD/weighted objectives chart is then created with the negation of the failure effects as WHATS, and the possible solutions, derived from the earlier morphological analysis and FMEA as HOWS. The solutions are ranked against the

customer requirements and a score derived for each solution, the best option of which is picked.

The final stage is a further FMEA, which is carried out on the new components in the design to determine the failure modes and identify possible problems with implementing the failure mode.

Once this process has been completed, the VISSIM failure model can be updated with the new design and tested in the manner described previously to produce the RFT.

9.2.4.1 Generating the FMEA chart

Table 9-4 is a FMEA chart based upon the RFT of the 2 nd generation model. It is used to carry out a detailed analysis of the system, forcing the designer to examine the system and examine the interaction between system components.

The first column in table 9-4, indicates the failure mode identification number, whilst the second shows the component identification number. The failure mode apertaining to the component and failure mode number is shown in the third column. The next three columns indicate which failure criteria the component failed. In this instance :-

- FC1:- Velocity = 0 after 0.06 seconds,
- FC2:- Position of source known to within 10%, after 0.06 seconds
- FC3:- when either FC1 or FC2 occurs

The seriousness for each mode is then ranked , 1= low, 5 = high. The column, headed by a P, is used as a measure of the probability of failure occurrence, 1= low, 5= high. The degree of detectability of failure is shown in the column headed by a D, 1= high, 5= low. Possible causes of failure are described in the 10^{th} column, and the Risk Priority Number is given in the 11^{th}. The final column indicates possible solutions to the failure, this is either achieved by detection or preventation.

No.	Comp/ No	Failure mode	FC 1	FC 3	FC 2	Seriousness 1-5	P	D	Possible causes	RPN	Possible solutions
0	0	Everything ok									
1	Switch (1)	Short Circuit	X	X		5 causes un-controlled start conditions	2	5	Switch contacts welded together Need to increase detectability of system to prevent uncontrolled starts	50	Employ key switch in series, with failure indicator lights and Operator feedback or Microprocessor unit which detects zero time between activation of sw1 and sw2
2	E stop (2)	Open Circuit	X	X		5 Unable to stop robot	2	5	Switch failure	50	2, Robust NC emergency stop switches in parallel, activated by single point load.
							2	5	Broken contacts	50	Use indicator light for detectability
3		Latch not operating	X	X		3 Movement is not prevented	2	5	Failure in relay Also system at present is configured such that power to motor controller is not cut off when E-stop is activated	30	As above or Change wiring configuration so power to motor is cut off and Brake is motor brake deployed
4	5V PSU(3)	Not working		X	X	5 Unknown state of limits, position and velocity source	2	2	Broken wiring	20	Failure, although not detected immediately would be found during initialisation
							1	2	General PSU failure	10	Feedback 5 volt signal to microprocessor or Use signal lights on components to indicate wire failure

Table 9-4 FMEA chart for 2nd generation model failures (continued on the next 2 pages)

No.	Comp/ No	Failure mode	FC 1	FC 3	FC 2	Seriousness 1-5	P	D	Possible causes	RPN	Possible solutions
5	Limits (4)	Not Ok, Ok									
6		Ok, not OK									
7		Nok, Nok									
8	Amp (5)	Noise	X	X		2 Leads to unpredictable movement of source and drift on robot axis	3	5	Electrical noise	30	Shield wiring, or Re-route control signals away from main power cables or Employ high quality wire with good grounding
							2	5	Breakdown of component + electrical noise	20	Compare scaled velocity feedback to voltage out if not = zero either stop or force to zero
9	Gain (6)	Drift	X	X		2 leads to unpredictable	2	5	Temperature variation	20	Employ thermostat controlled fan
						movement of source and drift on robot axis	2	5	Break down of component stability	20	Regular preventative maintenance
10	Encoder (7)	No pulses	X	X	X	5 Unknown position of source	2	3	Break in wire	30	Protect cable or
									Wiring fault in ES stop activation		Redesign wiring at E stop
							2	3	Component failure	30	Employ position/velocity confirmation, or Preventative maintenance

Table 9-4 FMEA chart for 2nd generation model failures (cont.)

No.	Comp/ No	Failure mode	FC1	FC3	FC2	Seriousness 1-5	P	D	Possible causes	RPN	Possible solutions
11		Intermittent pulses		X	X	5 Unknown position of source	4	3	Spurious noise	60	Employ 30 ma current loop circuitry
							2	3	Signal Input faster than count frequency	30	Re-specify counter
											Possible to detect no change in signal counts between sampling times
							1	3	Component failure	15	Employ secondary device to monitor position (integrate tacho signal)
12	Tacho (8)	Not OK, V = 0	**X**			5 velocity of source unknown	1	5	Double point failure caused by Tacho= Encoder when no voltage	25	Employ independent sensor to monitor position or velocity, and use voting to determine correct signals.
13		Intermittent voltage									
14	Mech (9)	Belt break		X	X	5 Lose of source position and velocity info	2	5	Overloaded system Possibly pulley jam	50	Use independent position monitoring device
									Note V not= 0 as it depends on axis		Torque sensor in motor to detect low load conditions
									orientation, Similarly position not known		Position sensor on pulley bearings
									due to axis movement Component fatigue		Preventive maintenance and inspection
15		Pulley Jam Similar to Gear box, motor jam									

Table 9-4 FMEA chart for 2nd generation model failures (cont.)

9.2.4.2 Morphological Analysis and Solution Selection

In BSI EN-292 (BSI[91a]), it is indicated that solutions should be generated to the failure problems identified, and appropriate measures selected. However, no indication is given of how this is to be achieved. To answer this lack of procedure, it was decide to employ a form of Morphological Chart to determine which solutions fulfil the negation of more than one primary and secondary failure function, and a combination of EQFD and Cross Weighted Objective Method to select appropriate solutions.

The advantage of finding solutions, which solve more than one failure function is that, it reduces the amount of work required when applying the solution to the design, as it reduces wasted effort if the negation of each failure was solved separately. The downside of this approach is that, if the same solution is used to solve two different failure problems, the solution becomes coupled to both functions. This goes against the first of Suh's axiom's (Suh[88]), that "the designer should maintain the independence of the functional requirements". The author decided to ignore the axiom, as the morphological chart also has the positive effect of identifying means of redundant sensing for a failure function, so that a few solutions are likely to be coupled across several failure functions.

A good example of this is indicated in Table 9-5, in which a Primary Function is used to describe a grouped set of failure modes, a strategy is then proposed, as a means of realising the Primary Function. A number of solutions, are taken from the FMEA chart (Table 9-4), and added to, if new solutions are thought of, they are placed underneath each Primary Function, and identified as Means. This complies with the application of Vertical Causality and Function Means Trees, described in chapter 2.

The system faults are then placed down the left hand side, and where a fault intersects with a possible solution, the box is highlighted. The sum of the boxes in each column is placed on the lowest row. From this chart, the minimum number of solutions that detect or prevent system failure can be seen. For instance, the Means, "Wire potentiometer and A/D" can be used to detect or prevent 5 possible failures. If this was combined with Shield signal (3 failure solutions), the use of Hardwired detection circuitry (4 failure solutions) and preventive maintenance (5 failure solutions), then all the failure modes present in the system could be detected or prevented. A degree of redundancy is also present in the system as 18 possible failure solutions are given for 14 failure modes.

Means	Encoder spurious noise	Belt Break	On/Off Switch S/C	Emergency stop Open circuit	Strategy	Primary Function
30 ma current loops						Prevent spurious noise Intermittent signals
Frequency to voltage converter + A-D						
Shield signal						
Position control wires away from power lines						
Rack and pinion encoder +					Monitor position of load and compare to encoder signals	Detect Spurious noise Belt break No pulses Intermitten t pulses
Analogue Laser position scanner + A/D						
Wire potentionmeter plus						
Employ double belt and pulley drives						Prevent belt breakage
Increase belt specification						
Replace belt regularly (Preventive maintenance)						
See devices for monitoring load position					Monitor load position	Detect belt breakage
Measure motor output toque					Monitor motor load	
Measure motor output current						
2 Switches in series						Prevent On/ Off /C
error detection lights					Monitor voltage progression through system using time lapses between start up stages	Detect On/ Off S/C E-stop O/C E-stop latch not OK 5 Volt PSU failure
Microprocessor detection						
Hardwire or transistor detection circuitry						
2 switches in parallel						Prevent E-stop O/C
2 lathes in parallel						Prevent E-stop latch not OK
See prevent spurious noise + Improve earth+ quality wire						Prevent Amp noise
Comparison of Velocity F/B with reference voltage						Detect amp noise
Protect wiring						Prevent no encoder pulses & comp failure
Potect component						
Preventaive maintenance						
Respecify Counter						Prevent high input frequency
Limit load velocity						
Monitor 5 volt output (see detect On/Off S/C etc.)						Detect PSU fail
Controlled cooling						Prevent Temp drift
Use of Motor brake						Prevent movement if error detected
Geared or EM brake						

Table 9-5 Morphological analysis for the determination of failure solutions (continued on next page.)

Emergency stop latch Not OK																														
Amplifier electrical noise																														
Encoder failure or wire break																														
Intermittent encoder pulses																														
Pulse Input frequency to fast																														
5 Volt PSU not OK Wire break/ PSU fail																														
Temperature/Volt drift on gain																														
break down of Comp stability																														
Gain																														
Amplifier																														
Number of failures solved	1	2	3	3	5	5	5	1	1	1	1	1	1	1	4	4	4	1	1	2	1	3	5	4	1	1	1	1	3	3

Table 9-5 Morphological analysis for the determination of failure solutions (cont.)

A chart, which is a combination of a second stage EQFD and a Weighted Objectives table, is used to select appropriate solutions, see table 9-6. The solutions which negate multiple failure effects, derived from the Morphological Chart (table 9-5), are grouped together as Primary functions with their RPN, the keywords for describing the functions are "PREVENT" or "DETECT". These are considered as Customer Requirements. A strategy that relates to the Primary Function, derived from table 9-4, is placed alongside the Primary Function. Means for realising the function are then entered according to the Morphological Chart of table 9-5.

Weighting based on technical difficulty (1-5, 5 Hard) is then entered and each solution rated under the following criteria:

- Robustness: - Measure of the ability of the solution to deal with different external conditions. 1 indicates high robustness, and 5, low.
- Systems Redesign: - Indication of how much the design would affect the original design. The intention being to identify solutions which can be added as modules, thus keeping down function coupling and retaining the integrity of the original design. 1 indicates a low degree of redesign and 5, high.
- Implementation Cost: - Ranking, based upon the relative cost of solutions within a particular Means Group as to how much proposed solution will take to implement. 1 indicates low cost and 5 a high cost.

The values for each selection criteria are multiplied to give the Relative Cost for each solution, the solutions with the lowest score generally being selected, indicated by the highlighted box on the right in table 9-6.

From table 9-6, the following solutions were selected.

- 30 ma Current Loops: - Prevents electronic noise, chosen over shielded signals and control wire positioning because these solutions are complementary to each other, giving a combined score of 18 as compared to 12.
- Wire potentiometer + A-D: - Detects electronic noise on sensors, load position, and belt breakage. It is used as part of redundant majority voting system.
- Change belt specification and preventative maintenance: - Used to prevent the belt breaking at inappropriate moments.

Risk Priority Number (RPN)	Primary Function (Whats)	Strategy	Selection Criteria ⇒ / Means (Hows) ⇓	Technical Difficulty	Robustness	System redesign	Implementation cost	Score	Options selected
60	Prevent encoder Spurious noise	Prevent Electric noise	30 ma Current loops	2	1	2	3	12	
			F-V converter & A-D	3	3	3	2	54	
			Shield signal	1	3	1	2	6	
60	Intermittent Signals		Positioning of control wires	1	3	2	2	12	
60	Detect Spurious noise	Monitor Position of load compare encoder signals	Rack & pinion encoder + counter	3	2	3	3	54	
			Laser position scanner + A/D	3	2	2	4	48	
60	Intermittent pulses		Wire Potention-meter + A/D	3	1	2	3	18	
50	Belt break								
30	No pulses								
50	Prevent Belt breakage	Increase strength/ life	Increase belt spec.	1	1	2	1	2	
			Double belt and pulleys	2	1	2	2	8	
			Regular belt replacement	1	1	1	1	1	
50	Detect Belt breakage	Monitor motor load	Measure motor torque	3	2	4	3	72	
			Measure motor current	2	2	2	3	24	
50	Prevent On/Off S/C	Double switches/ latches as required	2 switches in series	1	1	1	1	1	
			2 switches in parallel	1	1	1	1	1	
50	E-Stop O/C		2 latches in parallel	1	1	1	1	1	
30	E-Stop latch NOK								
50	Detect On/Off S/C		Error detection lights + Operator feed back	1	3	1	1	3	
50	E-Stop O/C		Microprocessor detection	2	2	1	3	12	
30	E-Stop latch NOK		Hard wired detection circuits	3	2	1	3	18	

Table 9-6 EQFD and weighted Objectives chart for FMEA analysis (continued below)

Risk Priority Number (RPN)	Primary Function (Whats)	Strategy	Selection Criteria ⇒ / Means (Hows) ⇓	Technical Difficulty	Robustness	System redesign	Implementation cost	Score	Options selected
30	Detect Amplifier noise		Compare Velocity F/B to ref. voltage	1	1	1	1	1	
30	Prevent No encoder pulses comp fail	Prevent comp damage	Protect wiring	1	1	1	2	2	
			Protect component	1	1	1	2	2	
20			Preventative maintenance	1	1	1	3	3	
30	Prevent High input encoder frequency		Respecify Counter	1	1	2	2	4	
			Limit load velocity	0	0	0	0	0	
20	Prevent Gain drift due to temperature	Control temp	Thermostat controlled fan	1	1	1	1	1	
	Prevent Load movement		Motor Brake	1	1	2	3	6	
			Electromagnetic clamp on load carrier	2	1	3	2	12	
			Rack & Pinion geared mechanism	2	1	4	3	24	

Table 9-6 EQFD and weighted Objectives chart for FMEA analysis (cont.)

- Use of parallel and series switches/latches: - Form of redundancy, which is used to ensure that failure of part of a system, is at least double point.
- Error detection lights for the operator and Microprocessor detection: - Used to detect failure of hardwired switches and relays.
- Improve earthing, wire quality and position: - Prevents analogue signal distortion.
- Software comparison of Velocity F/B and reference voltage: - Used to detect amplifier noise, however lag in the system due to system time constants needs to be taken into account.
- General protection of the system: - Guideline for the design to ensure that the system should protected in various ways.
- Limit Velocity: - Prevents the encoder frequency being to high for the encoder to count.
- Thermostat controlled fan: - Prevents gain drift due to temperature changes within the servo control box.
- Employ a motor brake: - To prevent uncontrolled movement of the load when a failure occurs.

9.2.4.3 Failure Analysis Of Additional Components

The solutions identified in section 9.3.4.2 are a mixture of recommendations and subsystems that need to be integrated with the original design. As part of the creation of the new VISSIM failure mode, the subsystems need to be analysed to identify their associated failure modes and effects. The subsystems generated for the new design are Thermostatically controlled cooling fan, a normally closed, Solenoid operated motor brake, a wire velocity/displacement transducer, redesign of the On/Off and E/stop switches and finally a 12 V PSU.

The failure modes and effects for each component are shown in table 9-7. Notes are included down the right hand side, to indicate which modes are mutually exclusive and the reasons why.

Component Description	Operation Modes	Failure effect	Notes
Cooling fan Bimetallic strip on/off temp monitoring	Operation OK	No failure	
	Fan Burn out	Fan not on	Apart from broken contacts , these failures are not mutually exclusive, although the effects are. Can be monitored by measuring temp using simple thermistor or secondary bimetallic thermostat.
	Wire break		
	Contacts break		
	PSU failure		
	Welded Contacts on bimetallic strip	Fan on continually	
Load Brake N/C spring loaded Solenoid mounted on Load carrier	Operation OK	No failure	These failures are not mutually exclusive, although the effects are. If failure does occur use of Velocity/displacement feedback from wire potentiometer or motor encoder can be used determine system state
	Signal wire break	Brake allowed to contact surface, load should stop	
	PSU failure		
	Spring failure	Brake will not contact surface Load will not stop	
Wire velocity/ displacement transducer	Operation Ok	No failure	
	Signal Wire break	Signal out = 0	
	Position wire break		
	V output error due to potentiometer wear	V in error likely to be less than expected	Error will vary dependant upon area of maximum movement
	V out error due to varying input (12V) voltage or noise	V can be greater or less than nominal	This is due to error in PSU or wiring and not in the Potentiometer so this will be modelled as part of the 12 V PSU Monitored by comparison to encoder / tachometer signals
NO On/Off switch	Operation Ok	OK	Mutually exclusive effects
	C/C	Continual Voltage passed	
	Latch fails	When switch is off no voltage passed	
E/Stop switch	Operation Ok	OK	Mutually exclusive effects
	C/C	Continual Voltage passed	
	Latch fails	When switch is off no voltage passed	
12 Volt PSU	Operation Ok	No failure	
	10 % Unstabilised	Random variation of power supply	
	No PSU	No signal , processor would not work, no activation of system	Fail safe condition

Table 9-7 Failure mode and behavioural analysis

9.2.4.4 Results from 4th Generation Model

The components and their associated failure modes indicated from table 9-7 were then added to the 2nd Generation model to produce the 4th. See figure 9-20

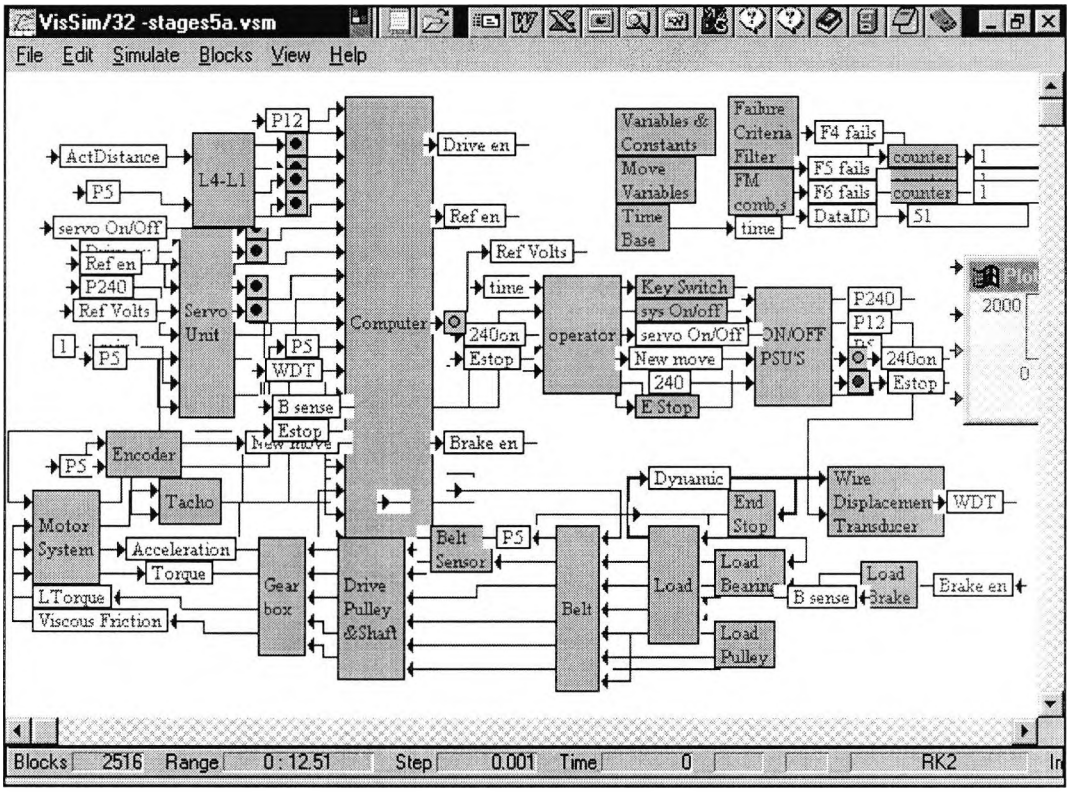


Figure 9-20 Overview of VISSIM 4th Generation Model (Stages4a.VSM)

The failure combination file, consists of 13 components, which encompasses 37 different aspects of component behaviour, this results in 559872 failure combinations. A table identifying each component and its associated behaviour with identification numbers is given in table 9-8.

Component	Comp ID	Function mode	F/mode ID	Function behaviour
12 V PSU	1	OK	0	OK
		Volatge gain	1	5% increase in voltage
5 V PSU	2	OK	0	OK
		Not OK	1	Output = 0
ON/OFF switch	3	OK	0	OK
		Short Circuit (S/C)	1	Output = 1
		Latch Not OK (NOK)	2	Output = 1 momentarily
Emergency Stop	4	OK	0	OK
		S/C	1	Output = 1
		Latch Not OK	2	Output = 1 momentarily
Limits Switches	5	OK / OK	0	Both Ok
		OK /NOK	1	L1 OK, L2 = 0
		NOK/ OK	2	L1 = 0, L2 = OK
		NOK/NOK	3	L1 and L2 = 0
Vref Signal	6	OK	0	Vref = Vout
		Noise	1	Vref = Vout + 5% random noise
Fan	7	OK	0	OK
		NOK	1	Fan not on
		On continuously	2	Fan on continuously
Thermistor	8	OK	0	OK
		No signal	1	Output = 0
		Noise	2	Output = output + 5 % noise
		S/C	3	Output = 5V
Displacement wire	9	OK	0	OK
Potentiometer		NOK	1	Output = 0
		Wear	2	0.95 * Output
Motor Brake	10	OK	0	OK
		Brake on	1	Brake activated continually
		Spring NOK	2	Brake not activated
Encoder	11	OK	0	OK
		No signal	1	Output = 0
Tachometer	12	OK	0	OK
		NOK	1	Output = 0
		Intermittent signal	2	Random intermittence of signal
Mechanical	13	OK	0	OK
		Belt Break	1	No load
		Jam	2	Increase in load so that Velocity=0

Table 9-8 Component and failure mode identification

The failure effects listed in table 9-8, were supplemented by increasing software checking of the various inputs to the system. The software checks employed covered functions to:-

- Startup sequence time out.
- Compare derived velocity and displacement from 3 sources (Encoder, Tachometer and Wire displacement potentiometer).
- Five volt PSU check.
- Temperature monitoring
- Sensing for brake
- Active temperature compensation, to take account of servo amplifier errors due to changes in temperature.

These modifications are shown in STAGES4A.VSM.

The program used to create the failure combination files was revised to include the new failure modes (see FMEASW4A.VSM). The failure analysis was then carried out in the same manner as described for previous models. One single point failure was found for Fc1 (FS414), Noise on the Vref signal , no single point failures were found for Fc2. Two double point failures were found under Fc2, these were caused by a combination of intermittent signals on the Tachometer and both failures modes of the potentiometer., signal=0 and discrepancies due to wear on the resistor. The RFT associated with these failure modes can be seen in figures 9-21 and 9-22.

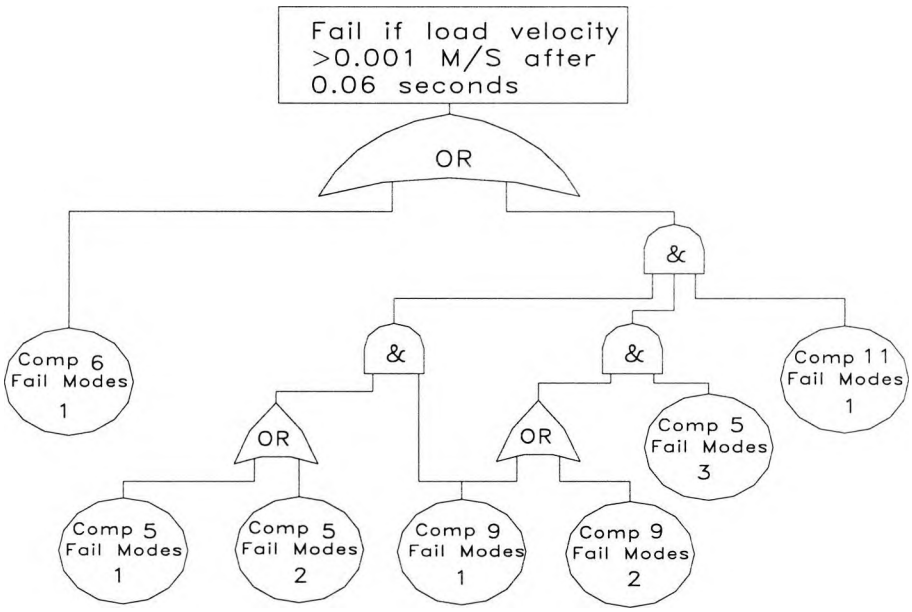


Figure 9-21 RFT for model 4, Fc1

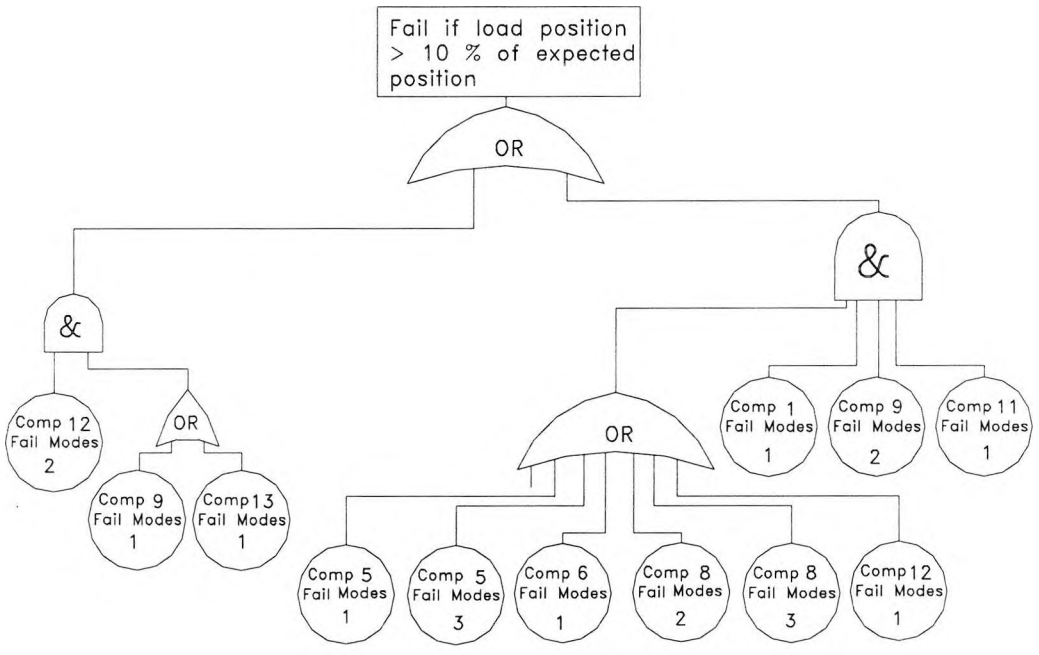


Figure 9-22 RFTs for Model 4, Fc2

Both fault trees show a significant reduction in the number of single point failures, which produce a system failure. The only single point failure present is in Fc1 (FS414.DAT) and relates to a noisy signal on the Vref line. Further analysis was carried out on the amount of noise on the Vref signal, and it was found that the failure only occurred when the noise level was very low. If a large amount of noise was introduced, the discrepancy between expected position and actual position is discovered and the fail line is sent high causing the motor to stop. The 4 triple point failures in Fc1 (FS43FC1.DAT) are due to failures in the Limit Switches, the Wire Displacement Potentiometer and the Encoder. Upon examination of the software coupling between these components, it became clear that the failures were, to a large part, due to the presence of no active signals from the 3 components concerned. Because no signal was received from more than two components, the computer assumed the system was behaving properly, hence the failure.

The fault tree for Fc2 consists of 2 double point (see FS42FC2.DAT) and 6, quadruple point (see FS44F2.DAT) failures. The significant failure mode for both double point failures is a random intermittent signal from the tachometer. When combined with a no signal failure from the displacement potentiometer or the belt breaking, a system failure is produced. The reason for the system failure is similar to the reason for system failure

under Fc1. The components involved in the failure produce no signal, which, when compared to the other sensors, is confirmed as being OK, even though a failure has occurred. The quadruple point failures of the system, under Fc2, is dependant upon failure of the 12 V PSU, the Potentiometer Displacement transducer and the Encoder, combined with various failure modes of the, Limit Switches, Vref signal, Thermistor and Tachometer.

The intermittent failures modes on the Vref Signal and the Tachometer are often the most difficult to remove, as was found when testing the finished robot. Possible solutions to the these failure modes could be, the inclusion of a Low Pass Filter to reduce signal noise and software based, averaging or median sampling, to reduce the effects of intermittent signals. The inclusion of these preventative measures will not be examined as the author feels that the effectiveness of the method in reducing single point failures has been demonstrated.

9.3 Summary and Conclusions

In this chapter, a procedure was examined, which detailed the application of proactive safety led design, with VISSIM failure modelling and function based design methods. The objective was to improve the safety of X/Y axes of the Gamma Radiography, by removing all single point failures. Two failure criteria were set as :

- F_{C1}: -The velocity of the NDT load is greater than +/- 0.0001 M/s after 0.06 seconds from the time that the difference between the perfect model and the actual model (with failure modes) is detected.
- F_{C2}: - The position of the NDT load, under failure conditions, differs by more than ± 10 % of that expected after 0.06 seconds from the time when that difference is first apparent.

The basic procedure for carrying out failure analysis using VISSIM models involved the following stages: -

- FMEA: - Carry out failure mode and effects analysis to determine the modes of failure and failure behaviour of each component in the system.
- Rationalisation of failure modes: - Due to the high number of possible failure combinations (approximately 4 million), and the unreasonably long time which it would have taken to analyse all of them, the failure

combinations and the failure modes were rationalised. This was achieved by looking for modes of failure with duplicate behavioural effects. This reduced the original 4 million failure combinations to 5184, without affecting system integrity.

- Build VISSIM failure model: - The VISSIM failure model is created by developing a perfect model and running it in parallel with a model corrupted by failure combinations. When the output of the perfect model differs from the corrupted model a failure is detected. If the system does not respond to the detection of failure, through its own means, and leaves the system in an unsafe state (defined by Fc1 and Fc2), then that failure combination is logged.
- Breadth first search of failure combinations: - Early work indicated that the time to run all failure combinations and analyse the results was unreasonably long. A breadth first search was applied to the application of failure combinations to the model, so that single failures were identified at the first stage. If a single point failure produced a system failure then the mode that resulted in the system failure was removed from further analysis. Successive levels of failure were then applied to the model, until the level of failure points exceeded the number of components present in that level of analysis.
- Production of Reduced Fault Tree:- Because the failure modes are identified at successive levels of failure, A Reduced Fault Tree can be produced directly from the analysis of the failure log files.

Four failure models were produced. The detection or prevention of failure modes in each of the models differed according to the strategy used to identify solutions to the failure modes:

- Model 1: - Simple model, with basic safety features, representing a system that could have been produced if a proactive design for safety had not been applied. It was used to test the model, the methodology, and provide a benchmark against which further models could be measured.
- Model 2:- Incorporated safety features, indicated by the use of proactive design for safety (CLASH/HAZOP approach combined with EQFD).

- Model 3: - Evolved from model 2, with improved safety features and functions derived from an intuitive approach to design, mainly implemented in software functionality.
- Model 4: - Evolved from model 2, with improved safety features derived from a structured approach to design for safety. The methodology used incorporated FTA, FMEA, Morphological Charts, EQFD and Cross's weighted objective.

9.3.1 Comparison of Results

Table 9-9 shows that the safety of the system was been improved by implementing a more sophisticated safety strategy. The number of wrong side failure modes for F_{C1} was reduced from 13 single point failures to 1 single point (noisy Vref) and 4 triple point failures. The number of failures for F_{C2} was reduced from 12 single point and 2 double point failures to 2 double point and 6 quadruple point failures.

Model No.	Failure Criteria	Points of failure				No. of Failure combin-ations	No of combin-ations tested	% of combin-ations tested
		Single	Double	Triple	Quad-ruple			
1	Fc1	13	0	0	0	5184	15	0.3 %
1	Fc2	12	2	0	0	5184	18	0.35%
2	Fc1	6	1	0	0	5184	53	1.02%
2	Fc2	4	0	0	0	5184	567	11.1 %
3	Fc1	5	0	0	0	5184	407	7.85 %
3	Fc2	4	7	0	0	5184	45	0.87 %
4	Fc1	1	0	4	0	559872	12541	2.23%
4	Fc2	0	2	0	6	559872	6537	1.16%

Table 9-9 Summary of failure analysis results

Model 1 provided the benchmark, against which the other models would be measured. Because minimal safety features were included, nearly all the single point failure modes

produced a system failure. This result was achieved by testing a very small proportion of the failure combination population, 0.3% for Fc1 and 0.35% for Fc2.

Model 2 showed an improvement in safety, a reduction of at least 50% in the number of failure modes that cause system failure. Because the number of single point failures was reduced, a larger number of failure combinations had to be examined. However, the percentage, when compared to all possible failure combinations was still small; 1.02 % for Fc1and 11.1% for Fc2. The high percentage of the combinations examined for Fc2 is due to the robustness of the system, up to 6 point failures examined for this particular case. The improvement in safety is attributed to the use of the structured CLASH/EQFD approach detailed in chapter 7.

Model 3 showed mixed results, with a decrease in single point failures for FC1 and an increase in single and double point failures in Fc2, as this model was a result of applying an intuitive, unstructured software based approach to design for safety, these results were expected.

Model 4 showed further improvement in the safety of the system, when compared to Models 2 and 3. This model was the result of taking a structured approach to design for safety. It proves that the tools used, if applied in an intelligent manner, are worth implementing. It should also be noted that, as the model had few system failures, a larger percentage of the failure combinations were examined, however, this is still very low; 2.23% and 1.16 % for Fc1 and Fc2 respectively. The high level of multiple point failures is unavoidable, as the system relies on checking sensor outputs against one another and the system will fail if enough of these components fail simultaneously, which is highly unlikely.

9.3.2 Conclusions

In conclusion the success of the methodology in significantly reducing the number of failure modes present in model 1 to those in model 4, in a reasonable time, is directly attributable to three factors:

- A proactive stance to failure analysis at an early stage of design, detailed in chapter 6. Proved by the reduction in failure modes between model 1 and model 2.
- The integration of a structured approach to design for safety, incorporating FTA, FMEA, EQFD, Morphological Charts and Cross's Weighted

Objectives. This is proved by the mixed results of model 3, achieved by applying an unstructured approach to design for safety, when compared to results generated from the approach taken to develop model 4.

- The use of computerised tools to determine system failure, automated the search of the failure space, the efficiency of which was improved by applying a breadth first search. This efficiency is achieved by pruning the failure tree at each level of point failure to remove failure modes which cause system failure, the failure space is reduced dramatically as is the time and ease of analysis. As an example of the time savings gained, if all 559872 failure combinations of model 4 had been run, it would have taken approximately 110 hours on a 166 MHz Pentium, not including the analysis time to generate the reduced fault tree. Using the breadth first search this was reduced to approximately 3 hours for fc1 and 2 hours for fc2, including the time to analyse the results and generate the reduced fault trees.

These results complete the proof of hypothesis 1 (stated in chapter 1) , that the integration of risk and failure analysis tools with design function methods can lead to improvements in product safety and reliability. The results of models 2 and 4, in which failure analysis tools were combined with function based design tools, showed that whilst not catching all single point failure modes, they were reduced dramatically. This can be compared to the results from models 1 and 3 in which minimal or unstructured analysis was carried out with poor to mixed results.

The proof of Hypothesis 2, relating to the usefulness of the application of computerised design tools and structured design and failure analysis has been completed. It would not have been possible to carry out such a detailed and efficient failure analysis without the use of a computerised tool, and the improvement in the design for safety would not have been achievable, in the same time span, without a structured approach. This is evident by the improvement in the predicted safety of the system, shown by the difference in the single and double point failures present in model 3 as compared to model 4.

Finally , the results of this chapter, complete objectives 1 and 2, described in chapter 1:-

- Objective 1, the development of a function based design methodology, that employs pro-active safety analysis features which incorporates computer

based tools for component selection and failure analysis. This objective was started in chapter 6, when the CLASH, EQFD approach was applied to carry out proactive design for safety. The objective is completed in this chapter as it details the use of computer-based tools for failure analysis.

- Objective 2, to carry out a case study based upon the design and failure analysis of the gamma radiography robot, is also met by the completion of this chapter, which describes the fault and failure analysis of four different models of the X/Y axes of the gamma radiography robot.

The final objective, stated in chapter 1, to compare the methodology described in this thesis with the methodology used for the original case study (Chamberlain[95,96]), will be carried out in chapter ten.

10 Summary of results and conclusions

10.1 Introduction

The intentions of this chapter are threefold, firstly, the work carried out in the case study, detailed in chapters 7, 8 and 9 will be summarised. This is so it can be compared to the tools and methods used in the original pilot study for the design of the gamma radiography robot, which will also be briefly described. This fulfils objective three, stated in chapter 1. Objectives one and two, relating to the development of a safety led, function based, computer integrated, design methodology and a case study showing its application are detailed and attained in chapters 7, 8, and 9. The method being supported by a literature review detailed in chapters 2,3, and 4.

Secondly, both of the theses, presented in chapter one, will be examined in relation to the objectives and the work carried out. The theses relate to the benefits achieved by the integration of failure analysis and computer tools with design function methods.

Finally, areas of new work will proposed, based upon the increased use of computer based tools for design, principally, fault analysis, component and concept selection.

The fulfilment of these objectives complete this thesis, providing a synopsis of the work carried out, and evaluating it in relation to the three objectives and two hypotheses described in chapter 1.

10.2 Assessment of pro-active safety led design for mechatronics

To asses the effectiveness of the methodology outlined in this thesis, it is necessary to compare it to a similar methodology which has been employed previously. The case study selected for comparison is the original pilot study (Chamberlain[95]). The pilot study methodology was selected as the author worked on the original project and was familiar with the methodology and its shortcomings.

10.2.1 Method 1: DFD/ACP and intuitive FMEA, FTA

A flow diagram showing the stages in the design methodology employed for the pilot study is shown in figure 10-1 (Chamberlain[98]). The principle tools used in the pilot study methodology were, Design Function Deployment (DFD), Quality Function Deployment (QFD), Activity Channel Pool (ACP) diagrams, Fault Tree Analysis (FTA), Failure Modes and Effects Analysis (FMEA) and VISSIM modelling. Design

Function Deployment provides the structure to the methodology, with the other tools supporting.

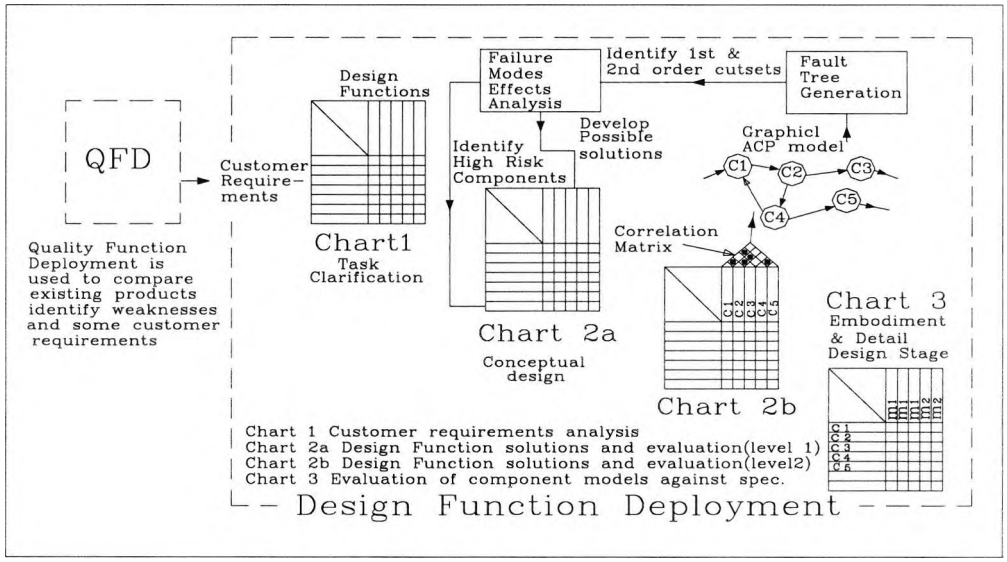


Figure 10-1 Design Methodology used for Pilot Study

The first stage of the design process is a QFD analysis, used to compare various top level concepts and dismiss inappropriate solutions. This is achieved by ranking solutions against a set of high level customer requirements.

These customer requirements are used to define top level functions. The relationship between the requirements and functions were mapped to produce a 1st stage DFD chart, from which technical difficulty and customer importance ratings can be extracted.

In a stage 2a chart, the customer requirements are divided in explicit and implicit requirements, and the functions are expanded into primary, secondary and tertiary levels. This expansion of information leads to a chart similar to that shown in figure 10-2. A form of Cross's weighted Objectives method is applied in the chart and used to select detailed solution concepts.

The solution concept selected from chart 2a is then expanded upon to find the means of realisation. This is achieved by selecting a variety of components that are evaluated against a further set of customer requirements. These customer requirements are typically more detailed than the requirements stated earlier in the design process, because the they are more defined. For example, a requirement for fast acceleration can be broken down into, low inertia matching between load and driver and high torque developed from the driver.

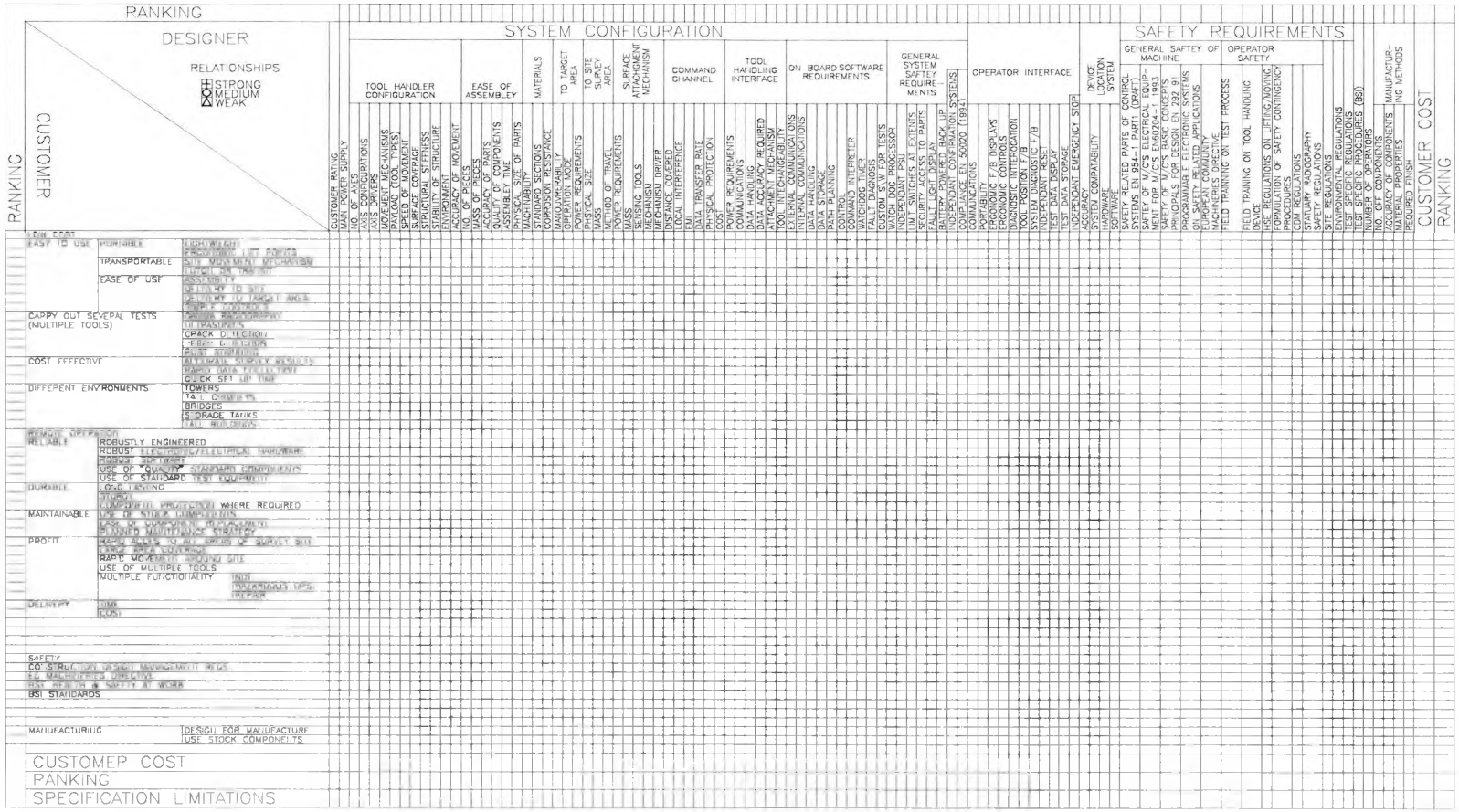


Figure 10-2 Incomplete stage 2a DFD chart

A correlation matrix is used to define the relationships between the system components, defining energy and signal relationships. These relationships form the correlation chain, which can be used to analyse the system and select the appropriate components (Cominicini[94]).

The correlation chain can also be used as the basis for a paper based ACP model (Bleakley[97]) of the system. This combines the logical model of the control system with the behavioural model of the physical components. From this model, the complete system behaviour can be defined.

By applying failure criteria to the paper based ACP model, a Fault Tree can be generated, the ACP model providing the designer with the information relating to the interaction of components and system behaviour. From the fault tree the first and second order cut sets are derived by observation. The next stage is an FMEA analysis of the most important failed components, this is used to identify the cause of failure of various components. The negation of the component failures are entered as customer requirements into a stage 2a DFD chart, and possible solutions entered as functions. The process is then repeated until the desired level of safety is achieved.

VISSIM models, derived from the ACP diagrams, were also employed as an aside to the above methodology. The models were used to check the safety of the original design. This proved to be a useful exercise as many functional errors were found in the original design. These were corrected using the VISSIM model, and further safety analysis using an intuitive approach. This work led to the application of VISSIM modelling for component selection and failure analysis.

The main problem associated with this methodology was the large size of the DFD stage 2 charts, which were found to be awkward to use and laborious to create. It was felt that a means of controlling the charts was required.

Although FTA and FMEA were employed successfully during the design life cycle, their application cannot be considered as proactive, as they were used once the first generation system design was almost complete. This led to the requirement for a method which took safety into account at the very beginning of the design process.

The application of a modelling tool was found to be effective and it was felt that the role of the system model was under-utilised, especially with respect to automating some aspects of the design process.

10.2.2 Method 2: CLASH led EQFD with computerised fault analysis

The pilot study provided the opportunity to investigate design, failure analysis and modelling tools, however, it was not felt to be as successful as it could have been, hence the development of CLASH led EQFD with computerised component selection and fault analysis. To improve the pilot study methodology, a number of tools were examined in the fields of design, failure analysis, software engineering and system modelling.

The main design tools selected (see chapter 2) for incorporation into the case study methodology were, Quality Function Deployment (QFD) and Enhanced Quality Function Deployment (EQFD). DFD as applied according to Edney[93], was essentially EQFD. These methods provided the basic structure for the design life-cycle. The law of Vertical Causality(Hubka[84]), Cross's Weighted Objectives (Cross[89]), Function Means Trees (Buur[90]) and Morphological Charts (Zwicky[48]) were chosen to control the use of EQFD.

ACP (Mort[90]) diagrams, combined with VISSIM models (Bleakley[97]) were selected as the means to model system behaviour and computerise component selection and fault analysis (Chapter 3).

Finally, Consequence Led Analysis of Safety Hazards (CLASH), (Seward[96]), combined with FTA and QFD, was selected to identify possible safety hazards and possible solutions early in the design process. FTA and FMEA were selected as supplementary methods required by the computerised failure analysis to depict results and analyse component failure behaviour. This is detailed in chapter 4. The basic process can be seen in figure 10-3, in which the various tools used are shown in relation to each other.

As mentioned above, CLASH was used to identify possible safety issues relating to the general design of the Gamma Radiography robot. This identified varying degrees of uncontrolled movement of the X and Y axis as being the worst safety hazard. FTA was then carried out to determine how uncontrolled movement could be caused. A QFD analysis of possible solutions was then carried out concurrently with the CLASH/FTA analysis. The negation of the causes of failure, identified by the CLASH/FTA, are entered as customer requirements into a stage 1 EQFD chart, combining with the customer requirements from the QFD analysis.

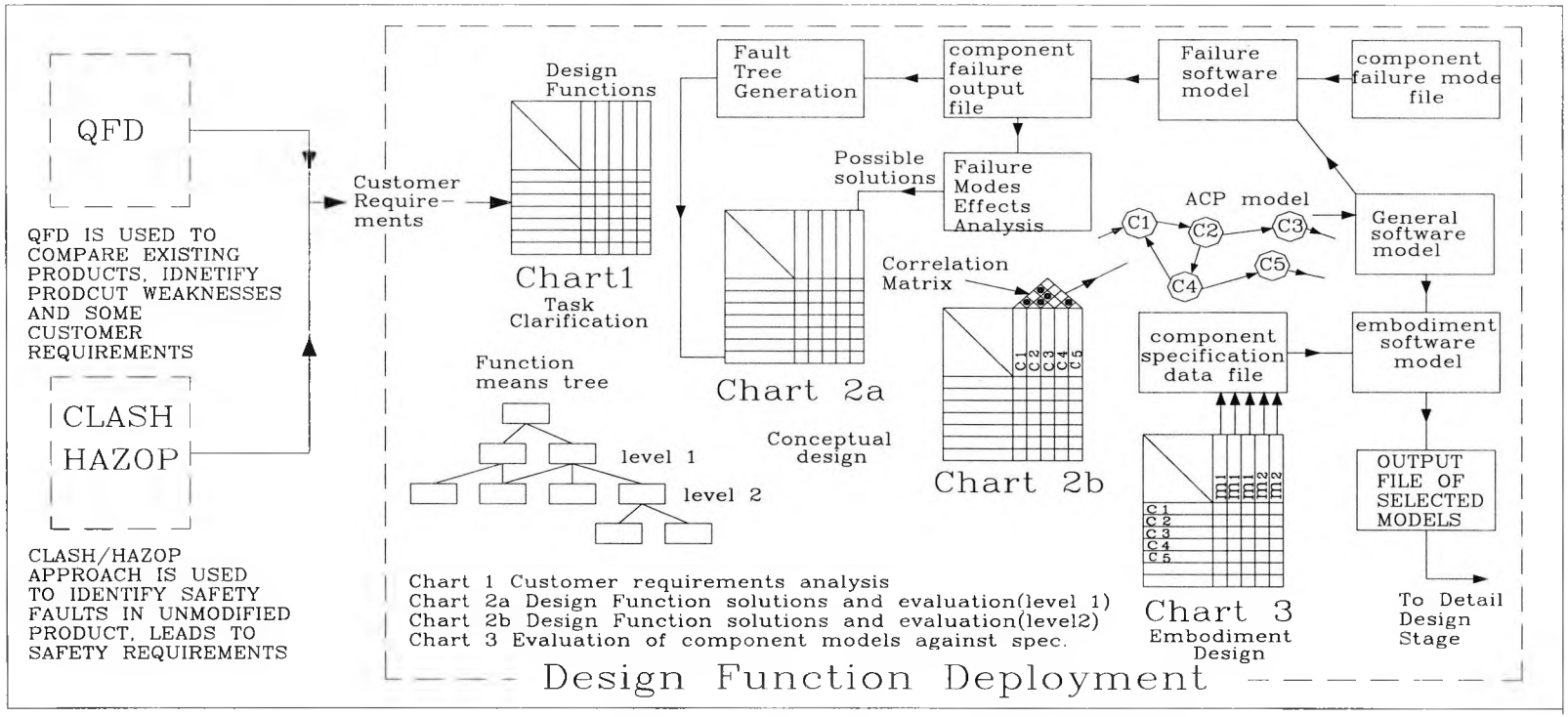


Figure 10-3 CLASH led EQFD with computerised analysis features

The Stage 1 EQFD analysis is then carried out, the process being identical to that applied in DFD. The results from the stage 1 analysis are then passed to a stage 2a analysis. However, instead of expanding functions wholemeal into primary, secondary and tertiary functions, the partition of functions is controlled by the law of Vertical Causality. The essence of this law is that a function cannot be reduced into sub-functions until a means for releasing the function has been identified. This controls the stage 2 analysis so that it can be mapped using Function Means Trees. Morphological Charts and Cross's Weighted Objectives method were employed at this stage to generate and evaluate, concept solutions. This process is detailed in chapter 7.

Computerised component selection is described in Chapter 8. A correlation chain is created from the roof of the house of quality, providing the basis of the ACP diagram, which is the first stage in creating the software model of the system. System behaviour at the Pool level can be described in a number of different ways, State Diagrams, Flowcharts, equations or pseudo-code (demonstrated in chapter 8), depending on the experience and requirements of the designer. Once the ACP model has been created, and the system behaviour defined, the VISSIM model for component selection can be developed. This incorporates a series of component databases, successful log file, failure log file and component combination file. All the combinations of components are tested against a specification criteria, and if successful they are logged. To reduce the analysis time, staged analysis is carried out. This is similar to the way in which a designer would do rough calculations to select an approximate range of components, prior to carrying out detailed analysis.

Finally, once a set of components have been selected using the VISSIM model, it is expanded to include all the components and safety methods identified in the Stage 2 EQFD analysis for safety. A simplified FMEA is carried out to identify failure modes and failure behaviour. This behaviour is then incorporated into the model as selectable component behavioural states. By selecting various failure states, and comparing the system output against a perfect system model previously specified, system failure can detected. A breadth first search, ranked by severity, of the failure combinations is carried out to increase the efficiency of the search of the failure space and to simplify analysis. This results in a Reduced Fault Tree, removing the need for a full Fault Tree to be drawn and analysed. Once a Reduced Fault Tree has been produced, a detailed FMEA can be carried out to determine possible causes of system failure, and identify

possible solutions. A variation on Morphological Charts were then employed to identify possible solutions, which solve more than one failure problem. This allows redundant checking to be integrated into the failure detection mechanisms. Each of the solutions are then evaluated against a set of requirements in a stage 2 EQFD/Cross weighted Objectives Chart. The solutions selected are then further evaluated to determine failure mode behaviour, this is incorporated into the VISSIM model and the failure analysis repeated. This process of failure analysis, system redesign and testing is repeated until the required level of safety is reached. Chapter 9 details the application of this method to the safety analysis of the X or Y axis of the Gamma-Radiography robot.

10.2.3 Comparison of methods 1 and 2

The method used for the case study was more sophisticated than that used for the Pilot Study, and it resulted in an improvement on the Pilot Study methodology.

Whereas the pilot study method did not include a true pro-active safety analysis, the case study did. It was shown in chapter 9 that the use of the CLASH/FTA method resulted in less single point failures than were present if no proactive safety analysis was carried out. This was proved by applying the computerised failure analysis technique, discussed in section 10.2.2, to a systems model, that could possibly be produced by an inexperienced designer (1^{st} generation model), a model based on the CLASH/FTA approach.

The use of Cross's Weighted Objectives method, Morphological Charts, the Law of Vertical Causality, and Function Means Trees when combined with EQFD proved very successful. They controlled the size of the charts and gave logic to the design process without stifling invention. They also enabled technology allocation to be carried out easily, as numerous solutions covering mechanical, electronic and software were examined and evaluated. This process was detailed in chapter 7.

The ACP diagrams, derived from the correlation chain, (used in both methods) were effective in identifying the direction of energy and signal flows between subsystems. However, the formal notation of Psuedo-Code, for pool definition, was laborious to carry out, and methods appropriate to the designers experience and the system requirements should be employed. The ACP diagrams also provided a useful starting point for the VISSIM models. Again this process is described in chapter 7.

The application of VISSIM models for computerised component selection was also successful, once the model was produced, with 216 combinations of components evaluated in approximately 15-20 minutes. The model took a long time to develop, but once the theory is known, the application of VISSIM to component selection becomes considerably easier.

The author is aware that Excel, or some other maths based software, could have been used for this stage of the analysis, but it is felt that the time taken to develop the appropriate macros would be greater than the time taken to develop a VISSIM model. Another advantage of a VISSIM model over Excel, is that an Excel model is a steady state model, whereas the VISSIM model is dynamic and can be user interactive.

The original calculations to determine the system components took approximately one month, and involved carrying out several sets of complex calculations by hand. Although the same combinations of components were selected by hand, as were selected by the computerised method, it was felt that the VISSIM model was considerably more efficient. Also, the time spent building the model and understanding the system was very useful, especially in light of the fact that the component selection model provided the basis for the failure model. This process is explained in chapter 8.

Although FMEA and FTA were integrated into DFD for the Pilot Study, it was felt that the analysis could have been improved by the application of computer based failure analysis and structured design methods. The Case Study dealt with this problem by developing VISSIM models for failure analysis, the failure models, derived from an FMEA, was used to update the VISSIM component selection model.

The process of carrying out the computer based analysis was very efficient, identifying combinations of failures, which the designer could otherwise possibly overlook. Not an easy task as there were 5184 failure combinations in the 1^{st}, 2^{nd} and 3^{rd} generation models and 559874 in the 4^{th} generation model. The application of a breadth first search to the failure space was the most important development toward making this an efficient process. It reduced the program run time from several days to hours in the case of the 4^{th} generation model (166 MHz Pentium being employed) and made the analysis of the results considerably easier.

The integration of structured design tools, such as, Cross's Weighted Objectives, and Morphological Charts with FTA and FMEA proved successful. Previously, guidelines such as, BS EN 954-1(BSI[92]) indicated that several options should be evaluated to

prevent or detect failure, but no guidelines are given as to how this could be achieved. In this instance, the application of the above methods resulted in a improvement in system safety. This is evidenced by the reduction in single and double point failure modes between the 2nd generation model and the 3rd and 4th generation model. A mixed result was obtained from the 3rd generation model, which was produced using an unstructured intuitive approach, with little overall reduction in failure modes between the 2nd and 3rd generation model. The 4th generation model, derived from a structured approach of integrated design and failure analysis tools, reduced the combined number of double and single point failures from 10 and 1, respectively in the 2nd generation model to 1 and 2 in the 4th generation model.

The dramatic reduction in failure combinations, between the 2nd and 4th generation model, which produce system failure is directly attributable to the application of a structured design method integrated with computerised failure analysis. It is unlikely that such an improvement in system safety would have been achieved if an intuitive, unstructured approach to design for safety was taken.

10.3 Results related to hypotheses and objectives

To test the two hypotheses stated in chapter 1, three objectives had to be reached, which when combined, provided the basis for the proof.

- Objective 1:- Develop a function based design methodology which employs pro-active safety analysis features. The methodology to be compatible with novel computer modelling tools for component selection and failure analysis.

The possibility for this is apparent in the literature review, detailed in chapters 2, 3 and 4. These chapters examine, tools and method used for engineering design, system modelling and fault analysis, respectively. From these chapters a number of tools are selected which enable a design methodology to be developed. Novel computer based tools for component selection and failure analysis were developed as a result of work carried out in the original Pilot Study. Examples of the use of these tools and how they are integrated are given in chapter 5.

- Objective 2:- Carry out a case study based upon the design and failure analysis of two axes of the gamma radiography robot.

The second objective is achieved by the application of the design methods and tools detailed in chapter 5, to the design and failure analysis of two axes of the gamma radiography robot. Chapter 6 defines the system requirements and safety specification. In Chapter 7, pro-active safety analysis is completed, providing input to the EQFD design process, from which concepts are generated and selected. VISSIM models for component selection are developed in chapter 8, these used to select specific components for the system. Chapter 9 demonstrates the application of computerised failure modelling, integrated with failure analysis tools and structured design methods.

- Objective 3:- Evaluate the success of the new methodology by comparing it to the Pilot Study methodology, originally used for the design of the product.

The final objective is attained in section 10.2 of this chapter, the Pilot Study methodology is described with its shortcomings. After this, the Case Study methodology, described in chapters 7, 8 and 9 is summarised, and compared to the Pilot Study methodology.

10.3.1 Hypothesis 1:Integration of failure analysis and design methods

The first hypothesis relates to the usefulness of the integration of failure analysis and design tools, the complete hypothesis stated below:-

> The integration of risk and failure analysis with design function methods can lead to improvements in new mechatronic product safety and reliability.

The hypothesis is proved positive by examining the results obtained when proactive failure analysis tools and FMEA/FTA are integrated with EQFD. This process is detailed in chapters 7 and 9.

In summary, the CLASH/FTA approach to proactive safety based design was successful, this claim is supported by the 60-70 % reduction in single point failure modes (see table 10-1, taken from chapter 9), between a model based upon a design with few safety features (1st generation model) and the model created as a result of applying CLASH/FTA integrated with EQFD (2nd generation model).

Model No.	Failure Criteria	Single point failures	Double point failures	Total No. of Failure combinations	No of Failure combinations tested	Percentage of failure comb-inations tested
1	Fc1	13	0	5184	15	0.3 %
1	Fc2	12	2	5184	18	0.35%
2	Fc1	6	1	5184	53	1.02%
2	Fc2	4	0	5184	567	11.1 %
3	Fc1	5	0	5184	407	7.85 %
3	Fc2	4	7	5184	45	0.87 %
4	Fc1	1	0	559872	12541	2.23 %
4	Fc2	0	2	559872	6537	1.16%

Table 10-1 Summary of results for computerised failure analysis

Further improvements in safety were obtained by combining Cross's Weighted Objectives and Morphological Charts with FMEA, FTA and EQFD. FTA was used to identify failure behaviour, after which FMEA was employed to identify possible cause of failure and solutions. Next, a Morphological Chart was generated to identify redundant solutions to failure modes, and Cross's weighted Objectives/EQFD applied to select appropriate solutions. The solutions are included in the next generation model, which is then analysed. If necessary this process is repeated until the required level of safety has been achieved.

By taking this structured approach (4th Generation model), the number of single point failures were reduced from 6 single and 1 double point failure (under Fc1, 2nd generation model) to 1 single point failure (Fc1, 4th generation model). Under Fc2, the number of failure modes were reduced from 4 single point failures (2nd generation model) to 2 double point failures (4th generation model). The number of multiple failures increased

in the 4th generation model as expected, due to the increased redundancy of components in the system. An intuitive approach was also attempted, which employed FTA, resulting in the 3rd generation model, for which no improvement in safety was achieved.

10.3.1.1 Benefits of applying CLASH led EQFD

Although the integration of FTA and FMEA with EQFD has been attempted previously (RFCA, Sontow[93], reviewed in chapter2), its application is extremely complex. It is not proactive in the sense that CLASH led EQFD is, and it does not employ any computer based tools to carry out the failure analysis, this being achieved by hand. It was observed that the early failure analysis stage carried out in RFCA is very reliant upon the experience of the designer in being aware of system faults. It was for these reasons that CLAS led QFD was developed.

The awareness of system failures required for RFCA, is not necessary for CLASH, because it is initiated by the designer examining key words relating to accidents, and examining how they may happen. The problem with employing CLASH/FTA is that it is possible that the designer may produce predisposed customer requirements. However, this is a double edged sword as the user may identify a particular safety hazard for which there may be only one HSE recommended solution. In this instance, a predefined solution forces the designer to take notice of it and employ the said solution.

Cross's Weighted Objectives, Function Means Trees, Morphological Charts and the law of Vertical Causality are all useful techniques which achieve the same results more readily than the tools described in RFCA, integrating seamlessly with EQFD.

The integration of FMEA with EQFD, demonstrated in case study, was easily achieved this was mainly due to the fact the FMEA is matrix based, as are the majority of the design tools used in the study. Because the tools employed for CLASH led EQFD are, on the whole, matrix based. An easy to understand, record of the design process is also created. The only problem is in the ranking analysis, the user must be honest when comparing and ranking different concepts, not allowing personal bias to overcome the assessment.

Early work, based on the Pilot Study, also highlighted the usefulness of employing a less sophisticated approach, employing ACP diagrams, FTA and FMEA to carry out fault analysis. The ACP diagrams provide the designer with a system model which allows failure propagation to be mapped. This is useful for Fault Tree production as it

indicates which components are affected by which faults and gives an indication of behavioural propagation.

10.3.2 hypothesis 2: Integration of computer tools with the design process

The second hypothesis is the proposition that:-

> The integration of computer based models for component selection and failure analysis is feasible and beneficial to the design process.

Two computer based tools were applied to the design of the Gamma Radiography robot. The first is the application of VISSIM models for component selection and the second its application for fault analysis. The model behind both computer tools is based upon the ACP diagram derived from the correlation chain.

10.3.2.1 The benefits of computerised component selection

The development of Computer based tools for component selection were described and demonstrated in chapter 8. From 216 possible component combinations, one combination was selected for both the X and Y axis. The combinations selected were the same as the combinations selected, when the calculations were carried out by hand for the Pilot Study. However, the computer model for component selection completed the calculations in a fraction of the time. A number of advantages, other than reduced calculation time, were realised by the use of computerised component selection:-

- The use of the dynamic models for component selection, effectively automates the correlation chain, a process not previously reported in the literature.
- The application of staged specification and evaluation allows the solution space to be reduced progressively. This improved the efficiency of the program, reducing the analysis time from approximately an hour to 20 minutes (on a 486, 75 MHz PC), in the case study. Again, this is a process which has not been reported in the literature.
- The model can be manipulated to test the effect of different variables on component selection, such as viscous friction or efficiency, with ease.

- The model is interactive with the user, enabling operator inputs or variables to be changed at run time, the effects being observable by the user.
- Software, electronic and mechanical components are all modelled using the same tool, in spite of the different energy flows.

As a result of creating a systems model the designer gains a deeper understanding of the system dynamics and behaviour. This helps later on when the model is expanded to include failure behaviour.

The author feels that these observations, coupled with the fact that dynamic computer models for component selection selected the same component combination as determined in hand calculations in a fraction of the time, proves the first part of hypothesis two, relating to computerised component selection.

10.3.2.2 The benefits of computerised failure analysis

The development and application of the VISSIM model for failure analysis was described in chapter 9. Because of the nature of failure analysis tools, it was not possible to directly produce a computer model that automatically generated RFT's and FMEA's. This is because such a system would require the integration of a large knowledge base of standardised VISSIM component failure models (which do not exist), with an Expert System program and VISSIM. The development of such program was beyond the scope of the original work.

The most important aspect of computer-based modelling for failure analysis was found to be the application of a breadth first search algorithm to the failure space. Because of this, it was possible to reduce the number of failure combinations examined and to generate Reduced Fault Trees, efficiently and readily by examining the failure logs for each stage of the analysis. It would have been impossible to carry out the extensive analysis detailed in chapter 9, if the computerised failure models described in chapter 9 were not developed and integrated with the design tools, FMEA and FTA. It was found that the application of the methodology was completely successful, removing all but one single and two double point failures from the system.

A number of observations were made on the benefits of computerised failure analysis.

- Because VISSIM is a graphical dynamic modelling tool, which enables all forms of component to be represented, it is possible to model failure of, mechanical, electronic and software components in a continuous manner.

This is form of modelling has not been previously been reported in the literature, the general case being to employ coarse grained failure models .

- The application of a breadth first search at each failure level, which has not been previously reported in the literature, was essential for the success of computer based failure analysis described in chapter 9. It allowed the failure tree to be pruned at different stages, removing failure combinations that produce failure, dramatically reducing the amount of combinations which need to be tested.
- Because a breadth first search strategy was applied, it allowed the author to generate Reduced Fault Trees directly from the failure logs at each stage. This bypasses the necessity for a full Fault Tree analysis, which would then need to be reduced to elicit the essential details (minimum cut sets). Again this process has not been previously reported in the literature.
- Because the failure analysis was computerised, it was possible to examine and evaluate extremely complex interactions between software, hardware, electronics and mechanics. A process which would have been impractical by other means.

The observations, described above, combined with the reduction in failure combinations due to the application of VISSIM failure modelling, completes the proof of hypothesis 2.

10.3.3 Cautionary observations

Despite the success of the methodologies and tools described in this thesis any user of - them should be aware of the following points relating to their application:-

- CLASH led EQFD cannot make a claim as to being a panacea, with the user exactly following the sequence of activity described in this thesis, for the design of safety critical mechatronics. The methodology proved successful because the author understood the methods selected and employed them because they were appropriate to the problem. Future users of the methodology should be aware that the choice of methods may be product/problem specific.

- In a similar vein, the results from software models for component selection and fault analysis should be accepted with caution. If the user does not attempt to gain a deep level of understanding of a system, through this questioning process, it will not be possible to ensure that the model is correct or the user, capable of generating effective solutions to detected failure modes.
- Where possible, a structured approach to design, whether it is for conceptual or failure modes reduction, should be carried out at all times. This is because it forces the designer to confront all the issues relating to the system, in a logical manner. Evidence of the failure of intuitive or unstructured methods is present in the results from the 1st and 3rd generation models. Contrastingly, structured methods such as Morphological Charts and Weighted Objectives were found to be especially useful for concept generation and selection.
- When creating dynamic models, using programs such as SIMULINK or VISSIM, the user should be aware of creating models which are too detailed. This can easily happen, as the act of model creation is very absorbing. Excessive detail detracts from the efficiency of the model and makes it over complex.
- When employing ACP diagrams the designer should consider them as a meta-modelling tool in which the system behaviour, buried at the pool level, should be represented by whatever method the designer is comfortable with, or represents the system behaviour best. This is because the formal notation, presented in this thesis to describe the system behaviour, is laborious to produce and can be difficult to understand if the user is not familiar with the behaviour of the system or subsystems.
- The mechatronics experience of the designer also needs to be taken into account when employing CLASH led EQFD for mechatronics design. This is because the user must have a good knowledge of the functionality available from the disciplines of, mechanics, software, electronics, and their integration. This knowledge should also cover an awareness of the advantages and disadvantages of employing various concepts. If this

knowledge is not present, then it would be impossible to judge whether a solution is capable of realising its functionality.

- Finally, the designer should be aware of common techniques to avoid system failure and know when they are appropriate. Guidelines PES[92] indicate the use of measures, such as, redundant sensing, watchdog timers and majority voting systems as general solutions to all problems, without considering whether they are necessary.

10.4 Novelty of work

This thesis details a new design methodology for proactive, safety led design of mechatronic systems. It employs a number of design and failure analysis methods which integrate with EQFD. As an aid to the design process two computer based tools were developed, one for component selection and the other for fault analysis. Both tools relied upon a computer model, of the system under development, as a means of evaluation. Based upon the observations made in this chapter and the information contained in the thesis, two main areas of novelty are identified.

1) The integration of new computerised tools into EQFD.
2) Design tools, integrated into the QFD family of design methods.

These main points and the benefits of their application to design problems are discussed in section s 10.4.1 and 10.4.2

10.4.1 The integration of new computerised tools into EQFD.

The authors work integrates computer models into the EQFD design structure and uses them to carry out component selection and failure analysis. This is a significant advance on the state of the art reported in section 1.3.1. The benefits of this innovation are threefold

1) The efficiency of the design process is improved by employing a computer to carry out searches of large component databases. The application of a staged process of specification development for the system also improves the efficiency of the process. Each stage of the specification development reduces the components present in the solution space.

2) All viable combinations of component failure modes which led to system failure are systematically tested and evaluated using a breadth first search algorithm. This logically reduces the number of failure combinations to be tested, and improves the efficiency of the analysis, removing the need for Fault Trees and minimum cut set analysis. Reduced Fault Trees can be generated directly from the failure logs.

3) It adds to the knowledge already in place about the application of computer-based tools to the QFD family of methods, paving the way to realisation of computerised DFD, based on the development of virtual models which can be used to evaluate potential solutions.

10.4.2 Integration of combined design tools into EQFD

The observations made in section 1.3.2, regarding the integration of failure analysis tools with the QFD family of methods, indicated that, a true pro-active risk/safety analysis approach has never been successfully attempted. This research thus contributes in the following areas:-

1) CLASH and HAZOP have been successfully employed to define possible safety hazards and enter the solution of these hazards as customer requirements.
2) FMEA and FTA have been successfully integrated into EQFD, in a manner similar to that proposed by Sontow and Clausing, but with greater effectiveness due to the simpler nature of the methodology.

A Further point of novelty is the generation of a system model which directly relates to the EQFD house of quality and the correlation chain. This thesis identified a direct relationship between the roof of the EQFD house of quality and ACP diagrams. From these simple models, intuitive fault analysis can be carried out. They also provide the basis for the computer model, as they layout the general structure of the components and the flows of energy and signals.

The two main advantages of employing these methods were found to be:-

1) Reduced iteration in the design process by incorporating safety and quality into the product at the beginning of the design life-cycle. Safety in the design of many products is often added as an afterthought to the design process. If these iterations are being carried out when the product design has nearly been completed the knock on effect of changes is likely to be very expensive and the proposed solutions barely adequate.

2) By adopting ACP diagrams or any modelling technique, as part of the design process, it is possible to generate a deeper understanding of the relationships which exist between components. This is especially useful if intuitive fault analysis is being carried out, for an understanding of the propagation of faults within a system and if a comprehensive fault tree needs generating.

10.5 Recommendations for future work

The use of the QFD family of methods and the importance of customer requirements seems to be well understood by some design companies. However, the application of these methods has been mainly paper based. It is for this reason that further work needs to be carried out in computerising the methodology, eventually realising the ideas described in Sivaloganathan[95] of a completely integrated design package, such as DFD. To achieve this aim, work needs to be carried out in the following areas:-

- Integration of Knowledge bases and Artificial Intelligence to propose and evaluate suitable concepts. In effect, an automation of Morphological charts based upon energy flows and transformations between standard system elements. This would reduce the need for the stage 2 EQFD chart.
- Although not demonstrated, it is possible that cost information and physical specification details, such as, thread sizes and pitch circle diameters could be included in the analysis, to aid matching of components.
- Improved algorithms for component selection and fault analysis, possibly employing Genetic Algorithms (Mc Crae, Bleakley[99]) or Taguchi techniques.

- Investment by component producers to produce compatible format descriptions, of their products, including operational characteristics and failure modes.
- Development of an Object based modelling standard, with defined inputs and outputs for standard components, leading to the creation of large databases of components. This would simplify the development of system models as they could be created in a modular fashion. This development is imperative as the simulated analysis of systems, using tools such as Simulink or VISSIM, before expensive prototypes are built, is becoming more common.
- The development of Patterns (Software term used to define a template which encapsulates a generic task), for engineering design concepts. These can be used to develop standard software models using classes, defined by their transformation function, and specific components as objects, thus enabling full automation of component selection and failure analysis, if integrated with Artificial Intelligence techniques.

On a less grand scale, the methods and tools detailed in the thesis could be expanded upon in the following ways.

- The effect of Velocity profiles, other than trapezoidal or triangular could be analysed with ease and system elements sized and selected accordingly. This would be especially useful if designing ,for example, a 3 axis revolute, robot, in which the system inertia, seen by the motor, changes constantly, depending upon the position of the gripper in the robot work space and the motor accelerations.
- Because detailed FMEA is combined with computerised failure analysis and the results of failure can be monitored, it is possible to generate operator instruction manuals that detail system failure effects from which failure causes can be identified. This could be further developed into Knowledge Base maintenance and repair system.

The technology to enable many of the advances, described above, already exist. However to achieve them, it is important that support is provided from component suppliers, and the tools developed, are taken up by industry.

References and Bibliography

Abrial J.R., (82), "The specification language Z", Technical report, Programming research Group , Oxford University, 1982.

Adamantopoulus M.,(96), "Risk Analysis and Risk Management in Constructions", M.Sc. Thesis, City University, Civil Engineering Dept. 1996.

Adelson B., & Soloway E.,(84) "A cognitive model of software design", Report 342, department of Computer Science, Yale, 1984.

Ahmed S., Braek R., (97), "Modelling the behaviour of mechatronic systems", 11th International Conference on Engineering Design, Tampere, Finland, Vol 3, pp 741-744, 1997.

AkaoY., (90),"Quality function deployment: integrating customer requirements into product design", Productivity press, 1990.

Andreasan M.M. & Hein L.(87), "Integrated product development", IFS Publications, Bedford, England, 1987.

Andreasan, M.M., (80), "Syntesemetoder pa systemgrundlag", Dissertation, Lunds Tekniska Hogskola, 1980.

Archer L.B.,(84), "Systematic method for Designers", in Cross N. (ed.), "Developments in Design methodology", John Wiley & Sons Ltd. London, 1984, pp 57-82.

ASME, (86) "Goals and priorities for research in engineering design", American Society of Mechanical Engineers, New York, USA. 1986.

Atherton M.A.A. (97), "Effective product design through enhancing robust engineering design and quality function deployment", PhD. Thesis,City University, School of Engineering, 1997.

Atherton M.A.A.., Private Communication, April, City University, 1998.

Atkinson W., Cunningham J., (91), "Proving properties of a safety critical system", Software Egineering Journal, March 1991.

Bennet S.(90) , "Design of software for real-time systems", pp 129-157, ed. S.Bennet and G.S.Virk, IEE 1990, Pereginus

Bently J.P. (99)., "Introduction to Reliability and Quality engineering", Addison Wesley, 2nd Edition, 1999.

Bleakley G.J., Edney, R.C & Chamberlain D.A., (97) "The Use Of DFD And ACP Diagrams For Mechatronics Systems Design", International Conference On Engineering Design, ICED 97 Tampere, August 19-21, 1997, Vol. 3.

Booch G., (91), "Object Orientated Design with Applications", Benjamin, Cummings Publishing Company, California, USA, 91.

Bowen J., Stavriduo V., (93),"Safety-critical systems, formal methods and standards", *Software Engineering Journal*, July 1993, pp 189-209.

Bracewell R.H., Chaplin R.V., Bradley D.A., (92),, "Schemebuilder and layout: computer based tools to aid the design of mechatronic systems", Mechatronics an integration of engineering design, IMechE, pp 1-6, 1992.

Bracewell R.H., Chaplin R.V., Langdon P.M., Li M., Oh V.K., Sharpe J.E.E, Yan X.T., (95). "Integrated platform for AI Support fo complex design, Parts 1&2" , Internal Reports, Lancaster University, Engineering Design Centre, EDC 12 & 13, 1995.

Bradley D.A, & Buur J., (93), "The representation of Mechatronic Systems", International Conference On Engineering Desig, 1993.

Bradley D.A. & Dawson D., (91) "Information Based Strategies in the Design of Mechatronic Systems", Design Studies Vol. 12, pp. 12-18, 1991.

Bradley D.A., (96), Private conversation with author, Technion University, Haifa, Israel, July 1996.

Bradley, D.A., Seward D.A., (92), "Lancaster University Computerised Intelligent Excavator programme", ImechE Conference mechatronics, Desinging intelligent machines, Dundee, 1992.

Bradley. D.A.,& Bracewell R.H., Chaplin R.V., (93a), "Engineering design and mechatronics: The schemebuilder project", Research in Engineering Design, No. 4, 1993, pp 241-248.

Bradshaw A., Bradley. D.A., (90), "Mechatronic Systems Engineering', Kluwer Acedamic Press, Vol. 1, pp 1-7, 1990.

Brehm E.W., Goettge R.T., Mc Caleb F.W., (95), "START/ES: An experts system tool for system peformance and reliability analysis", Performance Evaluation, Vol 22, No. 1, pp 43-48.

BSI (91a), "BS EN 292 part 1: Safety of Machinery- Basic concepts, general principles for design", British Standards Institute, 1991.

BSI (91b), "BS5760 : part 5, Reliability of systems equipment and failure: Guide to failure mode effect and criticality analysis (FMECA and FMEA)", HMSO, 1991

BSI (91c), "BS5760 : part 5, Reliability of systems equipment and failure: Guide to failure tree analysis" HMSO, 1991.

BSI (92), "BS EN 954-1: Safety of machinery-Safety Related parts of control systems-Part :general principles for design", Draft standards, British Standards Institute, December 1992.

Bucci G., Campanai M., Nesi P.(95), "Tools for specifiying real time systems", *Real Time Systems* , vol 7, pp 119-172, Kluwer Academic publishers1995.

Budgen D., (85), "Combining MASCOT with Modula-2 to aid the engineering of real-time systems", Software -Practice and Experience, Vol 15, No. 8, pp 767-793, August 1985.

Buur J., (90), "A theoretical approach to mechatronics design", PhD. Thesis, Technical university of Denmark,1990.

Buur J., (92), "Does Mechatronics Need A Special Attitude", Mechatronic Systems Engineering, Kluwer Academic Press, Vol 1. pp. 293-300, 1992.

Cameron J.R.,(89), "JSP and JSD: The Jackson approach to software development", IEEE Computer Society Press, 1989.

Carra G., De Paulis A., Tantussi G., (92), "SSR: a mobile robot on ferromagnetic surfaces", Automation in construction, Elsevier Science Publishers, 92, vol 1, pp 47-53.

Chakrabarti A., & Bligh, T.P., "An approach to functional synthesis of solutions in mechanical conceptual design. Part:1 Introduction and Knowledge representation", Research in Engineering Design, Springer Verlag, Vol. 6, 1994, pp. 127- 141.

Chamberlain D., Bleakley G.,(98) "Design methodology for mechatronic systems", for the DoE, City University, Construction robotics unit, 1998.

Chamberlain D.A., Bleakley G.J., Edney R.C, (95), "HSE research project 3262/R5925: remote handling of sources for radiographic inspection", Construction Robitics Unit, City University, November 95, published HSE

Chamberlain D.A., Bleakley G.J., Edney R.C, (96), "Robotic handling of Gamma ray sources in site radiography of steel storage tanks", 13th. International Symposium on Automation and Robotics in Construction, 1996, Elsevier.

CIA(77), "Hazard and Operability studies", Chemical Industries Association, 1977.

Clausing D. (89), "Quality Function Deployment: Applied system engineering", Quality and Productivity Research Conference, University of Waterloo, 1989

Clausing D. (92), "Total Quality Development- the competitive development of new products", Cambridge, 1992.

Clausing D.P., & Pugh S. (91), "Enhanced Quality Function Deployment", Proceedings of the Design and Productivity International Conference, Honolulu, Hawaii, February 6-8, 1991, pp15-25.

Clausing, D.P. , (94), "Total Quality Development- The development of competitive new products", Pub. ASME, new York, USA, 1994.

Coad P. & Yourdon E.,(91), "Object Orientated Analysis", Yourdon Press, New jersey, 1991.

Cohen, L., (95) , "Quality Function Deployment", Addison Wesley, 1995.

Collie, A.A, Billingsley, J., Von Puttkamer, E., (90), "Design and performance of the Portsmouth climbing robot", Mechatronic Systems Engineering, Kluwer Acedemic Publishers, Vol 1, pp 125-130, 1990.

Collins, (88), "The Collins Concise Dictionary", Ed, P.Hanks, William Collins and Sons, 1988.

Comincini S., (94), "design of a flexible manufacturing cell/system, optimizing for quality", Ph.D Thesis, City University, London, Department of Mechanical and Aeronautic Engineering, 1994.

Console L., Portinale L.,(92), "Model based diagnosis of system malfunction with petri-nets", Robotics and flexible manufacturing systems, pp 417-426, IMACS 92, Elsevier 1992.

Cop V., (92), "Methodology of mechatronics system decomposition", Mechatronics, Vol. 2, No. 3, pp 199-206, Pergammon, 1992.

Cross N. (89), "Engineering design methods", Wiley, 1989.

Cross N.,(93), "Science and Design methodology: a review", research in Egineering Design, Springer Verlag, 1993, pp 63-69

Da Silva J.C., Dawson D., (97), "the development of an expert system for hydraulic systems design focusing on concurrent engineering aspects", 11th International Conference on Engineering Design, Tampere, Finland, Vol 1, pp 271-276, 1997.

De Marco T., (79), "Structured anlysis and systems specification", Yourdon press, New York, 1979.

EC(93), Massimi P., Van Gheluwe J.P.,"Community legislation on machinery-Comments on directive 89/392/EEC and Directive 91/368/EEC", Comission of the European Communities, 1993.

Edney R.C. (93), "Introduction to Design Function Deployment", Technical Report, Engineering Design Centre, City Unversity , London, 1993.

Edney R.C., Atherton M.A., Lewington.N.P.,(99), "A square correlation matrix for design optimisation", Submitted to Journal of Design Optimisation, 1999.

Edney, R.C. Bleakley, G.J.,(95),"Feasibility design study for robotic handling of Gamma-ray sources for on site radiography", Mechatronics and Machine Vision in Practice 1996,

Electro-craft(77), "D.C. Motors, Speed controls, Servo systems", 3 ed. Electro-craft corp, Pergammon Press 1977.

Esterline A., Rosen. D., Otto K., Nelson L., Hessburg T., Riley D.R., Erdman A.G.,(88), "A methodology for capturing mechanical design expertise", Proceedings of the ASME computers in engineerig conference, American Society of mechanical Engineers, San Francisco, CA, 1988.

Evbuomwan N.F.O. , (95) "Design Function Deployment a concurrent engineering design system", Ph.D. Thesis, City University, London, 1995.

Evbuomwan N.F.O., & Anumba C.J., (95) "Towards a concurrent engineering model for design and build projects", Journal of the Institution of Structural Engineers, 1995.

Farsi M., Ratcliff K, Johnson P.J., Allen C.R., Karam K.Z., Pawson R., (94), "Robot control System for window cleaning", XI International Symposium on automation and robotics in construction, editor D.Chamberlain, 94.

Farsi M., Ratcliff K., Johnson P.J., Allen C.R., Karam K.Z., Pawson R.,(94), "Robot control system for window cleaning", 11th Int, Symp. on Automation and Robotics in Construction, ISARC 11.

Finger S. & Dixon J.R., (89a), "A review of research in mechanical engineering

design . Part 1: Descriptive, Prescriptive, and computer based models of design processes", Research in Engineering Design, Springer Verlag, Vol. 1, 1989, pp. 51- 67.

Fischer P.J., Daniel R.W., Siva K.V.,(90), "How does one specify and design an input device for teleoperation", Mechatronics: Designing Intelligent Machines, 12-13 Sept, Cambridge, Imeche, 1990, pp 115-121.

French M.J., (90), "The partition of functions in mechatronic design", Mechatronics: Designing Intelligent Machines, 12-13 Sept, Cambridge, Imeche, 1990, pp 209-211.

French M.J.,(71), "Engineering Design: the conceptual stage", Heinemann Educational Books, London, 1971.

Frolich, B, & Schlottmann,K(82) "Systematische Konstruktion elektronischer Gerate." VDI-Beridhte no. 460, 1982.

Gachet B., Hutinet T., Mignot J.,(92), "FIABEX modelling: Mars Rover's localisation unit", Safety and Reliability '92, Elsevier Appl. Sci, pp. 10-22, 1992.

Gawthorp P.J. , (92), "design of mechatronics using bond graphs", Mechatronics an integration of engineering design, IMechE, pp 15-21, 1992.

Gen M., Cheng R., (97), "Genetic Algorithms & Engineering Design", Wiley Series in engieering Design & Automation, 1997.

Golten J., Verwer A.,(91), "Control system design and simulation", McGraw-Hill, Berkshire Englzand, 1991.

Gomma H,(86), "Software development of real-time systems", Communications of the ACM, Vol. 29, No. 7, pp 657-668, July 1986.

Gomma H.,(84), "A software design method for real-time systems". Communications of the ACM, Vol. 27, No. 9, pp 938-949, Sept 1984

Goyku T, Takasu M., Fukuda, T., (96), "Development of wall painting robot", 13th

Int, Symp. on Automation and Robotics in Construction, ISARC 13, pp 949-956.

Graham-Jones P.J., Mellor B.G, (95), " Expert and Knowledge Based Systems in Failure Analysis", Engineering failure analysis, Elsevier Science Ltd. Vol. 2, No. 2, pp 137-149, 1995.

Hannah J., Hillier M.J., (88), "Applied Machanics", Longman Scientific & Technical, Harlow, England, 1988.

Harel D. , (88), "On visual formalisation", Communications of the ACM, Vol. 31, No. 5, pp 514-530, May 1988.

Harel D. et al, (87), "On the formal semantics of statecharts", Proc. 2nd IEEE Symposium on Logic in computer science, Ithaca, NY, USA, pp 54-64, June 1987.

Harel D. et el. , (90), "STATEMATE: A working model for the development of complex reactive systems", IEEE transactions on Software Engineering, Vol. 16, No 4, pp 403-411, 1990.

Hartley J. & Mortimer J., "Simultaneous Engineering- The Management Guide", 2nd ed., Industrial Newsletters Ltd., Dunstable.

Hatley D.J. & Pirbhai L.A., (87), "Strategies for real-time system specification", Dorset House Publishing, New York, 1987

Hauser J.R., & Clausing D. (88), "The house of quality", Harvard Business review, Vol. 66, No 3, Boston 1988.

Heilala J., Roppen T., Airila M., (92), "Mechatronic Design for Industrial Grippers", Mechatronics, Vol. 2, No. 3, pp 239-255, Pergammon, 1992.

Hein L., (90) "A product development approach to mechatronics", Mechatronics Systems Engineering, Kluwer Academic Press, Vol 1, pp 71-78, 1990.

Hoare C.A.R, (78), "Communicating Sequential Processes", *Communications of the ACM*, pp 666-677, August 1978.

Hoare C.A.R, (81), "A calculus of total correctness for communicating process", *Scientific. Computer Programing*, pp 49-72, vol 1, 1981.

Holt A., & Commoner F.,(70), "Events and Conditions", Record of the project MAC conference on concurrent systems and parallel computation,, ACM, June 1970, pp 1-52.

HSE(87) ,"Programmable electronic systems in safety related applications-

Part 1 an introductory guide", HMSO, 1987.

HSE(93) ,"Programmable electronic systems in safety related applications-

Part 2 General technical guidelines", HMSO, 1993.

Hubka V., (82), "WDK1 Principles of engineering design", Springer, 1982

Hull M.E.C., O' Donoghue P.G., (94), "Timed Petri-net approach to performance modelling with the MOON method", Software Engineering Journal, May 1994, pp 95-106.

Hull M.E.C., O' Donoghue P.G., Hagan B.J., (91), "MOON-Modular object orientated notation", *Software Engineering Journal*, January 1991, pp25-33, 1991.

Hunt, J.E. Pugh, D.R. Price, C.J.,(95),"Failure mode effects analysis: a practical application of functional modelling", Applied Artificial Intelligence, Vol: 9 Iss: 1 pp. 33-44, 1995.

Hykin D.H.W., & Lansing L.C., (75), "Design case histories: Report of a field study of design in the United Kingdom engineering industry", Proceedings of the Institute of Mechanical Engineers, Vol 189, No. 23, 1975.

Ingevaldsson L.,(79), "A practical method of program design", Chartwell-Bratt Ltd., England 1979.

Izquierdo-Rocha J.M., Sanchez-Perea M., (94), "Application of the integrated safety assessment methodology to the emergency procedures of a SGTR of a PWR", Reliability Engineering & System Safety, Vol: 45 Iss: 1-2 pp. 159-73, 1994.

Jackson K., & Simpson, H.R. (75), "MASCOT- A modular approach to software construction, operation and test", RRE Tech Note No.778. 1975.

Jackson K., (86), "MASCOT 3 and ADA", special issue on MASCOT 3, *Software Engineering Journal*, pp 121-135, May 1986.

Jackson M.A.,(75), "Principles of Program Design", Acedemic Press, London, 1975.

Jackson P., (90), "Introduction to Expert Systems", Addison-Wesley Publishing Company, 1990.

Jacobsen I., Christenson M., Jonsson J., Oevergaard G., (92), "Object Orientated Software Engineering: A Use Case Driven Appraoch", Harlow, Addison Wesley, 1992

Jebb A.,Edney R., Sivaloganathan S., Evbuomawan N. (92) "Design function deployment; a platform for cross disciplinary product development" , Mechatronics- The integration of engineering design. IMechE, pp23-27, 1992.

Jones J.C. & Thornley D.G., (ed.), Conference on design methods, Oxford, Pergammon Press, 1963,

Jones, J. (81), "Design Methods", Wiley, Chichester, 1981.

Kaassen K.B., Van Peppen J.C.L, (89), "System Reliability, Concepts and Applications", Hodder & Stoughton, 1989.

Kamhieh C.H., Cutts D.E., Purves R.B., (89), "Failure Modes and Effects Analysis Automation", 2nd Conference on artificial intelligence for space applications, Huntsville, Al, pp 169-176.

Karnopp D.C. , Rosenberg R.C., (75), "System dynamics a unified approach", John Wiley, 1975.

Kecskemethy A., (93), "MOBILE- An object oriented tool-set for the efficient modeling of mechatronic systems", Second Conference on mechatronics and robotics, Duisberg, Germany, September 27-29, 1993.

Kenjo T.,(93), " Electric motors and their controls", 1st ed. Oxford University Press, Oxford, England, 1993.

King B.,(87) , "Better designs in half the time, Implementing QFD in America",Gaol/QPC, 1987.

Klafter R.D.,Chmielewski T.A., Negin M., (89), "Robotic engineering, an integrated approach", Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

Knowles D., (90), "Mapping a MASCOT 3 design into OCCAM" , *Software Engineering Journal*, pp 207-213, July 1990.

Laplante P.,(93), "Real time systems design and analysis", IEEE Computer society press, 1994.

Lehtela M., (90), "Computer aided FMEA of electronic circuits", Microelectronics and reliability, pp 761-773, Vol. 30 No. 4, Pergammon, 1990.

Levenson N.G., Stolzy J.L., (87), "Safety analysis using petri-nets", *IEEE Transactions on software engineering*, Volume SE 13, No. 3, pp 386-397, March 1987.

Luvic I, (98), "Failure of Level crossing (automatic half barrier) to protect the public from passing trains, risk assessment study", Msc Thesis, City University, Department of Mechanical Engineering & Aeronautics, 1998.

Marples D.L., (61), "The decisions of engineering design". IRE transactions on engineering management, Vol. EM 8, 1961, pp.. 55-70.

Martin J. & Odell J., (91), "Object Oriented Analysis & Design", Prentice Hall International, Englewood Cliffs, New Jersey, 1991.

MASCOT , (80), "The official Handbook of MASCOT", MASCOT suppliers association, 1980.

MASCOT , (87), "The official Handbook of MASCOT", Computing division, RSRE Malvern 1987.

MASCOT 3 (86), "Special issue on MASCOT 3", *Software Engineering Journal*, pp 93-120, May 1986.

Mc Crae, A.M., Bleakley, G.J. (99), "Design for Reliability of a Construction Robot using Genetic Algorithms", International Symposium for Automation And Robotics in Construction, 1999, Madrid, Spain, Spetember 22-25, 1999.

Mealy G.H., (55), "A method for synthesizing sequential circuits", BST Journal, 34: 1045-1079, 1955.

Miller G.A., (56), "The magical number seven, plus or minus two: some limits on our capacity for processing information", Psychological Review, No. 63, 1956, pp. 81-97.

Miller T.J.E, (89), "Brushless permanent-magnet and reluctance motor drives", Monographs in electrical and electronic engineering, Oxford University Press, Oxford, England, 1989.

Millner R., (80), " A calculus for communicating systems", Lecture Notes in Computer Science 92, Spinger Verlag, New York, 1980.

Miyajima T, Terauchi S., Arai K., Miyamoto T., (96), "Development of a gondola type external wall painting robot", 13th Int, Symp. on Automation and Robotics in Construction, ISARC 13, pp 957-962.

Moore E.F., (56), "Gedanken Experiments on sequential machines", Automata Studies, Annals of mathematical studies, pp 129-153, 1956

Mort N., (90), "Realtime software using MASCOT", Computer control of real-time processes, pp 159-172, ed. S.Bennet and G.S.Virk, IEE 1990, Pereginus

Moser E., Melliar-Smith P.M., (90), "Formal verification of safety critical systems" , *Software- Practice and Experience*, pp 799-821, vol 20 (8), August 1990.

Murdoch J., (92), "Logic modelling of dependable Systems", Safety of Computer Control Systems 1992, (SAFECOMP 92), Computer Systems in Safety critical Applications, Proc. IFAC syposium, Pergammon, pp 111-116.

Ogata K., (70), " Modern Control Engineering", Prentice-Hall International Inc. , Englewood Cliffs, New Jersey, 1970.

Ostroff J.S. (89), "Temporal Logic for real time systems", Advanced Software Development Series, Research Studies Press Limited, John Wiley, England, 1989.

Pacific Scientific (93), "Motion Control application note 101", Pacific Scientific, Motor Control Division, Rockford, Il, USA, 1993.

Pahl G., Beitz W., (84), "Engineering design", London design council, 1984

Palumbo, D.,(94), "Automating failure modes and effects analysis", Proceedings of the Annual Reliability and Maintainability Symposium.,IEEE, pp. 304-9, 1994.

Pandey A., (92), "QFD: A study of Implementation and Enhancements",masters Thesis, MIT, Cambridge, 1992.

Paynter H.M., (61), "Analysis and design of engineering systems", MIT press, Cambridge, Mass, 1961.

Peterson J.L.,(81), "Petri-net theory and modelling of systems", prentice-Hall, 1981.

Petri C., (62) "Communication with automata", Phd dissertation, Univeristy of Bonn, West Germany, 1962

Phadke M.S., (89), "Quality Engineering using Robust Engineering Design", Prentice Hall, Englewood Cliffs, USA, 1989.

Philips C.S.E, (67), "Networks for realtime programming", Computer Journal, Vol 10, No.1, pp46-52, 1967

Potter B., Sinclair J, Till D., (91), "An introduction to formal specification and Z", Prentice-Hall International Series in Computer Science, 1991.

Price C.J, Snooke N., Pugh D.R., Hunt J.E., Wilson M.S., (97), Combining functional and structural reasoning for safety analysis of electrical designs", Knowledge Engineering Review, Vol. 12, iss, 3, pp 271-287, 1997.

Pugh D.R., Snooke, N, (96), "Dynamic Analysis of qualitative circuits for failure mode and effects analysis", Proceedings of the Annual reliability and maintainability symposium, IEEE, pp 37-42, 1996

Pugh S.,(81), "Total design-A method that works", proceedings of the International Conference on Engineering design, Rome, March 9-13, WKD 5, paper M3/16, 1981, pp 497-506.

Pugh S.,(91) "Total design", Addison Wesley Publishers Ltd., 1991.

Quatrani .T, (98), "Visual modeling with rational rose and UML", Addison Wesley Object Technology Series, Addison Wesley, 1998.

Robinson P.J., (92), "HOOD, Hierarchical Object-Orientated Design", Prentice Hall International, 1992.

Roozenburg N.F.M, & Eekels J., (95), "Product Design, Fundamentals and Methods", John-Wiley & Sons Ltd., Chichester, England, 1995.

Rosenburg, (87), "Exploiting bond graph causality in physical systems modeling", Transactions of the ASME, Journal of Dynamic Systems, measurement and Control, Vol 109(4), pp378-383, 1987.

Ross D.T., Schoman K.E., (77), "Structured Analysis for Requirements Definition", IEEE Transactions on Software Engineering, Vol.3 No. 1, pp 6-15, June 1977

Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorenson W.,(91), " Object Oriented Modelling and Design", Prentice Hall International, Englewood Cliffs, New Jersey, 1991

Russamanno J.D., Bonnell R.D. , Bowles J.B., (92), "Computer Aided FMEA: Toward an artificial intelligence approach", Int. Symp. On Artificial Intelligence, The Artificial Intelligence Transfer Conference, pp 103-112, Dec. 1992.

Salminem V., Verho A.., (92), "Systematic and Innovative design of a mechatronic product",*Mechatronics*, Vol. 2, No. 3, pp 257-275, 1992

Salminem V., Tanskanen K., Verho A., (90), "The Finnish Approach- experiences in adapting to educational research and industry purposes", Mechatronics: Designing Intelligent Machines, 12-13 Sept, Cambridge, Imeche, 1990, pp137-141.

Salome T., (95), "What every engineer should know about concurrent engineering", Marcel Dekker Inc., New York, 1995.

Savakoor D.S., Bowles J.B., Bonnell R.D.,(93),"Combining sneak circuit analysis and failure modes and effects analysis", Proceedings, Annual Reliability and Maintainability Symposium, IEEE, New York, pp. 199-205, 1993

Selic B., Gullekson G., McGee J. & Engelberg I., (92), "ROOM, an object oriented methodology for developing real-time systems", Proceedings of 5th Int. Conference on computer aided software engineering. CASE 92, Montreal, Quebec, Canada, July 1992.

Seward D.W, & Quayle S.D., (96), "System Architectures and Safety for Mobile Construction Robots", 13th. International Symposium on Automation and Robotics in Construction, 1996, Elsevier, pp223-231.

Seward D.W., Bradley D.A., Margrave F.W., (94), "Hazard anzalysis techniques for mobile construction robots", Automation and Construction in Robotics XI, D.A. Chamberlain Ed., Elsevier Science, 1994.

Shaw P., Dale B.G., (90), "FMEA in the UK motor industry: a state of the art study", QREI, Vol 6, pp, 179-188.

Sivaloganathan S., Evbuomwan N.F.O, Jebb A.,. Wynn H.P.,(95), "Design function deployment- a design systems for the future", *Design Studies*, Vol 16, pp 447-470, 1995.

Song X., & Osterwell L.J., (94), "Experience with an approach to comparing software design methodologies", IEEE Transactions on software Engineering, Vol. 20, No. 5, May 1994.

Sontow K. , & Clausing D.P.,(93) "Integration of Quality of Function Deployment with Further Methods of Quality Planning" , Working paper No. LMP-93-005, Laboratory for Manufacturing and Productivity, MIT, Cambridge, Massachusetts, April, 1993.

Sontow K., & Clausing P., (93), "Integration of Quality Function Deployment with Further methods of quality planning", Working paper LMP-93-005, Laboratory for Manufacturing and Productivity, Massachussetts Institute of Technology, April 3rd 1993.

Suh H.,(88), "The Principles of Design", Oxford University Press, 1988.

Syan C.S., & Menon U., eds.(94), "Concurrent Engineering, Concepts, implementation, and practice", Chapman Hall, 1994.

Taguchi G., Clausing D., (90), " Robust Quality", Harvard Business Review, 1990, Vol 1, pp 65-75.

Tebay R., Atherton J. & Wearne S.H., (84),"Mechanical Engineering design decisions:Instances of practice compared with theory", Proceedings of the Institute of Mechanical Engineers, Vol 198, No. 6, 1984.

Ullman D.G. & Dietterich T.A.,(86),"Mechanical design methodology", Computers in Engineering, American Society of Mechanical Engineers, 1986, pp. 173-180.

Ullman D.G. , Stauffer L.A. & Dietterich T.A.,(87),"Preliminary results of an experimental study of the mechanical design process", Proceedings from the NSF Workshop on the design process, Waldron M.B. ed. Ohio State University, Oakland, CA. Feb 8-10, 1987, pp. 145-188.

Ullman D.G., (92), "The mechanical design process", McGraw-Hill inc. USA 1992.

Ulrich K.T. & Seering W.P. ,(89), "Synthesis of Schematic descriptions in mechanical design", Research in Engineering Design, 1:3-18, Springer Verlag, 1989.

Unimatic Plc, (93), "Unimatic stepping motor applications guide", Unmatic, 1993.

VDI (77),"VDI-Richtlinien, VDI 2222, " Knostruktionsmethodik, Konzipieren technisher produkte, Blatt 1". Beuth-Verlag Berlin-Koln, 1977.

VDI (86),"VDI-Richtlinien, VDI 2221, Systematic approach to the design of technical systems and products". VDI-Verlag GmbH, Dussuldrorf, 1987.

Viitanen P., Yli-Pietila T, Yli-Paunu P, Suoranta R. (92), "Modelling and simulation of mechatronical devices", Mechatronics, Vol, 2, No 3, pp 231-238, 1992.

Wang J. & Ruxton T, (97), "A review of safety analysis methods applied to the design process", Journal of Engineering Design, Vol. 8, No. 2, pp 131-152, 1997.

Ward P.T., & Mellor S.J., (85), "Structured Development for Real Time Systems", Prentice-Hall, 1985.

Ward P.T., (86), "The Transformation Schema: an extension of the data flow diagram to represent control and timing", IEEE Transactions on Software Engineering, Vol. 12, No. 2, pp 198-210, February 1986.

Wing R.D.,(89), "Robotics in construction-a state of the art", Proc, Instn of Civil Engineers, Part 1, 1989, vol 86, October, pp 937-952.

Wood C., (90) "FADES: a tool for automated fault analysis of complex systems", Artificial intelligence in nuclear power plants, IAEA/IWG, NPPCI Specialists meeting CUTT symposium (109-110), pp 77-89, Vol. 2, 1990

Yan X.T. & Mc Callum K.J., (97), "An integrated novel approach to enhancing interdisciplinary product design", 11th International Conference on Engineering Design, Tampere, Finland, Vol 1, pp 267-270, 1997.

Yourdon E. & Constantine L.L., (79)"Structured design: fundamentals of a discipline of computer program and systems design", Prentice-Hall, New Jersey, USA, 1979.

Yourdon E. ,(94), " Object Orientated Systems Design, An integrated Approach", Prentice-Hall, New Jersey, 1994.

Yourdon E., (89), "Modern Structured Analysis", Prentice-Hall International Editions, 1989.

Zhang L.,(96), "Sensory safety systems for robotic construction plant", Draft M.Phil. Thesis, Lancaster University, 1996.

Zwicky F., (48)"The morphological method of analysis and construction", Courant Anniv. Wiley-Intersc., New York, 1948.

Appendix A : Glossary of Terms

A number of technical terms are used in this paper. The following glossary explains these and gives their abbreviation where appropriate.

Single Point Failure (SPF): Failure of one component causes complete system failure, this idea is extended to include Double Point Failures (DPFs), Triple Point Failures (TPFs) and Quadruple Point Failures (QPFs), where multiple component failure leads to system failure.

Activity Channel Pool diagrams (ACP): Hierarchical model of a dynamic system based upon design functions and the correlation chain.

Concurrent Design: Design methodology that simultaneously satisfies multiple design constraints e.g. design for manufacture and maintenance.

Consequence Lead Analysis of Safety Hazards (CLASH): Risk analysis tool that examines the consequence of an event and works back to its cause.

Correlation Chain: Cause and effect chain created from the dynamic relationships between the various functional parts of the system..

Customer Requirements: Depending on the product, these can be the collective requirements of the product owner, operator and manager and those of external agencies demanding compliance with safety standards and other statutory instruments.

Design for X (DFX): General term used to describe a design methodology with a specific aim e.g's design for manufacture.

Design Function Deployment (DFD): Concurrent Design method that is an extension of the *QFD* method. It gives improved support for conceptual design by including manufacturing constraints and also by breaking down the solution in a *Neutral Solution Space*

Design Function: The design for any product can be broken down into a series of modules, these modules are called design functions. For example a torch would have design functions relating to the power source, on/off switch, light source etc. each of these design functions should be described in a neutral solution manner. See neutral solution space.

Event Tree Analysis (ETA): Graphical representation of events leading to failure.

Failure combination: A set of failure modes which lead to system failure.

Failure criteria: Specification against which system failure is measured

Failure mode: Particular mode of component behaviour that results in component failure.

Failure Modes and Effects Analysis (FMEA): Analysis of the effects of component or system failure in relation to the mode of failure.

Failure space: Abstract area in which possible component failure combinations that lead to system failure exist.

Fault Tree Analysis (FTA): Graphical description of the failure combinations that lead to system failures.

First Order Cut-set: Single component failure that leads to failure of the entire system.

Formal Methods: Complex mathematical language used to describe and verify computer software

Function Means Tree: Graphical representation of hierarchical design functions in terms of *Transformation Functions* and *Purpose Functions.*

HAZard and OPerability study (HAZOP): Risk analysis tool for 'what if' analysis of events.

Law of vertical casuality: Law stating that secondary level functions cannot be decided before first level functions.

Mechatronic: A functional and spatial integration of mechanical and electrical components with information technology (Buur 90a).

Metamethodics: Mechatronic systems design methodology derived from *QFD, VDI-2221* and *SASD.*

Minimum Cut-set: As applied in *FTA*, a minimum cut-set is the shortest route to system failure e.g. the worst case of *First Order Cut-set.*

Morphological Analysis: Tabulated representation of *Purpose Functions* against *Transformation Functions* that allows combinations of solutions to be generated

Multi-state system: A system which can have more than two discrete states, for instance a car gearbox.

Neutral Solution Space: Description of a problem in terms that does not restrict the solution.

Object Oriented Programming: Object Orientated Programming, powerful software design philosophy, which breaks the program down into modules called objects.

Petri-Nets: Graphical tool which can be used to describe how events affect system sets. Associated with the theory of bags which is used to analyse petri-net models

Process function: Description of the means by which a transformation function is realised

Programmable and Electronic System (PES): Machinery or product comprising mechanical and electronic elements controlled by computer or microprocessor.

Purpose Function: Description of a *Transformation Function* in terms of how it may be realised e.g. a rack and pinion system is a candidate Purpose Function in the transformation of rotary to linear motion.

Quality Function Deployment (QFD): QFD:- Design methodology based upon the analysis of customer requirements. Used to describe the direction of design improvements for a product.

Reduced Fault Tree (RFT): Minimum fault tree derived from the full fault tree.

Redundant Voting System: Technique used in safety critical systems, whereby three different, independent sensors are used to monitor the same variable. If the result of one of the sensors is different from the other two the system would normally be configured to shut down.

Right-side failure: Failure of a component that results in the system being left in a safe state.

Risk Priority Number (RPN): Calculated by *Failure Modes Effects Analysis* of the system. It identifies components or systems that are most likely to fail or remain undetected when failure occurs.

Second Order Cut-set: Simultaneous failure of two components or sub-systems that leads to fail of the entire system.

Solution Space: Abstract area where solutions to a problem are defined.

State Space Diagram (SPD): Graphical modelling tool based upon the states of the

functions within a system.

State Transition Diagrams (STD's): Graphical modelling tool using the states of the functions within a system.

Systems Analysis - Systems Design (SASD): Graphical tool used to represent software systems [Ward and Mellor]

Theory of domain: Abstract space that defines the position of a design in terms of completion of the design and manufacturing processes.

Time-out: Technique used in safety critical systems whereby if an operation or predicted change in state of a system does not occur within a set time limit, the system is shut down.

Transformation Function: Description of a function that does not prescribe the solution e.g. the transformation of rotary motion to linear motion. It is used with *Function Means Trees.*

Variant Solution: One of a number of solutions to a specific design function, Variant evaluation is the process of selecting the variant solution.

VDI-2221: German Standard for a systematic approach to the design of technical systems and products (VDI 86)

VDI-2222: German standard for the design of technical products (VDI 77)

VISSIM: Graphical programming software for the modelling of dynamic systems

Weighted Objectives Method: Method for assessing the importance of each customer requirement and design function in terms of the overall design [ref(cross)]

Wrong-side failure: Failure of a component that results in the system being left in a unsafe state.

APPENDIX B: RESULT FILES GENERATED BY VISSIM MODELS FOR COMPUTERISED COMPONENT SELECTION AND FAILURE ANALYSIS

This appendix contains the output of the models for component selection and failure analysis. The first section displays the successful combination files for the X and Y models for component selection. The next section shows the output from the Failure logs of the X and Y models for component selection. The successful combinations at each stage are highlighted.

The Third section shows condensed versions of the output files from the VISSIM models for failure analysis. These files provide the data from which the Reduced Fault Trees of chapter 9 are produced. It is divided into four further sections which cover each generation of failure model.

All the information contained in this appendix can be found on the supplementary disks A and B. Appendix C details the location of the various files on the disks.

SUCCESSFUL COMBINATION FILES FOR COMPONENT SELECTION.

This section contains the successful combination output files from the VISSIM models for, the X and Y axis. The first and fourth column gives the new combination ID, the second the Motor ID, the third the Gearbox ID and fourth the Pulley ID, whilst the final column indicates the original combination ID.

Successful combinations for X axis component selection.

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	0	0	0	1	1
2	0	0	1	2	2
3	0	0	2	3	3
4	0	0	3	4	4
5	0	1	0	5	7
6	0	1	1	6	8
7	0	1	2	7	9
8	0	1	3	8	10
9	0	1	4	9	11
10	0	1	5	10	12
11	0	2	1	11	14
12	0	2	2	12	15
13	0	2	3	13	16
14	0	2	4	14	17
15	0	2	5	15	18
16	0	3	3	16	22
17	0	3	4	17	23
18	0	3	5	18	24
19	0	4	5	19	30
20	2	0	0	20	73
21	2	0	1	21	74
22	2	0	2	22	75
23	2	0	3	23	76
24	2	1	0	24	79
25	2	1	1	25	80
26	2	1	2	26	81
27	2	1	3	27	82
28	2	1	4	28	83
29	2	2	1	29	86
30	2	2	2	30	87
31	2	2	3	31	88
32	2	2	4	32	89
33	2	2	5	33	90
34	2	3	3	34	94
35	2	3	4	35	95
36	2	3	5	36	96
37	2	4	5	37	102

Table Appendix 0-10 Contents of STAGEX1.MAP

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	2	0	0	1	20
2	2	0	1	2	21
3	2	0	2	3	22
4	2	0	3	4	23
5	2	1	0	5	24
6	2	1	1	6	25
7	2	1	2	7	26
8	2	1	3	8	27
9	2	1	4	9	28
10	2	2	1	10	29
11	2	2	2	11	30
12	2	2	3	12	31
13	2	2	4	13	32
14	2	2	5	14	33
15	2	3	3	15	34
16	2	3	4	16	35
17	2	3	5	17	36
18	2	4	5	18	37

Table Appendix B- 11Contents of STAGEX2.MAP

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	2	0	3	1	23
2	2	1	4	2	28

Table Appendix B- 12 Contents of STAGEX3.MAP

Successful combinations for Y axis component selection.

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	0	0	0	1	1
2	0	0	1	2	2
3	0	0	2	3	3
4	0	0	3	4	4
5	0	1	0	5	7
6	0	1	1	6	8
7	0	1	2	7	9
8	0	1	3	8	10
9	0	1	4	9	11
10	0	2	1	10	14
11	0	2	2	11	15
12	0	2	3	12	16
13	0	2	4	13	17
14	0	2	5	14	18
15	0	3	3	15	22
16	0	3	4	16	23
17	0	3	5	17	24
18	0	4	5	18	30
19	2	0	0	19	73
20	2	0	1	20	74
21	2	0	2	21	75
22	2	0	3	22	76
23	2	1	0	23	79
24	2	1	1	24	80
25	2	1	2	25	81
26	2	1	3	26	82
27	2	1	4	27	83
28	2	2	1	28	86
29	2	2	2	29	87
30	2	2	3	30	88
31	2	2	4	31	89
32	2	2	5	32	90
33	2	3	3	33	94
34	2	3	4	34	95
35	2	3	5	35	96
36	2	4	5	36	102

Table Appendix B- 13 Contents of STAGEY1.MAP

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	0	1	0	1	5
2	0	2	1	2	10
3	0	2	2	3	11
4	0	3	3	4	15
5	0	3	4	5	16
6	0	4	5	6	18
7	2	0	0	7	19
8	2	0	1	8	20
9	2	0	2	9	21
10	2	0	3	10	22
11	2	1	0	11	23
12	2	1	1	12	24
13	2	1	2	13	25
14	2	1	3	14	26
15	2	1	4	15	27
16	2	2	1	16	28
17	2	2	2	17	29
18	2	2	3	18	30
19	2	2	4	19	31
20	2	2	5	20	32
21	2	3	3	21	33
22	2	3	4	22	34
23	2	3	5	23	35
24	2	4	5	24	36
25	2	5	5	25	37

Table Appendix B- 14 Contents of STAGEY2.MAP

ID No	Motor ID	Gearbox ID	Pulley ID	ID No	Original ID
1	2	0	3	1	22
2	2	1	4	2	27

Table Appendix B- 15 Contents of STAGEY3.MAP

Failed component combination details for 1st , 2nd and 3rd stage models for the X and Y axis.

The data contained in these files is quite extensive covering all the specification variables and flags,1 indicates a success and 0 an unsuccessful test. The contents of each of the columns is explained in Tables Appendix B 7 and B 8. Successful combinations passed to the next stage of analysis are highlighted within their respective tables.

Column	Function	Variable	Units
1	Data ID	Number	
2	Nominal Velocity of motor	Veln	Rad/s
3	Peak Linear Velocity	VLPeak	M/s
4	Vlpeak between 0.5-1 M/s , flag	Flag	
5	Angular Velocity	Velocity	Rad/s
6	Angular Velocity < 314 + 3%, flag	Flag	
7	Peak Torque developed by motor	Torque	NM
8	Tpeak < nominal motor Torque (Tnom), Flag	Flag	

Table Appendix B- 16 Column designation for Stage 1 result files

Column	Function	Variable	Units
1	Combination ID	Number	
2	General error flag	Flag	
3	Peak Torque produced by motor	Tpeak	NM
4	Tpeak > motor maximum torque (Tmax) flag	Flag	
5	Rms torque developed by motor	Trms	NM
6	Trms > motor nominal torque (Tnom) flag	Flag	
7	System inertia seen by motor	Inertia	kgM^2
8	Inertia < 3.5 * motor inertia (Jm) flag	Flag	
9	Rms Power developed by motor	Prms	Watts
10	Prms < motor nominal power (Pnom) flag	Flag	
11	Linear velocity of load	VLPeak	M/s
12	Angular velocity of gearbox < 314 rad/sec flag	Vpeak	Rad/s
13	Actdistance moved is +/- 5% of expected distance flag	Flag	
14	Distance moved by load	Actdist	M
15	Angular velocity of gearbox	Vpeak	Rad/s
16	Angular acceleration of motor	Apeak	Rad/s^2

Table Appendix B- 17 Column designation for stage 2 and 3 result files

Logged failed combinations for X axis component selection

Data ID	Veln	Vlpeak	Flag 1	Velocity	Flag 2	TPeak	Flag 3
1	314	0.7407	1	256.8	1	4.299	1
2	314	0.7875	1	239.9	1	4.343	1
3	314	0.8715	1	211.1	1	4.426	1
4	314	0.9481	1	190.6	1	4.507	1
5	314	1.02	0	173.6	1	4.583	1
6	314	1.084	0	159.8	1	4.657	1
7	314	0.6563	1	293.6	1	4.222	1
8	314	0.6995	1	273	1	4.258	1
9	314	0.7782	1	242.5	1	4.327	1
10	314	0.8516	1	218.1	1	4.398	1
11	314	0.9221	1	199	1	4.469	1
12	314	0.9872	1	182.9	1	4.54	1
13	314	0.5794	1	336	0	4.153	1
14	314	0.6185	1	313.8	1	4.184	1
15	314	0.6906	1	277.4	1	4.244	1
16	314	0.759	1	251.3	1	4.303	1
17	314	0.8259	1	225.9	1	4.363	1
18	314	0.8892	1	207.9	1	4.429	1
19	314	0.4678	0	394	0	4.061	1
20	314	0.5003	1	392.9	0	4.084	1
21	314	0.5606	1	348.5	0	4.129	1
22	314	0.6189	1	313.1	1	4.176	1
23	314	0.677	1	285	1	4.223	1
24	314	0.7334	1	261.3	1	4.271	1
25	314	0.3916	0	395.7	0	4.006	1
26	314	0.4191	0	395.2	0	4.023	1
27	314	0.4704	0	394.1	0	4.058	1
28	314	0.5204	1	376.9	0	4.094	1
29	314	0.5706	1	342.4	0	4.131	1
30	314	0.62	1	313.8	1	4.17	1
31	314	0.3762	0	396	0	3.996	1
32	314	0.4027	0	395.6	0	4.012	1
33	314	0.4521	0	394.5	0	4.044	1
34	314	0.5004	1	392.9	0	4.078	1
35	314	0.5488	1	356.4	0	4.113	1
36	314	0.5966	1	326.2	0	4.151	1
37	628	1.304	0	257	1	3.315	1
38	628	1.341	0	239.9	1	3.315	1
39	628	1.395	0	212.1	1	3.315	1
40	628	1.427	0	189.7	1	3.315	1
41	628	1.449	0	175	1	3.315	1
42	628	1.454	0	158.4	1	3.315	1
43	628	1.231	0	292.7	1	3.315	1
44	628	1.275	0	271.8	1	3.315	1
45	628	1.343	0	243.8	1	3.315	1
46	628	1.392	0	217.9	1	3.315	1
47	628	1.429	0	197.2	1	3.315	1
48	628	1.45	0	180.5	1	3.315	1
49	628	1.146	0	336.1	0	3.315	1
50	628	1.195	0	311	1	3.315	1
51	628	1.273	0	275.5	1	3.315	1
52	628	1.334	0	249.4	1	3.315	1
53	628	1.385	0	228.5	1	3.315	1

Table Appendix B- 18 Contents of FAIL1X.DAT

54	628	1.42	0	208.4	1	3.315	1
55	628	0.9919	1	416.9	0	3.315	1
56	628	1.043	0	391.5	0	3.315	1
57	628	1.129	0	347.3	0	3.315	1
58	628	1.203	0	313.2	1	3.315	1
59	628	1.269	0	284.7	1	3.315	1
60	628	1.323	0	260.9	1	3.315	1
61	628	0.865	1	501.5	0	3.315	1
62	628	0.914	1	466.9	0	3.315	1
63	628	1	0	418.2	0	3.315	1
64	628	1.077	0	375.4	0	3.315	1
65	628	1.148	0	342.6	0	3.315	1
66	628	1.21	0	310.6	1	3.315	1
67	628	0.8373	1	524.4	0	3.315	1
68	628	0.8857	1	489.8	0	3.315	1
69	628	0.971	1	431.9	0	3.315	1
70	628	1.048	0	389.2	0	3.315	1
71	628	1.119	0	356.4	0	3.315	1
72	628	1.182	0	324.4	0	3.315	1
73	314	0.718	1	255.8	1	8.127	1
74	314	0.7694	1	237.3	1	8.279	1
75	314	0.8661	1	213.2	1	8.57	1
76	314	0.9611	1	189.1	1	8.865	1
77	314	1.057	0	172.4	1	9.169	1
78	314	1.153	0	156.1	1	9.474	1
79	314	0.6283	1	294.2	1	7.839	1
80	314	0.6732	1	273.5	1	7.968	1
81	314	0.7579	1	239.5	1	8.219	1
82	314	0.8412	1	216.8	1	8.472	1
83	314	0.9257	1	193.8	1	8.722	1
84	314	1.01	0	178	1	8.995	1
85	314	0.55	1	336.3	0	7.599	1
86	314	0.5892	1	312.3	1	7.706	1
87	314	0.6632	1	275.3	1	7.917	1
88	314	0.7361	1	249.3	1	8.134	1
89	314	0.8101	1	228	1	8.352	1
90	314	0.8841	1	206	1	8.578	1
91	314	0.4405	0	370.3	0	7.291	1
92	314	0.4719	0	370.3	0	7.367	1
93	314	0.5309	1	348	0	7.519	1
94	314	0.589	1	313.2	1	7.678	1
95	314	0.6482	1	283.7	1	7.842	1
96	314	0.7074	1	259.7	1	8.017	1
97	314	0.3675	0	370.3	0	7.112	1
98	314	0.3936	0	370.3	0	7.167	1
99	314	0.4429	0	370.3	0	7.28	1
100	314	0.4913	0	370.3	0	7.399	1
101	314	0.5405	1	341	0	7.524	1
102	314	0.5897	1	313.7	1	7.66	1
103	314	0.3528	0	370.3	0	7.079	1
104	314	0.378	0	370.3	0	7.13	1
105	314	0.4252	0	370.3	0	7.235	1
106	314	0.4717	0	370.3	0	7.347	1
107	314	0.519	1	356.1	0	7.464	1
108	314	0.5662	1	323.6	0	7.592	1
109	628	1.694	0	254.6	1	7.38	1

Table Appendix B- 9 Contents of FAIL1X.DAT (Continued)

110	628	1.826	0	238.6	1	7.38	1
111	628	2.074	0	209.2	1	7.38	1
112	628	2.322	0	188.5	1	7.38	1
113	628	2.579	0	169.5	1	7.38	1
114	628	2.835	0	157	1	7.38	1
115	628	1.469	0	287.7	1	7.38	1
116	628	1.582	0	271.8	1	7.38	1
117	628	1.795	0	242	1	7.38	1
118	628	2.008	0	214.4	1	7.38	1
119	628	2.227	0	195.2	1	7.38	1
120	628	2.451	0	182.7	1	7.38	1
121	628	1.272	0	336.5	0	7.38	1
122	628	1.37	0	312.5	1	7.38	1
123	628	1.554	0	275	1	7.38	1
124	628	1.738	0	247	1	7.38	1
125	628	1.926	0	227.6	1	7.38	1
126	628	2.119	0	208.8	1	7.38	1
127	628	1.005	0	412.8	0	7.38	1
128	628	1.08	0	388.8	0	7.38	1
129	628	1.223	0	342.7	0	7.38	1
130	628	1.367	0	306.5	1	7.38	1
131	628	1.515	0	279.8	1	7.38	1
132	628	1.664	0	261.3	1	7.38	1
133	628	0.8318	1	502.2	0	7.38	1
134	628	0.8933	1	469.1	0	7.38	1
135	628	1.01	0	413.9	0	7.38	1
136	628	1.127	0	369	0	7.38	1
137	628	1.246	0	342.2	0	7.38	1
138	628	1.367	0	307.8	1	7.38	1
139	628	0.797	1	524.1	0	7.38	1
140	628	0.8555	1	491.1	0	7.38	1
141	628	0.9679	1	435.7	0	7.38	1
142	628	1.08	0	390.5	0	7.38	1
143	628	1.194	0	355.2	0	7.38	1
144	628	1.309	0	320.7	1	7.38	1
145	314	0.6959	1	256.3	1	11.18	0
146	314	0.7443	1	237	1	11.37	0
147	314	0.8348	1	213	1	11.75	0
148	314	0.9233	1	188.8	1	12.13	0
149	314	1.013	0	173.2	1	12.51	0
150	314	1.103	0	157.6	1	12.93	0
151	314	0.6112	1	291.1	1	10.82	0
152	314	0.6538	1	274.8	1	10.98	0
153	314	0.7336	1	238.6	1	11.29	0
154	314	0.8117	1	215.7	1	11.62	0
155	314	0.8906	1	192.6	1	11.95	0
156	314	0.9692	1	177.6	1	12.29	0
157	314	0.5364	1	334.8	0	10.53	0
158	314	0.574	1	311	1	10.66	0
159	314	0.6444	1	276.1	1	10.92	0
160	314	0.7133	1	248.6	1	11.19	0
161	314	0.7829	1	227	1	11.46	0
162	314	0.8521	1	204.7	1	11.76	0
163	314	0.4307	0	352.5	0	10.16	0
164	314	0.4611	0	352.5	0	10.25	0
165	314	0.5181	1	347.7	0	10.43	0

Table Appendix B- 9 Contents of FAIL1X.DAT(Continued)

166	314	0.574	1	311.7	1	10.62	0
167	314	0.6304	1	284.6	1	10.82	0
168	314	0.6865	1	259.3	1	11.04	0
169	314	0.3597	0	352.5	0	9.952	0
170	314	0.3852	0	352.5	0	10.02	0
171	314	0.433	0	352.5	0	10.15	0
172	314	0.48	0	352.5	0	10.29	0
173	314	0.5274	1	342.5	0	10.44	0
174	314	0.5747	1	312.1	1	10.6	0
175	314	0.3455	0	352.5	0	9.914	0
176	314	0.37	0	352.5	0	9.974	0
177	314	0.4159	0	352.5	0	10.1	0
178	314	0.4611	0	352.5	0	10.23	0
179	314	0.5067	1	356.9	0	10.37	0
180	314	0.5522	1	323.1	1	10.52	0
181	628	1.979	0	257.2	1	7.02	1
182	628	2.06	0	239.4	1	7.02	1
183	628	2.189	0	211.2	1	7.02	1
184	628	2.288	0	189.6	1	7.02	1
185	628	2.368	0	174	1	7.02	1
186	628	2.422	0	159	1	7.02	1
187	628	1.778	0	293.9	1	7.02	1
188	628	1.909	0	270.5	1	7.02	1
189	628	2.053	0	242.2	1	7.02	1
190	628	2.168	0	215.4	1	7.02	1
191	628	2.268	0	199.9	1	7.02	1
192	628	2.343	0	180.2	1	7.02	1
193	628	1.551	0	336.3	0	7.02	1
194	628	1.666	0	312.6	1	7.02	1
195	628	1.882	0	278.6	1	7.02	1
196	628	2.026	0	251.5	1	7.02	1
197	628	2.139	0	225.7	1	7.02	1
198	628	2.231	0	205.9	1	7.02	1
199	628	1.234	0	416.9	0	7.02	1
200	628	1.323	0	386.9	0	7.02	1
201	628	1.496	0	347	0	7.02	1
202	628	1.661	0	314.2	1	7.02	1
203	628	1.834	0	282.4	1	7.02	1
204	628	1.99	0	257	1	7.02	1
205	628	1.024	0	500.2	0	7.02	1
206	628	1.099	0	470.1	0	7.02	1
207	628	1.242	0	417.7	0	7.02	1
208	628	1.381	0	372.6	0	7.02	1
209	628	1.524	0	340.7	0	7.02	1
210	628	1.665	0	309.3	1	7.02	1
211	628	0.9847	1	522.3	0	7.02	1
212	628	1.055	0	485.8	0	7.02	1
213	628	1.189	0	433.4	0	7.02	1
214	628	1.324	0	388.2	0	7.02	1
215	628	1.461	0	356.3	0	7.02	1
216	628	1.597	0	324.9	0	7.02	1

Table Appendix B- 9 Contents of FAIL1X.DAT (Continued)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0.06087	1	0.05921	1	0.000372	0	0.004372	1	0.2472	1	1	0.2495	127.5	131.6
2	0	0.05918	1	0.05749	1	0.000392	0	0.003958	1	0.2469	1	1	0.2495	118.9	122.8
3	0	0.05693	1	0.05516	1	0.000432	0	0.003368	1	0.2464	1	1	0.2494	105.4	109.1
4	0	0.05562	1	0.05374	1	0.000476	0	0.002951	1	0.2459	1	1	0.2494	94.72	98.26
5	0	0.06448	1	0.06285	1	0.000338	0	0.005311	1	0.2476	1	1	0.2495	145.9	150.4
6	0	0.06224	1	0.06062	1	0.000353	0	0.004777	1	0.2474	1	1	0.2495	136.1	140.3
7	0	0.05911	1	0.05745	1	0.000384	0	0.004017	1	0.247	1	1	0.2495	120.7	124.6
8	0	0.05705	1	0.05532	1	0.000418	0	0.00348	1	0.2466	1	1	0.2495	108.6	112.3
9	0	0.05557	1	0.05376	1	0.000454	0	0.003068	1	0.2462	1	1	0.2494	98.46	102
10	0	0.05477	1	0.05285	1	0.000495	0	0.002759	1	0.2456	1	1	0.2494	90.03	93 48
11	0	0.06642	1	0.06481	1	0.000324	0	0.005845	1	0.2478	1	1	0.2496	155.8	160.4
12	0	0.06234	1	0.06074	1	0.000347	0	0.004861	1	0.2475	1	1	0.2495	138.2	142.5
13	0	0.05949	1	0.05786	1	0.000373	0	0.004167	1	0.2471	1	1	0.2495	124.4	128.3
14	0	0.05732	1	0.05565	1	0.000401	0	0.003637	1	0.2468	1	1	0.2495	112.8	116.6
15	0	0.05589	1	0.05415	1	0.000433	0	0.003239	1	0.2464	1	1	0.2494	103.2	106.8
16	0	0.06597	1	0.06438	1	0.000321	0	0.005809	1	0.2478	1	1	0.2496	155.8	160.4
17	0	0.06255	1	0.06097	1	0.000339	0	0.004995	1	0.2476	1	1	0.2495	141.5	145.8
18	0	0.06001	1	0.05842	1	0.000359	0	0.004381	1	0.2473	1	1	0.2495	129.5	133.5
19	0	0.06559	1	0.06402	1	0.000319	0	0.005771	1	0.2478	1	1	0.2496	155.7	160.3
20	1	0.06745	1	0.06686	1	0.000372	1	0.005	1	0.2508	1	1	0.2497	129.4	131.6
21	1	0.06532	1	0.06471	1	0.000392	1	0.004516	1	0.2507	1	1	0.2497	120.7	122.8
22	1	0.06238	1	0.06175	1	0.000432	1	0.003826	1	0.2506	1	1	0.2497	107.1	109.1
23	1	0.06054	1	0.05987	1	0.000476	1	0.003339	1	0.2504	1	1	0.2497	96.46	98.26
24	1	0.07202	1	0.07141	1	0.000338	1	0.006107	1	0.2509	1	1	0.2497	147.9	150.4
25	1	0.06927	1	0.06868	1	0.000353	1	0.00548	1	0.2509	1	1	0.2497	138	140.3
26	1	0.06534	1	0.06475	1	0.000384	1	0.004587	1	0.2508	1	1	0.2497	122.5	124.6
27	1	0.06266	1	0.06205	1	0.000418	1	0.003959	1	0.2506	1	1	0.2497	110.3	112.3
28	1	0.06067	1	0.06003	1	0.000454	1	0.003478	1	0.2505	1	1	0.2497	100.2	102
29	1	0.0745	1	0.07386	1	0.000324	1	0.006738	1	0.251	1	1	0.2497	157.8	160.5
30	1	0.06948	1	0.06889	1	0.000347	1	0.005581	1	0.2509	1	1	0.2497	140.1	142.5
31	1	0.0659	1	0.06532	1	0.000373	1	0.004766	1	0.2508	1	1	0.2497	126.2	128.3
32	1	0.06315	1	0.06255	1	0.000401	1	0.004145	1	0.2507	1	1	0.2497	114.6	116.6
33	1	0.06123	1	0.06062	1	0.000433	1	0.003678	1	0.2506	1	1	0.2497	104.9	106.8
34	1	0.07406	1	0.07342	1	0.000321	1	0.006701	1	0.251	1	1	0.2497	157.9	160.6
35	1	0.06986	1	0.06927	1	0.000339	1	0.005743	1	0.2509	1	1	0.2497	143.4	145.8
36	1	0.06669	1	0.06612	1	0.003594	1	0.005021	1	0.2509	1	1	0.2497	131.3	133.6
37	1	0.07369	1	0.07304	1	0.000319	1	0.006659	1	0.251	1	1	0.2497	157.7	160.4

Table Appendix B- 10 Contents of FAIL2X.DAT

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
20	0	0.6594	1	0.5276	1	0.000372	1	0.1239	1	0.7892	0	1	0.4989	407.1	1286
21	0	0.6385	1	0.5104	1	0.000392	1	0.1118	1	0.7892	0	1	0.4989	379.9	1200
22	0	0.6098	1	0.4865	1	0.000432	1	0.09468	1	0.7892	0	1	0.4989	337.4	1066
23	1	0.5918	1	0.4711	1	0.000476	1	0.08258	1	0.7892	1	1	0.4988	304	960.6
24	0	0.7039	1	0.5641	1	0.000338	1	0.1514	1	0.7892	0	1	0.499	465.2	1470
25	0	0.6771	1	0.5422	1	0.000353	1	0.1358	1	0.7892	0	1	0.499	434.1	1372
26	0	0.6387	1	0.5108	1	0.000384	1	0.1136	1	0.7892	0	1	0.4989	385.7	1219
27	0	0.6126	1	0.4891	1	0.000418	1	0.098	1	0.7892	0	1	0.4989	347.5	1098
28	1	0.5931	1	0.4727	1	0.000454	1	0.08604	1	0.7892	1	1	0.4988	315.7	997.4
29	0	0.728	1	0.5836	1	0.000324	1	0.1671	1	0.7892	0	1	0.499	496.2	1568
30	0	0.6791	1	0.544	1	0.000347	1	0.1383	1	0.7892	0	1	0.499	440.8	1393
31	0	0.6443	1	0.5155	1	0.000373	1	0.1181	1	0.7892	0	1	0.4989	397.1	1255
32	0	0.6173	1	0.4933	1	0.000401	1	0.1026	1	0.7892	0	1	0.4989	360.8	1140
33	0	0.5986	1	0.4776	1	0.000433	1	0.09104	1	0.7892	0	1	0.4989	330.6	1044
34	0	0.7237	1	0.5802	1	0.000321	1	0.1662	1	0.7892	0	1	0.499	496.4	1569
35	0	0.6828	1	0.5472	1	0.000339	1	0.1424	1	0.7892	0	1	0.499	451	1425
36	0	0.652	1	0.522	1	0.000359	1	0.1244	1	0.7892	0	1	0.4989	413.2	1306
37	0	0.72	1	0.5772	1	0.000319	1	0.1652	1	0.7892	0	1	0.499	495.9	1568

Table Appendix B- 11 Contents of FAIL3X.DAT

Logged failed combinations for Y axis component selection

Data ID	VeIn	VIpeak	Flag 1	Velocity	Flag 2	TPeak	Flag 3
1	314	0.7464	1	256.7	1	4.171	1
2	314	0.7957	1	240.4	1	4.205	1
3	314	0.8859	1	212.5	1	4.263	1
4	314	0.9707	1	190.1	1	4.323	1
5	314	1.053	0	174.3	1	4.385	1
6	314	1.129	0	158.7	1	4.446	1
7	314	0.659	1	292.9	1	4.113	1
8	314	0.7035	1	275	1	4.139	1
9	314	0.7857	1	242.7	1	4.19	1
10	314	0.8641	1	219.2	1	4.241	1
11	314	0.9411	1	198	1	4.291	1
12	314	1.014	0	183.1	1	4.347	1
13	314	0.5803	1	336.8	0	4.061	1
14	314	0.6202	1	313.2	1	4.083	1
15	314	0.6942	1	276.9	1	4.126	1
16	314	0.7654	1	251.6	1	4.17	1
17	314	0.8361	1	226.6	1	4.214	1
18	314	0.9046	1	206.3	1	4.26	1
19	314	0.4675	0	393.7	0	3.995	1
20	314	0.5002	1	392.9	0	4.01	1
21	314	0.5612	1	348	0	4.041	1
22	314	0.6205	1	314.4	1	4.074	1
23	314	0.68	1	284.4	1	4.107	1
24	314	0.7384	1	261.2	1	4.143	1
25	314	0.3911	0	395.2	0	3.956	1
26	314	0.4186	0	394.7	0	3.968	1
27	314	0.4702	0	393.8	0	3.99	1
28	314	0.5205	1	376.8	0	4.015	1
29	314	0.5713	1	341.8	0	4.041	1
30	314	0.6215	1	313.2	1	4.068	1
31	314	0.3757	0	395.4	0	3.949	1
32	314	0.4022	0	395	0	3.96	1
33	314	0.4518	0	394.2	0	3.981	1
34	314	0.5003	1	392.9	0	4.004	1
35	314	0.5492	1	356.8	0	4.028	1
36	314	0.5976	1	327.2	0	4.054	1
37	628	1.417	0	255.5	1	3.268	1
38	628	1.472	0	240.1	1	3.265	1
39	628	1.558	0	211.6	1	3.258	1
40	628	1.622	0	191.8	1	3.252	1
41	628	1.672	0	173.4	1	3.246	1
42	628	1.704	0	159	1	3.239	1
43	628	1.316	0	292	1	3.274	1
44	628	1.375	0	272.6	1	3.271	1
45	628	1.472	0	243.8	1	3.265	1
46	628	1.549	0	216.7	1	3.26	1
47	628	1.614	0	198.2	1	3.254	1
48	628	1.662	0	180.5	1	3.249	1
49	628	1.209	0	332.6	0	3.279	1
50	628	1.269	0	313.1	1	3.276	1
51	628	1.371	0	276	1	3.272	1
52	628	1.458	0	248.6	1	3.267	1
53	628	1.533	0	226.3	1	3.262	1
54	628	1.594	0	208.5	1	3.257	1
55	628	1.028	0	417.9	0	3.286	1

Table Appendix B- 12 Contents of FAIL1Y.DAT

56	628	1.086	0	389.3	0	3.284	1
57	628	1.188	0	347.5	0	3.28	1
58	628	1.279	0	311.3	1	3.276	1
59	628	1.363	0	284.9	1	3.273	1
60	628	1.437	0	259.1	1	3.269	1
61	628	0.8865	1	504.2	0	3.291	1
62	628	0.9403	1	470.8	0	3.289	1
63	628	1.037	0	414.8	0	3.286	1
64	628	1.126	0	374	0	3.283	1
65	628	1.21	0	338.7	0	3.28	1
66	628	1.287	0	312.8	1	3.276	1
67	628	0.8564	1	521.8	0	3.292	1
68	628	0.9091	1	488.4	0	3.29	1
69	628	1.004	0	432.4	0	3.287	1
70	628	1.091	0	391.5	0	3.284	1
71	628	1.175	0	356	0	3.281	1
72	628	1.252	0	325.7	0	3.278	1
73	314	0.716	1	254.8	1	7.749	1
74	314	0.767	1	236.7	1	7.864	1
75	314	0.863	1	206.5	1	8.088	1
76	314	0.9575	1	191.2	1	8.316	1
77	314	1.053	0	168.5	1	8.545	1
78	314	1.149	0	153.6	1	8.782	1
79	314	0.6271	1	293.3	1	7.525	1
80	314	0.6717	1	272.4	1	7.62	1
81	314	0.7556	1	238.6	1	7.807	1
82	314	0.8383	1	217.1	1	8.001	1
83	314	0.9223	1	195	1	8.199	1
84	314	1.006	0	180.5	1	8.405	1
85	314	0.5493	1	336.4	0	7.342	1
86	314	0.5883	1	311.6	1	7.419	1
87	314	0.6617	1	274	1	7.573	1
88	314	0.734	1	248.4	1	7.734	1
89	314	0.8074	1	228.1	1	7.901	1
90	314	0.881	1	206.8	1	8.078	1
91	314	0.4401	0	369.7	0	7.114	1
92	314	0.4714	0	369.6	0	7.167	1
93	314	0.5303	1	348.5	0	7.274	1
94	314	0.5881	1	312.6	1	7.388	1
95	314	0.6469	1	282.5	1	7.508	1
96	314	0.7056	1	258.6	1	7.638	1
97	314	0.3672	0	369.8	0	6.983	1
98	314	0.3934	0	369.7	0	7.021	1
99	314	0.4425	0	369.7	0	7.099	1
100	314	0.4908	0	369.6	0	7.183	1
101	314	0.5398	1	341.3	0	7.272	1
102	314	0.5888	1	313.2	1	7.369	1
103	314	0.3526	0	369.8	0	6.959	1
104	314	0.3777	0	369.8	0	6.995	1
105	314	0.4249	0	369.7	0	7.067	1
106	314	0.4713	0	369.6	0	7.145	1
107	314	0.5184	1	356.8	0	7.228	1
108	314	0.5654	1	327.1	0	7.319	1
109	628	1.675	0	252.1	1	7.333	1
110	628	1.804	0	237.8	1	7.33	1
111	628	2.045	0	210.9	1	7.323	1

Table Appendix B- 12 Contents of FAIL1Y.DAT (Continued)

112	628	2.288	0	185.8	1	7.317	1
113	628	2.538	0	169	1	7.311	1
114	628	2.791	0	159	1	7.304	1
115	628	1.454	0	290.2	1	7.339	1
116	628	1.562	0	267.8	1	7.336	1
117	628	1.773	0	240.9	1	7.33	1
118	628	1.981	0	215.4	1	7.325	1
119	628	2.194	0	198.2	1	7.319	1
120	628	2.412	0	181.7	1	7.314	1
121	628	1.26	0	328.4	0	7.344	1
122	628	1.355	0	314.2	1	7.341	1
123	628	1.536	0	278.9	1	7.337	1
124	628	1.717	0	245.1	1	7.332	1
125	628	1.9	0	228.1	1	7.327	1
126	628	2.086	0	204.2	1	7.322	1
127	628	0.9984	1	418.1	0	7.351	1
128	628	1.074	0	386.3	0	7.349	1
129	628	1.213	0	342.2	0	7.345	1
130	628	1.355	0	307.9	1	7.341	1
131	628	1.497	0	283	1	7.338	1
132	628	1.642	0	258.6	1	7.334	1
133	628	0.8276	1	505.1	0	7.356	1
134	628	0.888	1	463.8	0	7.354	1
135	628	1.003	0	410.4	0	7.351	1
136	628	1.118	0	376	0	7.348	1
137	628	1.235	0	342.3	0	7.345	1
138	628	1.354	0	309.2	1	7.341	1
139	628	0.7932	1	517.2	0	7.357	1
140	628	0.8521	1	485.2	0	7.355	1
141	628	0.9623	1	431.6	0	7.352	1
142	628	1.072	0	388.2	0	7.349	1
143	628	1.184	0	354.5	0	7.346	1
144	628	1.298	0	321.4	1	7.343	1
145	314	0.6979	1	250.2	1	10.72	0
146	314	0.7466	1	239.1	1	10.86	0
147	314	0.8379	1	208.2	1	11.14	0
148	314	0.9271	1	185.4	1	11.43	0
149	314	1.017	0	171.3	1	11.73	0
150	314	1.107	0	157.3	1	12.04	0
151	314	0.6125	1	291.8	1	10.44	0
152	314	0.6555	1	268.9	1	10.56	0
153	314	0.7361	1	240.6	1	10.79	0
154	314	0.8148	1	218.9	1	11.03	0
155	314	0.8942	1	196.9	1	11.28	0
156	314	0.9732	1	183.1	1	11.54	0
157	314	0.5373	1	335.6	0	10.23	0
158	314	0.5752	1	312	1	10.32	0
159	314	0.6461	1	277.2	1	10.5	0
160	314	0.7155	1	250.6	1	10.7	0
161	314	0.7857	1	221.5	1	10.9	0
162	314	0.8554	1	208.5	1	11.12	0
163	314	0.4312	0	352	0	9.958	0
164	314	0.4617	0	352	0	10.02	0
165	314	0.519	1	348.6	0	10.14	0
166	314	0.5752	1	312.8	1	10.28	0
167	314	0.632	1	278.8	1	10.42	0
168	314	0.6885	1	260.8	1	10.58	0

Table Appendix B- 12 Contents of FAIL1Y.DAT (Continued)

169	314	0.3599	0	352.1	0	9.806	1
170	314	0.3855	0	352.1	0	9.85	1
171	314	0.4335	0	352	0	9.94	0
172	314	0.4806	0	352	0	10.04	0
173	314	0.5283	1	340.5	0	10.14	0
174	314	0.5759	1	313.2	1	10.26	0
175	314	0.3456	0	352.1	0	9.779	1
176	314	0.3702	0	352.1	0	9.82	1
177	314	0.4163	0	352.1	0	9.903	0
178	314	0.4616	0	352	0	9.994	0
179	314	0.5075	1	356.6	0	10.09	0
180	314	0.5532	1	324.1	0	10.2	0
181	628	2.031	0	256.1	1	7.067	1
182	628	2.183	0	239.5	1	6.97	1
183	628	2.382	0	213.4	1	6.963	1
184	628	2.527	0	188.6	1	6.957	1
185	628	2.653	0	170.1	1	6.951	1
186	628	2.754	0	157.1	1	6.944	1
187	628	1.77	0	290.6	1	7.061	1
188	628	1.901	0	273.9	1	7.064	1
189	628	2.147	0	242	1	7.07	1
190	628	2.352	0	217	1	6.965	1
191	628	2.491	0	198.3	1	6.959	1
192	628	2.608	0	180.1	1	6.954	1
193	628	1.543	0	330.8	0	7.056	1
194	628	1.659	0	314.2	1	7.059	1
195	628	1.869	0	276.2	1	7.063	1
196	628	2.082	0	251	1	7.068	1
197	628	2.296	0	226.6	1	6.967	1
198	628	2.435	0	208.4	1	6.962	1
199	628	1.233	0	420.9	0	7.049	1
200	628	1.321	0	391.7	0	7.051	1
201	628	1.487	0	347.2	0	7.055	1
202	628	1.656	0	309.8	1	7.059	1
203	628	1.826	0	285.4	1	7.062	1
204	628	1.997	0	261.3	1	7.066	1
205	628	1.025	0	502.2	0	7.044	1
206	628	1.096	0	466.5	0	7.046	1
207	628	1.237	0	415.5	0	7.049	1
208	628	1.377	0	371.6	0	7.052	1
209	628	1.514	0	341	0	7.055	1
210	628	1.659	0	310.8	1	7.059	1
211	628	0.9815	1	524	0	7.043	1
212	628	1.053	0	488.2	0	7.045	1
213	628	1.187	0	430.6	0	7.048	1
214	628	1.318	0	386.7	0	7.051	1
215	628	1.455	0	356.1	0	7.054	1
216	628	1.59	0	325.9	0	7.057	1

Table Appendix B- 12 Contents of FAIL1Y.DAT (Continued)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0.05518	1	0.05367	1	0.000329	0	0.003838	1	0.2417	1	1	0.2379	124.7	131.6
2	0	0.05307	1	0.05157	1	0.000342	0	0.003421	1	0.2407	1	0	0.2362	115.9	122.8
3	0	0.05337	1	0.04854	1	0.000368	0	0.002824	1	0.2386	1	0	0.2326	102	116.3
4	0	0.05924	1	0.04644	1	0.000398	0	0.0024	1	0.2362	1	0	0.2287	90.99	121.4
5	1	0.0595	1	0.05797	1	0.000305	1	0.004783	1	0.2434	1	1	0.2407	143.5	150.4
6	0	0.0569	1	0.0554	1	0.000315	0	0.004246	1	0.2426	1	1	0.2394	133.5	140.3
7	0	0.05308	1	0.0516	1	0.000335	0	0.00348	1	0.241	1	0	0.2366	117.8	124.6
8	0	0.05184	1	0.04887	1	0.000358	0	0.002938	1	0.2392	1	0	0.2336	105.3	115.6
9	0	0.05705	1	0.04676	1	0.000382	0	0.002523	1	0.2372	1	0	0.2302	94.86	120.7
10	1	0.06175	1	0.06022	1	0.000295	1	0.005318	1	0.2441	1	1	0.2418	153.5	160.4
11	1	0.05707	1	0.05559	1	0.000310	1	0.00433	1	0.2428	1	1	0.2397	135.6	142.5
12	0	0.05362	1	0.05216	1	0.000327	0	0.003632	1	0.2415	1	0	0.2374	121.5	128.3
13	0	0.05091	1	0.04946	1	0.000346	0	0.003101	1	0.2399	1	0	0.2348	109.7	116.6
14	0	0.05449	1	0.04743	1	0.000367	0	0.002697	1	0.2382	1	0	0.2319	99.77	119.1
15	1	0.06128	1	0.05977	1	0.000292	1	0.005281	1	0.2441	1	1	0.2418	153.6	160.4
16	1	0.05743	1	0.05596	1	0.000304	1	0.004467	1	0.2432	1	1	0.2401	138.9	145.8
17	0	0.05441	1	0.05297	1	0.000317	0	0.00385	1	0.2421	1	1	0.2383	126.7	133.5
18	1	0.06092	1	0.05944	1	0.000290	1	0.005246	1	0.2442	1	1	0.2418	153.4	160.3
19	1	0.0618	1	0.06125	1	0.000329	1	0.00453	1	0.249	1	1	0.2458	128.4	131.6
20	1	0.05922	1	0.0587	1	0.000342	1	0.004044	1	0.2486	1	1	0.2452	119.7	122.8
21	1	0.0555	1	0.055	1	0.000368	1	0.003351	1	0.2479	1	1	0.2439	106	109.1
22	1	0.05924	1	0.05238	1	0.000398	1	0.002862	1	0.2471	1	1	0.2426	95.19	110.1
23	1	0.06716	1	0.06649	1	0.000305	1	0.005639	1	0.2495	1	1	0.2467	147.1	150.7
24	1	0.064	1	0.06341	1	0.000315	1	0.005011	1	0.2493	1	1	0.2462	137.1	140.5
25	1	0.05934	1	0.05882	1	0.000335	1	0.004116	1	0.2487	1	1	0.2453	121.5	124.7
26	1	0.05596	1	0.05547	1	0.000358	1	0.003484	1	0.2481	1	1	0.2443	109.2	112.3
27	1	0.05705	1	0.05286	1	0.000382	1	0.003005	1	0.2474	1	1	0.2431	98.97	109.1
28	1	0.07	1	0.06924	1	0.000295	1	0.006271	1	0.2498	1	1	0.2471	157	160.9
29	1	0.0643	1	0.06369	1	0.000310	1	0.005111	1	0.2493	1	1	0.2463	139.2	142.7
30	1	0.06008	1	0.05954	1	0.000327	1	0.004294	1	0.2489	1	1	0.2456	125.2	128.4
31	1	0.05675	1	0.05626	1	0.003466	1	0.003674	1	0.2483	1	1	0.2447	113.5	116.6
32	1	0.05449	1	0.05375	1	0.000367	1	0.003206	1	0.2478	1	1	0.2437	103.8	107.3
33	1	0.06955	1	0.06878	1	0.000292	1	0.006231	1	0.2498	1	1	0.2471	157.1	161
34	1	0.06486	1	0.06421	1	0.000304	1	0.005276	1	0.2494	1	1	0.2465	142.5	146.1
35	1	0.06116	1	0.0606	1	0.000317	1	0.004552	1	0.2491	1	1	0.2459	130.4	133.7
36	1	0.06921	1	0.06842	1	0.000290	1	0.006193	1	0.2498	1	1	0.2471	156.9	160.9
37	1	0.07133	1	0.07048	1	0.000285	1	0.006649	1	0.2499	1	1	0.2473	163.6	167.7

Table Appendix B- 13 Contents of FAIL2Y.DAT

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
5	0	0.5816	1	0.4493	1	0.000305	1	0.1192	1	0.7844	0	1	0.493	462.4	1470
10	0	0.6037	1	0.4671	1	0.000295	1	0.1323	1	0.785	0	1	0.4937	493.5	1568
11	0	0.5579	1	0.4306	1	0.000310	1	0.1081	1	0.7838	0	1	0.4925	437.7	1393
15	0	0.599	1	0.4638	1	0.000292	1	0.1314	1	0.785	0	1	0.4937	493.7	1568
16	0	0.5614	1	0.4337	1	0.000304	1	0.1115	1	0.7841	0	1	0.4928	448	1425
18	0	0.5956	1	0.4613	1	0.000290	1	0.1306	1	0.785	0	1	0.4937	493.2	1567
19	0	0.6039	1	0.4839	1	0.000329	1	0.1132	1	0.7871	0	1	0.497	406	1287
20	0	0.5789	1	0.4636	1	0.000342	1	0.1012	1	0.7868	0	1	0.4966	378.7	1200
21	0	0.5425	1	0.4339	1	0.000368	1	0.08401	1	0.7861	0	1	0.496	336.1	1066
22	1	0.5169	1	0.4128	1	0.000398	1	0.0719	1	0.7854	1	1	0.4953	302.5	960.6
23	0	0.6559	1	0.5257	1	0.000305	1	0.1408	1	0.7876	0	1	0.4975	464.3	1472
24	0	0.6253	1	0.5012	1	0.000315	1	0.1252	1	0.7874	0	1	0.4972	433.1	1373
25	0	0.58	1	0.4646	1	0.000335	1	0.103	1	0.7868	0	1	0.4967	384.5	1219
26	0	0.547	1	0.4377	1	0.000358	1	0.08731	1	0.7863	0	1	0.4962	346.2	1098
27	1	0.5215	1	0.4168	1	0.000382	1	0.07543	1	0.7857	1	1	0.4956	314.3	997.4
28	0	0.6832	1	0.5476	1	0.000295	1	0.1564	1	0.7878	0	1	0.4977	495.3	1570
29	0	0.6281	1	0.5034	1	0.000310	1	0.1277	1	0.7874	0	1	0.4973	439.7	1394
30	0	0.5871	1	0.4704	1	0.000327	1	0.1074	1	0.787	0	1	0.4969	396	1255
31	0	0.5547	1	0.4441	1	0.000346	1	0.09202	1	0.7865	0	1	0.4964	359.5	1140
32	0	0.5302	1	0.424	1	0.003676	1	0.08041	1	0.786	0	1	0.4959	329.2	1044
33	0	0.6787	1	0.544	1	0.000292	1	0.1555	1	0.7878	0	1	0.4977	495.5	1571
34	0	0.6334	1	0.5077	1	0.000304	1	0.1317	1	0.7875	0	1	0.4974	450	1427
35	0	0.5976	1	0.4789	1	0.000317	1	0.1138	1	0.7872	0	1	0.497	412.1	1306
36	0	0.6753	1	0.5412	1	0.000290	1	0.1545	1	0.7878	0	1	0.4977	495	1570
37	0	0.6958	1	0.5575	1	0.000285	1	0.1658	1	0.788	0	1	0.4978	515.7	1636

Table Appendix B- 14 Contents of FAIL3Y.DAT

RESULTS OF VISSIM MODELS FOR FAILURE ANALYSIS

The section displays the output files created by the VISSIM models for failure analysis. Again, the files are in a tabular format, each of the columns relating to a particular component or function, and each row relating to a particular combination of behavioural modes. For the first, second and third generation models, this relationship is shown in Table Appendix B 15. For numbers relating to behavioural mode information see table 9-2.

Column No.	Component/ Function	No. of behavioural modes
1	New combination ID	
2	Switch	2
3	Latched Emergency stop	3
4	Power Supply Unit	2
5	Limit Switches	4
6	Reference Voltage and Amplifier	2
7	Gain	2
8	Encoder	3
9	Tachometer	3
10	Mechanical Failures	3
11	Original combination ID from 5184f.MAP	
12	Time failure detected	

Table Appendix B- 15 Column position against component/function

First Generation failure model

Results for 1st generation model, FC1

Combination	1	2	3	4	5	6	7	8	9	ID No.	Time
1	1	0	0	0	0	0	0	0	0	1	0.0615
2	0	1	0	0	0	0	0	0	0	2	0.2405
3	0	2	0	0	0	0	0	0	0	4	0.2405
4	0	0	1	0	0	0	0	0	0	6	0.0625
5	0	0	0	1	0	0	0	0	0	12	0.1375
6	0	0	0	2	0	0	0	0	0	24	0.1055
7	0	0	0	3	0	0	0	0	0	36	0.0835
8	0	0	0	0	1	0	0	0	0	48	0.0615
9	0	0	0	0	0	1	0	0	0	96	0.0645
10	0	0	0	0	0	0	1	0	0	192	0.1805
11	0	0	0	0	0	0	2	0	0	384	0.1805
12	0	0	0	0	0	0	0	1	0	576	0.1805
13	0	0	0	0	0	0	0	2	0	1152	0.1805

Table Appendix B- 16, FS114.DAT , SPF, Model 1, FC1

No remaining double point failures

Results for 1st generation model, FC2

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	1	0	0	0	0	0	0	0	0	1	0.127
2	0	2	0	0	0	0	0	0	0	4	0.241
3	0	0	1	0	0	0	0	0	0	6	0.127
4	0	0	0	1	0	0	0	0	0	12	0.1375
5	0	0	0	2	0	0	0	0	0	24	0.127
6	0	0	0	3	0	0	0	0	0	36	0.127
7	0	0	0	0	1	0	0	0	0	48	0.1275
8	0	0	0	0	0	1	0	0	0	96	0.1275
9	0	0	0	0	0	0	1	0	0	192	0.1805
10	0	0	0	0	0	0	2	0	0	384	0.1805
11	0	0	0	0	0	0	0	1	0	576	0.186
12	0	0	0	0	0	0	0	2	0	1152	0.185

Table Appendix B- 17, FC116.DAT SPF, Model 1, FC2

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	1	0	0	0	0	0	0	1	578	0.127
2	0	1	0	0	0	0	0	0	2	1154	0.127

Table Appendix B- 18, FC126.DAT DPF, Model 1, FC2

No remaining failure modes.

Second Generation failure model

Results for 2nd generation model, FC1

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	1	0	0	0	0	0	0	0	0	1	0.0615
2	0	1	0	0	0	0	0	0	0	2	0.2405
3	0	2	0	0	0	0	0	0	0	4	0.2405
4	0	0	1	0	0	0	0	0	0	6	0.0625
5	0	0	0	0	1	0	0	0	0	48	0.0645
6	0	0	0	0	0	1	0	0	0	96	0.0645

Table Appendix B- 19, FC214.DAT, SPF, Model 2, FC1

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	0	0	0	0	0	1	1	0	1920	0.1805

Table Appendix B- 20, FC2244.DAT, DPF, Model 2, FC1

No further failures discovered, checked up to quadruple point failures.

Results for 2nd generation model, FC2

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	0	1	0	0	0	0	0	0	6	0.0625
2	0	0	0	0	0	0	1	0	0	192	0.1805
3	0	0	0	0	0	0	2	0	0	384	0.1805
4	0	0	0	0	0	0	0	0	1	1728	0.1805

Table Appendix B- 21, FC216.DAT, SPF, Model 2, FC2

No further failures discovered, checked up to six point failures.

Third Generation failure model

Results for 3nd generation model, FC1

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	1	0	0	0	0	0	0	0	3	0.2415
2	0	0	0	0	1	0	0	0	0	9	0.0615
3	0	0	0	0	0	1	0	0	0	10	0.0655

Table Appendix B- 22, FC314.DAT, SPF, Model 3, FC1

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	0	0	0	0	0	2	1	0	2112	0.1805

Table Appendix B- 23, FC3244.DAT, DPF, Model 3, FC1

No further failure modes found, checked up to 7 point failures

Results for 3nd generation model, FC2

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	2	0	0	0	0	0	0	0	4	0.2405
2	0	0	0	0	0	1	0	0	0	10	0.086
3	0	0	0	0	0	0	1	0	0	11	0.1805
4	0	0	0	0	0	0	2	0	0	12	0.1835
5	0	0	0	0	0	0	0	0	1	15	0.1805

Table Appendix B- 24, FC316.DAT, SPF, Model 3, FC1

ID no.	1	2	3	4	5	6	7	8	9	ID No.	Time
1	0	1	0	0	1	0	0	0	0	50	0.092
2	0	0	1	0	1	0	0	0	0	54	0.102
3	0	1	0	0	0	0	0	0	2	1154	0.1805
4	0	0	0	0	1	0	0	0	2	1200	0.099
5	0	0	0	0	1	0	0	1	0	1776	0.096
6	0	0	0	0	0	0	0	1	2	2880	0.1805
7	0	0	0	0	1	0	0	2	0	3504	0.096

Table Appendix B- 25, FC3266.DAT, DPF, Model 3, FC2

No remaining failure modes.

Fourth Generation failure model

Because the fourth generation model had extra components added to the system Table Appendix B-15 is no longer valid. Table Appendix B-26 details the new relationship between output file columns and components. Details of the numbers relating to behavioural modes can be seen in Table 9-8.

Column No.	Component/ Function	No. of behavioural modes
1	New combination ID	
2	12 V Power Supply Unit	2
3	5 V Power Supply Unit	2
4	On/Off Switch	3
5	Emergency Stop	3
6	Limit Switches	4
7	Reference Voltage	2
8	Fan	3
9	Thermistor	4
10	Displacement Wire Potentiometer	3
11	Motor Brake	3
12	Encoder	2
13	Tachometer	3
14	Mechanical failure	3
15	Original Combination ID	
16	Time Failure Detected	

Table Appendix B- 26 Column position against component/function

Results for 4th generation model, FC1

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	ID	Time
1	0	0	0	0	0	1	0	0	0	0	0	0	0	11	0.0695

Table Appendix B- 27, FC414.DAT, SPF, Model 4, FC1

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	ID	Time
1	0	0	0	0	1	0	0	0	1	0	1	0	0	611	0.121
2	0	0	0	0	2	0	0	0	1	0	1	0	0	612	0.121
3	0	0	0	0	3	0	0	0	1	0	1	0	0	613	0.121
4	0	0	0	0	3	0	0	0	2	0	1	0	0	626	0.1225

Table Appendix B- 28, FC43FC1.DAT, TPF, Model 4, FC1

No further failure modes, checked up to 9 point failures.

Results for 4th generation model, FC2

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	ID	Time
1	0	0	0	0	0	0	0	0	1	0	0	2	0	216	0.182
2	0	0	0	0	0	0	0	0	0	0	0	2	1	242	0.18

Table Appendix B- 29, FC42FC2.DAT, DPF, Model 4, FC2

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	ID	Time
1	1	0	0	0	1	0	0	0	2	0	1	0	0	2316	0.12
2	1	0	0	0	3	0	0	0	2	0	1	0	0	2328	0.12
3	1	0	0	0	0	1	0	0	2	0	1	0	0	2334	0.12
4	1	0	0	0	0	0	0	2	2	0	1	0	0	2375	0.12
5	1	0	0	0	0	0	0	3	2	0	1	0	0	2387	0.12
6	1	0	0	0	0	0	0	0	2	0	1	1	0	3377	0.12

Table Appendix B- 30, FC44FC2.DAT, QPF, Model 4, FC2

No further failure modes, checked up to 9 point failures.

APPENDIX C: VISSIM MODELS AND SUPPORTING FILES

Because the size of many of the VISSIM models used throughout this thesis are extremely large, many in excess of 200 pages, it was felt that reproducing them would be impractical, however it is still necessary that they be included in the thesis. To achieve this, the models and supporting files are contained on two disks, A and B, this appendix details the contents of those disks.

The VISSIM models can be examined by opening them using VISSIM Viewer. This program is contained, on a third disk, in a self extracting zip file which should be loaded onto the readers hard disk and then clicked onto, to open the file. The viewer itself is initiated by clicking onto VISSIM32.EXE and then using the FILE command to open the appropriate document. The system hierarchy is shown down the left hand side of the viewer and any part of the system by clicking into the appropriate box. MAP and DAT files can be viewed using the Windows Notebook or Wordpad.

DISK VISSIM

VSVIEWER.EXE	Self extracting zip file containing VISSIM viewer and associated files

DISK A

A:\WORKED EXAMPLES:- VISSIM models and support files for the example programs used to describe component selection and failure analysis in chapter 5.

A:\WORKED EXAMPLES\PNUSELECT:- VISSIM models and supporting files for the selection of components for the pneumatic lifting system detailed in chapter 5.

PNUEXSW.VSM	VISSIM program used to generate component combination file PNUSW.MAP.
PNUEMEX.VSM	VISSIM model for component selection.
PNUSW.MAP	Component combination file used in PNUEMEX.VSM.
PNUEMEX.DAT	File containing successful combinations, produced by PNUEMEX.VSM.
PISTONS.MAP	Piston data, used in PNUEMEX.VSM
RADIUS_1.MAP	Pulley 1 data, used in PNUEMEX.VSM
RADIUS_2.MAP	Pulley 2 data, used in PNUEMEX.VSM

A:\WORKED EXAMPLES\FAILEX:- Failure models and supporting files for latched on switch and emergency stop.

FAILEX1.VSM	Basic VISSIM model of system
FAILEX2.VSM	Expanded model of FAILEX1.VSM. Includes failure behaviour and supporting files.
FAILINPS.VSM	VISSIM model used to create system operating sequence map, FAILINPS.MAP.

FAILSEL.VSM	VISSIM model used to create failure combination file, FAILSEL.MAP.
FAILSEL.MAP	Failure combination failure combination used in FAILEX2.VSM.
FAILINPS.MAP	System operating sequence file for FAILEX2.VSM.
FAILEX2.DAT	Complete logged failure combinations, produced by FAILEX2.VSM, from which a complete Fault Tree can be derived.
FAILRFT.DAT	File created from edited version of FAILEX2.DAT, lists failure combinations from which a Reduced Fault Tree is derived.

A:\COMMON MAP FILES:- Common .MAP files used to support VISSIM models for component selection and failure modelling, see chapters 8 and 9.

5184F.MAP	Failure combinations for 1st ,2nd and 3rd generation models.
GB.MAP.MAP	Gearbox data, used in all models.
LIMITS1.MAP	Controls failure states of the limits switches.
LREF.MAP	Look up table used to limit reference voltage when datuming axis in 1st generation model.
LVMAP.MAP	Look up table used to limit the velocity of the load between limit switches in 1st generation model.
LVMAP1..MAP	Look up table used to limit the velocity of the load between limit switches in 2nd , 3rd and 4th generation models.
MOTOR.MAP	Motor data , used in all models.
PULLEY.MAP	Pulley data, used in all models.
SETPOINT.MAP	Lookup table used to control velocity whilst datuming in 2nd , 3rd and 4th generation models.
SWEEP.MAP	216 component combinations used in initial model for component selection.

A:\COMPONENT SELECTION:- Contains VISSIM models for component selection for the X and Y axis.

DFDSW.VSM	VISSIM model used to create SWEEP.MAP for component combinations.

A:\COMPONENT SELECTION\ VISSIMX :- VISSIM models for component selection for the X axis .

FAIL1X.DAT	Failure data for 1st stage component selection.
FAIL2X.DAT	Failure data for 2nd stage component selection.
FAIL3X.DAT	Failure data for 3rd stage component selection.
STAGE1X.VSM	VISSIM model for 1st stage component selection, X axis.
STAGE2X.VSM	VISSIM model for 2nd stage component selection, X axis.
STAGE3X.VSM	VISSIM model for 3rd stage component selection, X axis.
STAGE1X.MAP	Successful component combinations for 1st stage component selection, also used to drive 2nd stage model.
STAGE2X.MAP	Successful component combinations for 2nd stage component selection, also used to drive 3rd stage model.

STAGE3X.MAP Successful component combinations for 3rd stage component selection.

A:\COMPONENT SELECTION\ VISSIMY :- VISSIM models for component selection for the Y axis.

FAIL1Y.DAT Failure data for 1st stage component selection.

FAIL2Y.DAT Failure data for 2nd stage component selection.

FAIL3Y.DAT Failure data for 3rd stage component selection.

STAGE1Y.VSM VISSIM model for 1st stage component selection, Y axis.

STAGE2Y.VSM VISSIM model for 2nd stage component selection, Y axis.

STAGE3Y.VSM VISSIM model for 3rd stage component selection, Y axis.

STAGE1Y.MAP Successful component combinations for 1st stage component selection, also used to drive 2nd stage model.

STAGE2Y.MAP Successful component combinations for 2nd stage component selection, also used to drive 3rd stage model.

STAGE3Y.MAP Successful component combinations for 3rd stage component selection.

A:\SYSTEM FAILURE:- Contains VISSIM programs and supporting files for 4 generations of failure model.

SWFILT2D.VSM First generation VISSIM model used to generate failure combination files.

FMEASW1A.VSM Advanced VISSIM model used to generate failure combination files.

SPFAILS.MAP Single point failure combination file, used by 1st, 2nd and 3rd generation models.

A:\SYSTEM FAILURE\1ST GENERATION MODEL

STAGES1.VSM First generation failure model.

A:\SYSTEM FAILURE\1ST GENERATION MODEL\FC1:- Failure combination MAP and DAT files for Velocity >0.0001 M/s after 0.06 from the time failure is detected.

FS114.DAT Single point failure combinations which produce system failure for FC1, 1st generation model.

A:\SYSTEM FAILURE\1ST GENERATION MODEL\FC2:- Failure combination MAP and DAT files for apparent load position +/- 10% different from actual position.

DPFAIL2. MAP Double point failure combinations for FC2, 1st generation model.

FS116.DAT Single point failure combinations which produce system failure for FC2, 1st generation model.

FS126.DAT Double point failure combinations which produce system failure for FC2, 1st generation model.

A:\SYSTEM FAILURE\2nd GENERATION MODEL

STAGES2.VSM Second generation failure model.

A:\SYSTEM FAILURE\2nd GENERATION MODEL\FC1:- Failure combination MAP and DAT files for Velocity >0.0001 M/s after 0.06 from the time failure is detected.

DP2FAIL4. MAP Double point failure combinations for FC1, 2nd generation model.

TP2FAIL4. MAP	Triple point failure combinations for FC1, 2nd generation model.
QP2FAIL4. MAP	Quadruple point failure combinations for FC1, 2nd generation model.
FS214.DAT	Single point failure combinations which produce system failure for FC1, 2nd generation model.
FS2244.DAT	Double point failure combinations which produce system failure for FC1, 2nd generation model.
FS2344.DAT	Triple point failure combinations which produce system failure for FC1, 2nd generation model.
FS2444.DAT	Quadruple point failure combinations which produce system failure for FC1, 2nd generation model.

A:\SYSTEM FAILURE\2nd GENERATION MODEL\FC2:- Failure combination MAP and DAT files for apparent load position +/- 10% different from actual position.

DP2FAIL6. MAP	Double point failure combinations for FC1, 2nd generation model.
TP2FAIL6. MAP	Triple point failure combinations for FC1, 2nd generation model.
QP2FAIL6. MAP	Quadruple point failure combinations for FC1, 2nd generation model.
FS216.DAT	Single point failure combinations which produce system failure for FC2, 2nd generation model.
FS2266.DAT	Double point failure combinations which produce system failure for FC2, 2nd generation model.
FS2366.DAT	Triple point failure combinations which produce system failure for FC2, 2nd generation model.
FS2466.DAT	Quadruple point failure combinations which produce system failure for FC2, 2nd generation model.

A:\SYSTEM FAILURE\3rd GENERATION MODEL

STAGES3.VSM	Third generation failure model.

A:\SYSTEM FAILURE\3rd GENERATION MODEL\FC1:- Failure combination MAP and DAT files for Velocity >0.0001 M/s after 0.06 from the time failure is detected.

DPFAIL3FC1.MAP	Double point failure combinations for FC1, 3rd generation model.
TPFAIL3FC1. MAP	Triple point failure combinations for FC1, 3rd generation model.
QPFAIL3FC1. MAP	Quadruple point failure combinations for FC1, 3rd generation model.
5PFAIL3FC1. MAP	Five point failure combinations for FC1, 3rd generation model.
6PFAIL3FC1. MAP	Six point failure combinations for FC1, 3rd generation model.
7PFAIL3FC1. MAP	Seven point failure combinations for FC1, 3rd generation model.
FS314.DAT	Single point failure combinations which produce system failure for FC1, 3rd generation model.
FS3244.DAT	Double point failure combinations which produce system failure for FC1, 3rd generation model.
FS3344.DAT	Triple point failure combinations which produce system failure

for FC1, 3rd generation model.

FS3444.DAT — Quadruple point failure combinations which produce system failure for FC1, 3rd generation model.

FS3544.DAT — Five point failure combinations which produce system failure for FC1, 3rd generation model.

FS3644.DAT — Six point failure combinations which produce system failure for FC1, 3rd generation model.

FS3744.DAT — Seven point failure combinations which produce system failure for FC1, 3rd generation model.

A:\SYSTEM FAILURE\3rd GENERATION MODEL\FC2:- Failure combination MAP and DAT files for apparent load position +/- 10% different from actual position.

DPFAIL3FC2.MAP — Double point failure combinations for FC2, 3rd generation model.

FS316.DAT — Single point failure combinations which produce system failure for FC2, 3rd generation model.

FS3266.DAT — Double point failure combinations which produce system failure for FC2, 3rd generation model.

DISK B

A:\SYSTEM FAILURE\3rd GENERATION MODEL

STAGES4.VSM — Fourth generation failure model.

A:\SYSTEM FAILURE\4th GENERATION MODEL\FC1:- Failure combination MAP and DAT files for Velocity >0.0001 M/s after 0.06 from the time failure is detected.

DP4FAIL4.MAP — Double point failure combinations for FC1, 4th generation model.

TP4FAIL4. MAP — Triple point failure combinations for FC1, 4th generation model.

QP4FAIL4. MAP — Quadruple point failure combinations for FC1, 4th generation model.

5P4FAIL4. MAP — Five point failure combinations for FC1, 4th generation model.

6PFAIL4. MAP — Six point failure combinations for FC1, 4th generation model.

7PFAIL4. MAP — Seven point failure combinations for FC1, 4th generation model.

8PFAIL4. MAP — Eight point failure combinations for FC1, 4th generation model.

9PFAIL4. MAP — Nine point failure combinations for FC1, 4th generation model.

FS414.DAT — Single point failure combinations which produce system failure for FC1, 4th generation model.

FS42FC1.DAT — Double point failure combinations which produce system failure for FC1, 4th generation model.

FS43FC1.DAT — Triple point failure combinations which produce system failure for FC1, 4th generation model.

FS44FC1.DAT — Quadruple point failure combinations which produce system failure for FC1, 4th generation model.

FS45FC1.DAT — Five point failure combinations which produce system failure for FC1, 4th generation model.

FS46FC1.DAT — Six point failure combinations which produce system failure for

FC1, 4th generation model.

FS47FC1.DAT — Seven point failure combinations which produce system failure for FC1, 4th generation model.

FS48FC1.DAT — Eight point failure combinations which produce system failure for FC1, 4th generation model.

FS49FC1.DAT — Nine point failure combinations which produce system failure for FC1, 4th generation model.

A:\SYSTEM FAILURE\4th GENERATION MODEL\FC2:- Failure combination MAP and DAT files for apparent load position +/- 10% different from actual position.

DPFAIL6.MAP — Double point failure combinations for FC2, 4th generation model.

TPFAIL6. MAP — Triple point failure combinations for FC2, 4th generation model.

QPFAIL6. MAP — Quadruple point failure combinations for FC2, 4th generation model.

5PFAIL6. MAP — Five point failure combinations for FC2, 4th generation model.

6PFAIL6. MAP — Six point failure combinations for FC2, 4th generation model.

7PFAIL6. MAP — Seven point failure combinations for FC2, 4th generation model.

8PFAIL6. MAP — Eight point failure combinations for FC2, 4th generation model.

FS416.DAT — Single point failure combinations which produce system failure for FC1, 4th generation model.

FS42FC2.DAT — Double point failure combinations which produce system failure for FC1, 4th generation model.

FS43FC2.DAT — Triple point failure combinations which produce system failure for FC1, 4th generation model.

FS44FC2.DAT — Quadruple point failure combinations which produce system failure for FC1, 4th generation model.

FS45FC2.DAT — Five point failure combinations which produce system failure for FC1, 4th generation model.

FS46FC2.DAT — Six point failure combinations which produce system failure for FC1, 4th generation model.

FS47FC2.DAT — Seven point failure combinations which produce system failure for FC1, 4th generation model.

FS48FC2.DAT — Eight point failure combinations which produce system failure for FC1, 4th generation model.

Appendix D: Papers and Reports

Whilst the work was being carried out for this thesis a number of papers and research reports were written. In summary, the author made major contributions to one journal paper, seven refereed conference papers and two major government reports as a result of the work described in this thesis.

JOURNAL PAPERS

D.A.Chamberlain and G.J.Bleakley, "CURIO NDT Robot for the Construction Industry", Industrial robot, Vol.21 No.4 1994 pp 29-31.

CONFERENCE PAPERS

Bleakley G.J. Mc Crea A.M., "Design for reliability of a construction robot using Genetic Algorithms", 16th. International Symposium On Automation And Robotics In Construction, 1999, Madrid, Spain.

Bleakley G.J., Edney, R.C & Chamberlain D.A.,"The Use Of DFD And ACP Diagrams For Mechatronics Systems Design", International Conference On Engineering Design, ICED 97 Tampere, August 19-21, 1997, Vol. 3.

Edney, R.C. Bleakley, G.J. "Feasibility design study for robotic handling of Gamma-ray sources for on site radiography", Mechatronics and Machine Vision in practice 1996, Portugal

D.A.Chamberlain, G.J.Bleakley, R.C.Edney, "Robotic Handling of Gamma Ray Sources in Site Radiography of steel Storage Tanks", XIII th. International Symposium of Automation And Robotics in Construction, May 96.

Edney, R.C. Bleakley, G.J, "Feasibility design study for robotic handling of Gamma-ray sources for on site radiography", Mechatronics and Machine Vision in Practice, Vol 1. pp 108-112, 1996.

Chamberlain D.A., Bleakley G.J., Edney R.C, "Robotic handling of Gamma ray sources in site radiography of steel storage tanks", 13th. International Symposium on Automation and Robotics in Construction, 1996, Elsevier.

Bleakley G.J., Chamberlain D.A. (94), "NDT tasking robot for concrete and steel structures", Mechatronics and Machine Vision in Practice, pp 146-154, Toowoomba, Australia, September, 1994.

RESEARCH REPORTS

Chamberlain D., Bleakley G.,(98) "Design methodology for mechatronic systems", Report for the DoE, City University, Construction robotics unit, 1998.

Chamberlain D.A., Bleakley G.J., Edney R.C, "HSE research project 3262/R5925: Remote handling of sources for radiographic inspection", Construction Robotics Unit, City University, November 95, published HSE

PAPERS IN PREPARATION

Bleakley G.J. "Combining computerised failure analysis with QFD",