EGEM

électronique – génie électrique – microsystèmes

Commande des systèmes électriques

perspectives technologiques

sous la direction de

Luc Loron

537-25.1

Hermes Lavoisier

Table des matières

Avant-propos 17
Luc LORON

Chapitre 1. Etat de l'art et spécification des besoins 21
Luc LORON

1.1. Etat de l'art des commandes industrielles de systèmes électriques 21
1.1.1. Evolution de la commande des systèmes industriels 23
1.1.2. Evolution des entraînements à vitesse variable 24
1.1.3. Principales caractéristiques d'un variateur industriel 27
1.1.4. Bilan des entraînements à vitesse variable 30
1.2. Structures de commande types 33
1.2.1. Commande d'axe et asservissement de position 33
1.2.2. Variateur de vitesse 36
1.3. Spécification des besoins 39
1.3.1. Structure cascade 39
1.3.2. Correction PID 40
1.3.3. Génération de trajectoire et anticipation 43
1.3.4. Commande par retour d'état et structure RST 45
1.3.5. Estimateurs et observateurs associés à un correcteur PID 47
1.3.6. Autres approches de commandes 47
1.4. Implantation numérique des commandes 48
1.4.1. Quantification 48
1.4.2. Discrétisation temporelle 49
1.5. Conclusion 50
1.6. Bibliographie 51

Chapitre 2. Conception conjointe matériel-logiciel 53
Jean-Paul CALVEZ et Olivier PASQUIER

2.1. Le problème de la conception 53
2.1.1. Quelques termes 54
2.1.2. Le cycle de développement en V 55
2.2. Méthodologie pour la conception des systèmes 58
2.2.1. Les grandes lignes de la méthodologie MCSE 58
2.2.2. Un modèle pour la description des systèmes matériels/logiciels. . 60
2.2.3. Approche fonctionnelle – approche physique 61
2.3. Conception conjointe matériel/logiciel : de quoi s'agit-il ? 62
2.4. Cahier des charges 65
2.4.1. Présentation du problème illustratif 65
2.4.2. Activités pour la rédaction du cahier des charges 66
2.5. Elaboration des spécifications 67
2.5.1. Analyse de l'environnement 68
2.5.2. Délimitation du système 69
2.5.3. Spécifications fonctionnelles du système 71
2.5.4. Synthèse des spécifications fonctionnelles 73
2.5.5. Spécifications non fonctionnelles 75
2.6. Conception fonctionnelle 76
2.6.1. Délimitation du système et première décomposition fonctionnelle 77
2.6.2. Décomposition fonctionnelle pour la commande de chaque axe . 79
2.6.3. Synthèse et vérification de la solution fonctionnelle 81
2.7. Conception architecturale 82
2.7.1. Partitionnement géographique de la solution 83
2.7.2. Interface pour l'observation de chaque moteur d'axe 84
2.7.3. Interface pour la commande vectorielle de chaque moteur d'axe 85
2.7.4. Analyse des contraintes de temps et partitionnement matériel-logiciel 86
2.7.5. Vérification et évaluation des performances 88
2.8. Déduction du prototype 88
2.9. Conclusion 90
2.10. Bibliographie 91

Chapitre 3. Processeurs pour le temps réel 93
Eduardo MENDES

3.1. Introduction 93
3.2. Principales architectures 94
3.3. Prévisibilité du temps de calcul 101
3.4. Types de mémoires 102

3.5. Périphériques 103
3.5.1. Ports série et parallèle 104
3.5.2. Port hôte 104
3.5.3. DMA 104
3.5.4. Timers 105
3.5.5. Périphériques spécifiques 106
3.5.6. Périphériques pour la mise au point 107
3.6. Gestion des événements 108
3.7. Sûreté de fonctionnement 110
3.8. Représentation des nombres 111
3.9. Précision des calculs en virgule fixe 113
3.10. Familles de processeurs 115
3.10.1. Microcontrôleurs généralistes 116
3.10.2. Microcontrôleurs à noyau DSP 118
3.11. Conclusion 121
3.12. Annexe – représentation des nombres en virgule fixe 122
3.13. Bibliographie 124

Chapitre 4. Nouvelles technologies de commande 125
Hubert RAZIK et Lofti BAGHLI

4.1. Introduction 125
4.2. Logique floue 125
4.2.1. Principe et définitions 125
4.2.2. Opérateurs et normes 127
4.2.3. Inférence 128
4.2.4. Structure d'un régulateur flou 131
4.2.5. Application en commande des machines électriques 137
4.2.6. Conclusion 142
4.3. Réseaux de neurones 143
4.3.1. Principe et définitions 143
4.3.2. Perceptrons multicouches 143
4.3.3. Réseaux de neurones à fonction de base radiale (RBF) 146
4.3.4. Applications 147
4.3.5. Régulation par réseau de neurones 148
4.3.6. Conclusion 153
4.4. Circuits spécialisés numériques et analogiques 153
4.4.1. Composants numériques 153
4.4.2. Composants analogiques 156
4.5. Conclusion 160
4.6. Bibliographie 160

Chapitre 5. Apports des FPGA dans la commande des systèmes électriques 163
Eric MONMASSON et Yves-André CHAPUIS

5.1. Introduction 163
5.2. Description sommaire des FPGA 165
5.2.1. Contexte technologique 165
5.2.2. Architecture générique des FPGA SRAM 165
5.2.3. Introduction au langage VHDL dans la description des FPGA . . 167
5.2.4. Conception et validation des FPGA à partir de langages HDL. . . 168
5.3. Méthodologie de conception de commandes au moyen de FPGA, adéquation algorithme architecture 174
5.3.1. Spécification de l'algorithme à implanter 174
5.3.2. Détermination du chemin de donnée optimale, définition d'une architecture spécifique 176
5.4. Apports et limites des FPGA dans la commande de systèmes électriques 178
5.4.1. Domaine d'exploitation des FPGA 178
5.4.2. Apport des FPGA dans une solution d'implantation logiciel-matériel 180
5.4.3. Apport des FPGA dans une solution d'implantation matérielle . . 181
5.5. Description de plusieurs architectures à base de FPGA – applications . 184
5.5.1. MLI vectorielle 185
5.5.2. Contrôleur du vecteur courant pour machines à courant alternatif 189
5.5.3. DTC pour machines à courant alternatif 193
5.6. Conclusion et perspectives 202
5.7. Bibliographie 203

Chapitre 6. Capteurs et interfaces spécialisés 207
Luc LORON et Jean-Claude LE CLAIRE

6.1. Mesure des grandeurs électriques et magnétiques. 207
6.1.1. Mesure de courant 208
6.1.2. Mesure de tension 212
6.1.3. Mesure de flux 213
6.2. Mesure des grandeurs mécaniques. 214
6.2.1. Mesure de position. 215
6.2.2. Mesure de vitesse 219
6.2.3. Mesure de couple 220
6.3. Mesure de température. 221
6.3.1. Techniques de mesure avec contact. 222
6.3.2. Pyrométrie optique. 224
6.4. Conditionnement des signaux analogiques 225
6.4.1. Câblage d'une liaison analogique 225

6.4.2. Amplification différentielle 227
6.4.3. Filtres analogiques et filtres antirepliement 232
6.5. Fonctions d'interfaçage spécialisées 232
6.5.1. Entrées analogiques 233
6.5.2. Entrées-sorties numériques dédiées 235
6.6. Estimateurs et observateurs 239
6.6.1. Estimateurs 239
6.6.2. Estimateurs de vitesse utilisant une mesure de position numérique 239
6.6.3. Reconstructeurs d'état 244
6.7. Conclusion 247
6.8. Bibliographie 247

Chapitre 7. Les réseaux de terrain 251
Jean-Pierre THOMESSE

7.1. Rôles et fonctions des réseaux de terrain 251
7.1.1. Origine 251
7.1.2. Applications aux machines électriques 252
7.1.3. Rôle des réseaux de terrain 253
7.1.4. Architecture des réseaux de terrain 253
7.2. Définitions 255
7.2.1. Topologie 255
7.2.2. Débit binaire 255
7.2.3. Moment élémentaire, vitesse de modulation, valence 256
7.2.4. Délai de propagation 256
7.2.5. Délai de transmission 257
7.2.6. Temps de transfert sur voie physique 257
7.2.7. Longueur de canal 258
7.3. *Medium Access Control* 258
7.3.1. Principes généraux 259
7.3.2. Gestion par compétition 260
7.3.3. Gestion par multiplexage temporel synchrone 261
7.3.4. Gestion par droit de parole explicite 262
7.4. Trafics et contraintes de temps 265
7.4.1. Exemple de trafic périodique 265
7.4.2. Trafic périodique 266
7.4.3. Trafic apériodique 266
7.4.4. Techniques d'adressage 267
7.4.5. Protocoles MAC et contraintes de temps 268
7.5. Liaison de données 269
7.6. Couche application 272
7.6.1. Introduction 272

7.6.2. Concepts 273
7.7. Exemples de réseaux de terrain. 276
7.7.1. Interbus (http://www.interbusclub.com) 276
7.7.2. Profibus DP (http://www.profibus.com) 277
7.7.3. ASI (http://www.as-interface.com) 279
7.7.4. SERCOS (http://www.sercos.com) 279
7.7.5. CAN (http://www.can-cia.com) 281
7.7.6. WorldFIP (http://www.worldfip.org) 282
7.8. Conclusion 285
7.9. Bibliographie 286

Chapitre 8. Langages et développement logiciel 289
François AUGER et Mohamed SHAWKY

8.1. Introduction. 289
8.2. Existence et conception d'un logiciel embarqué. 290
8.2.1. Le cycle de vie d'un logiciel 290
8.2.2. Le développement d'un logiciel 293
8.3. Principes généraux de la programmation embarquée 296
8.3.1. La modularité. 296
8.3.2. Le contrôle du flux. 297
8.3.3. Le niveau d'abstraction 299
8.3.4. La détection et la gestion des erreurs. 299
8.3.5. L'optimisation du code 301
8.3.6. L'adéquation algorithme-architecture 303
8.4. Langages de programmation usuels 304
8.4.1. Les critères de choix. 304
8.4.2. L'assembleur 305
8.4.3. Les langages C et C++ 306
8.4.4. La technologie Java 308
8.5. Conclusion 313
8.6. Bibliographie 314

Chapitre 9. Générateurs automatiques de code et outils de haut niveau . . 317
Régis MEURET

9.1. Introduction aux générateurs de code 317
9.1.1. Principe d'un générateur de code 317
9.1.2. Intérêts et besoins 318
9.2. Commande de machine tournante 320
9.2.1. Décomposition de la commande. 320
9.2.2. Langages de description et générateurs de code pour la réalisation du convertisseur. 321
9.3. ISaGRAF : générateur de code pour les fonctions d'automatisme 322

9.4. RTW : générateur de code pour les fonctions d'automatique (environnement Matlab/Simulink) 324
9.4.1. Environnement Matlab-Simulink 324
9.4.2. *Real-Time Workshop* (RTW). 325
9.5. Architecture et fonctionnement interne de RTW 328
9.5.1. Procédure de génération d'un programme 329
9.5.2. *Target Language Compiler* (TLC) 330
9.6. VISUAL HDL générateur de code VHDL pour FPGA 332
9.7. Exemple d'une réalisation de commande de machine 334
9.7.1. Réalisation. 334
9.7.2. Organisation logicielle 337
9.7.3. Mise au point du programme. 339
9.8. Conclusion 340
9.9. Bibliographie 340

Chapitre 10. Noyaux d'exécutif temps réel pour la commande de systèmes 341
Yvon TRINQUET

10.1. Introduction aux systèmes temps réel 341
10.1.1. Systèmes réactifs 342
10.1.2. Particularités des systèmes temps réel 343
10.1.3. Domaines d'application des systèmes temps réel. 345
10.2. Implantation d'une application temps réel 346
10.2.1. Notion d'architecture opérationnelle 346
10.2.2. Approches synchrone et asynchrone 347
10.2.3. Différentes structures d'exécutif temps réel 350
10.2.4. Classification des exécutifs temps réel 352
10.3. Services et objets du noyau d'un exécutif temps réel 353
10.3.1. Historique de la proposition OSEK/VDX 354
10.3.2. La gestion des tâches. 355
10.3.3. La synchronisation des tâches 357
10.3.4. Partage de ressources et exclusion mutuelle 360
10.3.5. Les objets alarmes et compteurs 362
10.3.6. La communication 363
10.3.7. Prise en compte des interruptions 365
10.3.8. Les classes de conformité 366
10.4. Exemple d'application 367
10.5. Conclusion 369
10.6. Bibliographie. 370

Chapitre 11. Mise au point et validation des systèmes commandés en temps réel 373
Jean-Claude HAPIOT

11.1. Introduction 373
11.2. Spécificités de la commande numérique en temps réel 374
11.2.1. Partie opérative ou procédé 375
11.2.2. Capteurs 376
11.2.3. Actionneurs 377
11.2.4. Partie commande 377
11.2.5. Protections rapides 380
11.2.6. Premiers constats 381
11.3. Outils d'aide à la mise au point 381
11.3.1. Critères de choix 382
11.3.2. Présentation des principaux outils d'aide à la mise au point . . . 384
11.3.3. Conclusion 391
11.4. Développement de la partie commande 392
11.4.1. Développement du matériel 393
11.4.2. Développement du logiciel 396
11.5. Apports de la simulation 404
11.5.1. Simulation « macroscopique » 405
11.5.2. Cosimulation 406
11.5.3. Emulation (du procédé) 407
11.6. Conclusion 407
11.7. Bibliographie 408

Index 411

Avant-propos

Les commandes de machines électriques sont des systèmes complexes faisant intervenir des composantes matérielles et logicielles fortement imbriquées. L'évolution rapide des technologies informatiques (processeurs, circuits spécialisés d'interfaçage, circuits logiques programmables, réseaux, langages, outils de développement) est à double tranchant. En effet, si elle met à la disposition des concepteurs et des développeurs des composants et des outils de plus en plus puissants, ceux-ci sont également de plus en plus complexes et demandent des formations de plus en plus lourdes, même si la convivialité et l'ergonomie des outils progressent. Ainsi, dans un contexte où l'obsolescence est rapide, il devient très difficile de faire les meilleurs choix technologiques.

L'objectif de cet ouvrage, qui s'adresse principalement aux concepteurs et aux développeurs de systèmes de commande de machines électriques (élèves-ingénieurs, étudiant en thèse, ingénieur de R&D...), est d'apporter un éclairage concret sur les principales facettes technologiques de ces systèmes.

Le chapitre 1 « Etat de l'art et spécification des besoins » précise les ressources nécessaires à la réalisation des commandes de machines électriques triphasées. Après avoir présenté un état de l'art des systèmes industriels de variation de vitesse, nous rappellerons les principes fondamentaux applicables à la commande des machines, afin d'évaluer les besoins technologiques de ces commandes, en termes de calcul, d'instrumentation et de traitement de l'information.

Le chapitre 2 « Conception conjointe matériel-logiciel » propose une méthodologie globale de conception destinée à offrir une solution opérationnelle respectant le cahier des charges et basée sur une association judicieuse de matériel et de logiciel. Une telle approche est indispensable, afin de satisfaire au mieux les inévitables compromis de coût, de performances et de fiabilité, ainsi que pour tirer le meilleur parti des circuits logiques programmables (FPGA).

Les cinq chapitres suivants concernent plus spécifiquement les aspects matériels, dans le cadre des applications de commande des systèmes électriques :

– le chapitre 3 « Processeurs pour le temps réel » analyse et compare les principales familles de processeurs (microprocesseurs, processeurs de signaux, microcontrôleurs) en fonction de leur structure interne et de leur capacité d'interfaçage ;

– le chapitre 4 « Nouvelles technologies de commande » débute par le rappel des principes fondamentaux de la logique floue et des réseaux de neurones, avant de présenter quelques composants numériques et analogiques destinés à l'implantation de telles commandes ;

– le chapitre 5 « Apports des FPGA dans la commande des systèmes électriques » décrit l'architecture générique de ces composants et présente leurs principaux outils de développement. Il discute ensuite des apports et des limites des FPGA dans le domaine de la commande et présente plusieurs études de cas ;

– le chapitre 6 « Capteurs et interfaces spécialisés » rappelle le principe et les principales propriétés des capteurs et des circuits d'interfaçage couramment utilisés en commande des machines. Il met l'accent sur les aspects pratiques (problèmes de mise en œuvre, conditionnement des signaux…) et sur les critères de choix de ces éléments ;

– le chapitre 7 « Réseaux de terrain » introduit les concepts de base des réseaux de terrain. Un réseau de terrain est un système de communication qui possède des couches de service et de protocoles. Il offre une qualité de service qui s'exprime selon des critères dont il s'agit de préciser les plus pertinents.

Les quatre derniers chapitres considèrent principalement les outils et les méthodologies de développement logiciel :

– le chapitre 8 « Langages et développement logiciel » dresse un panorama des langages adaptés aux systèmes commandés (assembleur, C, C++, Java), ainsi que leurs outils de développement logiciel, en examinant leur adéquation aux besoins de la commande des machines ;

– le chapitre 9 « Générateurs automatiques de code et outils de haut niveau » montre comment les générateurs automatiques de code, qui visent tout d'abord à accélérer le développement logiciel en facilitant le travail du développeur, réduisent également les risques d'erreur logicielle et standardisent l'écriture des programmes ; ce qui facilite leur maintenance ;

– l'objectif du chapitre 10 « Noyaux d'exécutif temps réel pour la commande de systèmes » est de montrer l'intérêt d'un noyau temps réel aux concepteurs de commande de machine. Pour cela, il s'appuie sur noyau adapté aux systèmes embarqués : OSEK/VDX ;

– enfin, le chapitre 11 « Mise au point et validation des systèmes commandés en temps réel » aborde le problème délicat de la correction des erreurs de

programmation dans un système asservi en fonctionnement : le procédé et la commande étant alors en interaction dynamique. Après avoir rappelé le principe et les fonctionnalités des multiples outils disponibles, il propose une démarche rationnelle de développement logiciel et de recherche des erreurs.

Luc LORON