

I. Définition :

1. Système

Notion très large: système d'exploitation, réparti, de gestion de base de données, temps réel, critique, embarqué, logique, de connaissances, expert, ...

Point commun :

- ensemble composé d'éléments qui interagissent
- pour remplir une fonction particulière
- dans un environnement donné
- en respectant un ensemble de contraintes

2. Système réactif (SR)

Terme réactivité introduit par D. Harel et A. Pnueli en 1985

- un SR est un système qui réagit de façon continue avec son environnement
- un SR fait intervenir
 - ➔ un calculateur (système contrôlant)
 - ➔ un environnement depuis lequel il reçoit des informations et sur qui il exécute ses commandes (système contrôlé)
 - ➔ des appareils interconnexion entre les deux composants.

3. Système réactif temps réel (SR_TR)

Un SR_TR est un SR soumis à de fortes contraintes de temps imposées par son environnement :

- exactitude logique (logical correctness) : sorties adéquates en fonction des entrées.
- Exactitude temporelle (timeliness) : respect des contraintes temporelles

échéance : délai dans lequel un SR_TR doit produire un résultat

II. Classification :

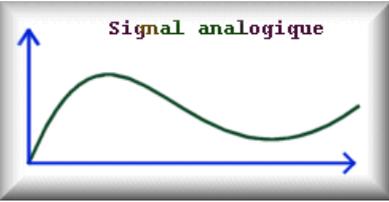
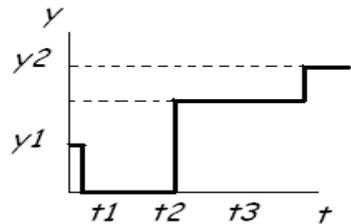
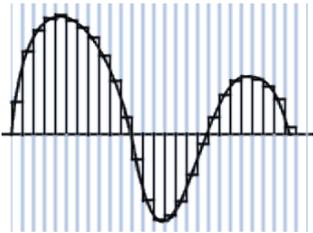
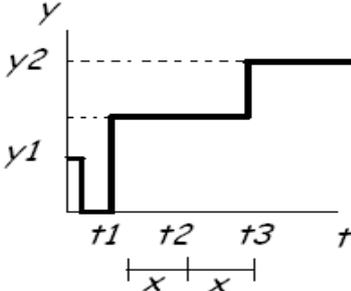
1. Systèmes continus (SC) / Systèmes à événements discrets (SED)

La classification SC / SED est fonction de :

- la nature des informations traitées
- le type d'observation de ces informations

Il y a 4 types d'informations :

- ★ Analogique (SC)
- ★ Echantillonnée (SC)
- ★ Numériques / logiques asynchrones et synchrones (SED)

Type d'observation \ Nature de l'info	Continue	Quantifiée
Continue		
Discrete		

2. Système à événements discrets (SED)

Dynamique d'un SC : décrite par des variables continues du temps

Dynamique d'un SED :

- ★ l'état du système ne change qu'à certains instants (état = mémoire)
- ★ est une fonction d'une séquence d'événements d'entrées.

Définition d'un SED : Système à espace d'états discrets dont les transitions entre états sont associées à l'occurrence d'événements discrets.

Phases d'étude d'un système :

- ★ Relations du système avec son environnement (E/S)
- ★ Typologie des E/S (numérique, analogique, logique)
- ★ Modèles adaptés aux types d'E/S
- ★ Outils de modélisation qui caractérisent chaque modèle
- ★ Représentation du système par les outils retenus.

3. SED combinatoires / séquentiels

- Systèmes combinatoire : les sorties ne dépendent que des entrées.
- Systèmes séquentiels : les sorties dépendent des entrées et des états antérieurs du système.

$$X_{t+1} = f(X_t, E_t)$$

E_t : vecteur d'entrées ; X_t : vecteur d'états ;

$$Y_t = g(X_t, [E_t])$$

Y_t : vecteur de sorties ; f et g : fonctions combinatoires

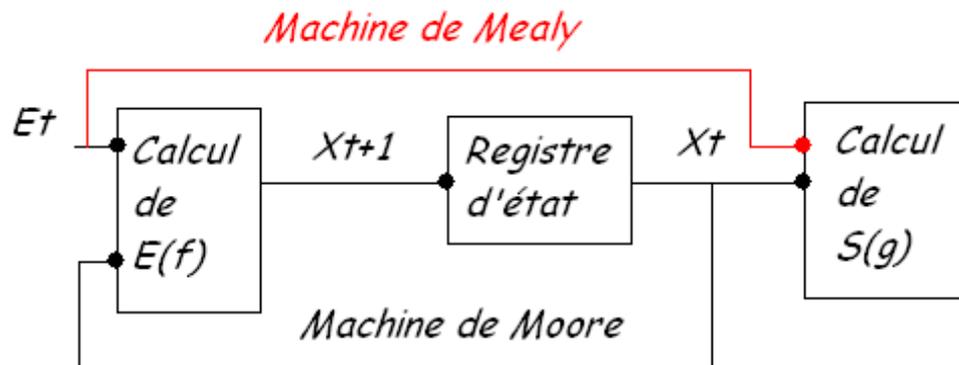
Machine de Moore : $X_{t+1} = f(X_t, E_t)$ et $Y_t = g(X_t)$

Machine de Mealy : $X_{t+1} = f(X_t, E_t)$ et $Y_t = g(X_t, E_t)$

Les fonctions f et g se représentent par une machine d'état (F.S.M.: Finite State Machine) et des diagrammes de transition :

★ une machine d'état se décompose en trois blocs :

- Calcul de l'état futur (fonction f)
- Registre d'état (mémoire de la machine)
- Calcul des sorties (fonction g)



Un diagramme de transition comprend :

- ★ des sommets qui représentent les états
- ★ des flèches qui représentent les transitions entre états
- Machine de Moore : actions associées aux états
- Machine de Mealy : actions associées aux transitions

4. SED asynchrones/synchrones

- Systèmes asynchrones : tout changement sur l'environnement est immédiatement signalé au système et suppose un traitement immédiat.
- Systèmes synchrones : les informations qui arrivent au système sont planifiées dans le temps et sont traitées immédiatement.
 - ★ une horloge permet de synchroniser les instants d'évolution du système.

Hypothèses (systèmes. Synchrones) :

- ★ les hypothèses sont produites dans le même instant que les informations d'entrées qui ont provoqué leur déclenchement.
- ★ l'intervalle de discrétisation du temps est infiniment petit par rapport au temps de changement des informations d'entrées.

5. Fonctionnement cyclique des SED

Un SED est censé fonctionner en permanence.

Algorithme standard du cycle de traitement :

```
Module RTControl
(*Déclaration*)
Begin
(*initialisation*)
    LOOP
    (*lecture des entrées*)
    (*traitement*)
    (*envoi des sorties*)
    (*test condition de
EXIT*)
    END (*Loop*)
END RTControl
```

Fonctionnement synchrone : mémorisation de l'état des entrées et sorties pendant un cycle de traitement
(Mémoires Images des E/S)

III. Méthodologie de développement :

Cycle de développement d'un système

Modèle pour la conception des SED

- ★ *exemples d'outils de modélisation statique*
- ★ *exemples d'outils de modélisation cinématique*
- ★ *exemples d'outils de modélisation dynamique*

1 Cycle de développement des SED

L'ensemble des étapes successives qui doivent aboutir à l'utilisation du système, en partant de la définition des besoins :

- ★ **spécification**
- ★ **conception générale**
- ★ **conception détaillée**
- ★ **réalisation**
- ★ **tests**
- ★ **intégration**
- ★ **Recette**

Spécification : traduction des besoins

Conception générale : définition de l'architecture

Conception détaillée : définition séparée des différents modules

Réalisation : création, codage

Tests : tests séparés des différents modules

Intégration : validation de l'ensemble des modules

Recette : implantation, vérification de la conformité.

Présentation typique sous la forme d'un « V » : met en évidence les relations entre les étapes de spécification/conception et les étapes de tests/validation.

Besoins	(conformité)	système en exploitation
Spécifications	(vérification)	recette
Conception générale	(validation)	intégration
Conception détaillée	(tests)	Tests unitaires
REALISATION		

2 Modèles pour la conception des SED

Objectif : recenser les principaux modèles et outils de modélisation utilisés en conception de SED

Terminologie :

- ★ **Méthode :** ensemble de règles à appliquer pour traiter un problème
- ★ **Outil :** moyen proposé pour appliquer la méthode
- ★ **Modélisation :** phase d'établissement d'un ensemble de modèles des objets à concevoir
- ★ **Modèle :** représentation schématique des objets à étudier ou à concevoir destinée à en faciliter l'étude.

Critères de choix des modèles :

- ★ Phase du cycle de développement à traiter
- ★ Point de vue considéré du système
 - **Aspect informationnel :** structuration des données ;
 - **Aspect fonctionnel :** activités qui traitent les données ;
 - **Aspect comportemental :** évolution du système dans le temps.
- ★ Prise en compte du temps
 - **Modèles statiques :** composition du système ou un des états du système ;
 - **Modèles cinématiques :** enchaînement des actions du système ;
 - **Modèles dynamiques :** comportement du système dans le temps.

Exemples d'outils de modélisation statique :

- ★ **Diagrammes Entité/Association**
 - Domaine d'application : Etude de données
 - Notation : 2 symboles (Entité et Relation entre les entités)

Utilisés dans de nombreuses méthodes de conception de systèmes d'information (Merise, UML,...)

- ★ **Diagramme SADT/IDEFO**
 - Domaine d'application : spécification fonctionnelle d'un système.
 - Notation : 2 modèles (Datagrammes et Actigrammes)

Exemples d'outils de modélisation cinématique :

Domaine d'application : aspect chronologique (événements, actions,...)

★ **Chronogrammes** : représentation de séquences d'opérations sur un diagramme des temps

★ **Organigrammes** : représentation de l'enchaînement des opérations séquentielles d'un programme

Exemples d'outils de modélisation dynamique :

Dérivés de la théorie des automates et des graphes

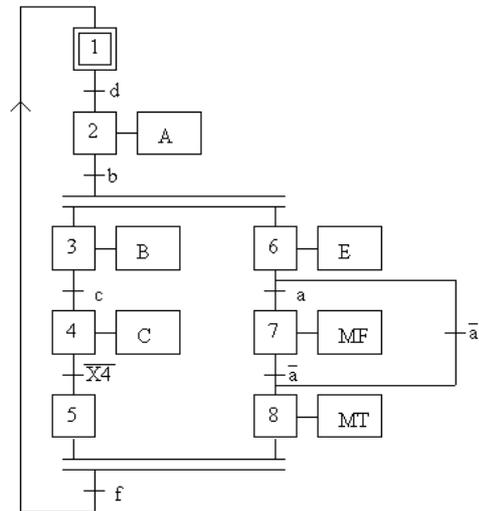
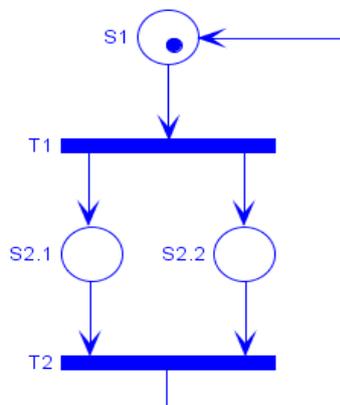
★ **Réseaux de Petri** : Modélisation du comportement des SED

● Application de gestion de flux (modèle numérique)

● Modèle analysable (vérification de propriétés)

★ **Grafcet** : Modélisation du comportement des SED

● Conception du contrôle/commande des systèmes (modèle logique)



★ **Langages synchrones** : Langages développés pour les SR_TR

● Manipulation de signaux :

▪ Flots de données : suite de valeurs qui circulent

▪ Horloges : instants de présence des valeurs

● Langages textuels : ESTEREL, LUSTRE, SIGNAL,...

● Langages graphiques : STATECHARTS, SDL, SyncCharts,...

STATECHARTS :

★ décrit les changements d'état d'un objet en réponse à des événements ;

★ permet une hiérarchisation des états afin d'éviter une explosion combinatoire d'états ;

★ permet de représenter la concurrence des activités ;

★ Concepts :

● Etats

● Transitions

● Actions

● Événements