

TABLE DES MATIERES

AVANT-PROPOS	7
1. INTRODUCTION	9
1.1. Qu'est ce qu'un système de commande en temps réel	10
1.1.1. Les tâches	11
1.1.2. Partie commande et partie données	13
1.2. Une méthode rigoureuse de conception	14
1.2.1. La vie d'un système	15
1.2.2. Comment valider les spécifications fonctionnelles ?	15
1.2.3. Comment choisir un outil de spécification fonctionnelle ?	17
1.3. Le parallélisme	19
1.3.1. Qu'est ce que le parallélisme ?	19
1.3.2. L'explosion combinatoire du nombre des états	19
1.3.3. La possibilité de blocage	23
2. LE RESEAU DE PETRI	27
2.1. Introduction	28
2.2. Présentation informelle	28
2.2.1. Notion de réceptivité	29
2.2.2. Etat partiel, notion de place	29
2.2.3. Places contenant plus d'un jeton, poids associé à un arc	31
2.3. Définition du réseau de Petri	33
2.3.1. Définition du graphe	33
2.3.2. Règles d'évolution	33
2.4. Exemple	36

* 2.5. Schéma à réseau de Petri	39
2.5.1. Le graphe de données	40
2.5.2. Le graphe de commande	40
2.5.3. L'interprétation	41
2.6. Exemple de schéma à réseau de Petri	41
2.7. Conclusion	44
 3. DESCRIPTION D'UN SYSTEME	 45
3.1. Principales difficultés rencontrées et limitations des réseaux de Petri	 46
3.1.1. Test d'une variable	46
3.1.2. Choix du niveau de description pour les systèmes de grande taille	 49
3.2. Cas des automatismes logiques	50
3.2.1. Prise en compte des arrêts d'urgence	51
3.2.2. Représentation d'un ensemble d'automatismes logiques communicants	 53
3.2.3. Prise en compte du temps	55
3.2.4. Association de tâches aux transitions, tirs non instantanés	 56
3.3. Description fonctionnelle par affinements. Exemple d'un modèle matériel	 60
3.3.1. Définition d'un bloc	60
3.3.2. Remplacement d'une transition par un bloc	61
3.3.3. Exemple	63
3.4. Représentation de l'enchaînement des tâches dans un système multiprogrammé	 67
3.4.1. Principe	68
3.4.2. Représentation d'une tâche interruptible par une autre	 69
3.4.3. Exemple	70
3.5. Mécanismes d'allocation de ressources	73
3.5.1. Principe	73
3.5.2. Exemple	75
3.5.3. Utilisation de variables de synchronisation	76
3.6. Conclusion : diverses manières d'utiliser les réseaux de Petri	 79

4. VALIDATION D'UN SYSTEME	81
4.1. Introduction	82
4.2. Analyse de la partie "commande"	82
4.2.1. Les "bonnes " propriétés	83
4.2.2. Exemple de propriétés spécifiques	89
4.2.3. Méthode d'analyse de base	89
4.2.4. Analyse par réduction	95
4.2.5. Exemple	96
4.2.6. Analyse par affinements	97
4.2.7. Exemple	99
4.2.8. Analyse structurelle	99
4.2.9. Exemple	101
4.2.10. Conclusion	103
4.3. Analyse de l'ensemble : partie commande et partie données	104
4.3.1. Introduction	104
4.3.2. Schéma à réseau de Petri déterministe et déterminé	105
4.3.3. Cas de l'utilisation de variables auxiliaires de synchronisation	109
5. REALISATIONS	113
Remarque préalable	114
5.1. Réalisations matérielles	115
5.1.1. Réalisations par modification du modèle	115
5.1.1.1. Par obtention du graphe des marquages accessibles	115
5.1.1.2. Par décomposition du réseau de Petri	118
5.1.2. Réalisations directes	122
5.1.2.1. Réalisations synchrones	122
5.1.2.2. Réalisations asynchrones	124
5.1.2.3. Réalisation matricielle	129
5.1.2.4. Initialisation	134
5.1.3. Conclusion	135

5.2. Réalisations logicielles	136
5.2.1. Réalisations logicielles par Automate Programmable	136
5.2.1.1. Utilisation d'un Automate Programmable basé logigramme	138
5.2.1.2. Automate programmable par réseau de Petri	140
5.2.1.3. Conclusion	150
5.2.2. Implémentation logicielle d'algorithmes de synchronisation	151
5.2.2.1. Utilité d'un moniteur temps réel	151
5.2.2.2. Primitives de synchronisation	152
5.2.2.3. Synchronisation des tâches	153
5.2.2.4. Précautions et limites	157
5.2.2.5. Synchronisation centralisée	159
5.2.2.6. Prise en compte de variables externes	161
CONCLUSION	165
BIBLIOGRAPHIE	169
ANNEXE 1	175
ANNEXE 2	179