

Étude des microcontrôleurs

Pierre-Antoine

November 27, 2018

1 Introduction

1.1 Prérequis et rappels

- logique câblée
 - Circuit séquentiels
 - circuit combinatoire
- très rapide
- économique en énergie
- complexe à définir
- figé ou difficilement modifiable
- inadapté aux tâches complexes

1.2 Autre approche : logique programmée

- Le programme est stocké et mémorisé dans le système
- Quantum d'information adopté en général : l'octet : 8bit.
- stocké dans la mémoire morte (ROM) sous la forme d'une série de mots.
- les données manipulées sont rangées en mémoire vive (RAM).

2 **espaces** sont nécessaires :

- Programme
- Données

Ces espaces peuvent être disjoints (ex: PIC), ou communs (PC)

1.3 Organisation interne de petit système

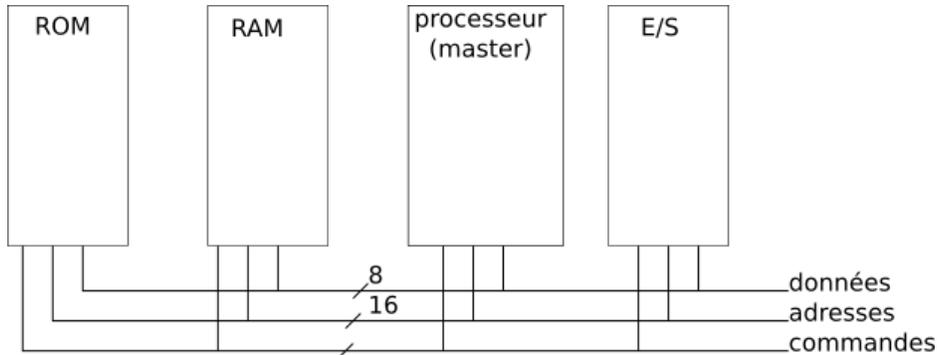
- ROM
- RAM
- Processeur
- Entrée/Sortie

1.3.1 interconnexion des éléments:

Utilisation des buffers 3 états : **images**

ES	E	S
0	0	0
0	1	1
1	0	Z
1	0	Z

avec **Z** état à haute impédance. on a alors une structure de Bus.



La structure de bus est composé deux trois type de bus:

- bus d'adresse : désigne une case menu
- bus de donnée : circuit des données
- bus de commande (contrôle) , pour séquencer les opération d'échange

1.4 Analyse d'un exemple

- identification des éléments
 - U1 : processeur Z80 (CPU) boitier 40 broches
 - U2 : RAM 32 kB
 - U3 : EPROM : Enable Programmable Read Only Memory
 - U4 : PIO :Parallel Input Output

2 Le timer et les séries

Les timers sont nécessaires pour pouvoir mesurer le temps, sans monopoliser le processeur pour ca

2.1 Fonction des timers:

- mesurer la durée d'une impulsion
- compter des évènements (exemple codeur)
- Générer une PWM
- Lancer une fonction périodique

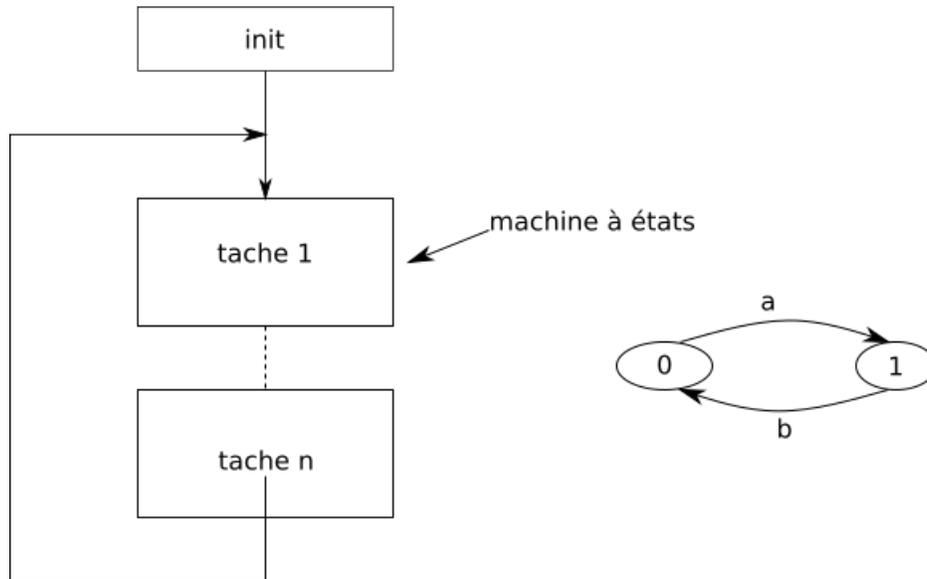
2.2 fonctionnement du timer

Voir slide du poly

2.3 Timers du LPC804

3 Programmation des machines à états (FSM)

3.1 principe général pour μC



L'enjeu est de conserver et de mettre à jour facilement l'état de la machine en fonction des différentes tâches.

4 Appel de sous-programme, interruption et utilisation de la pile par le micro contrôleur.

-> faire cohabiter des tâches différentes:

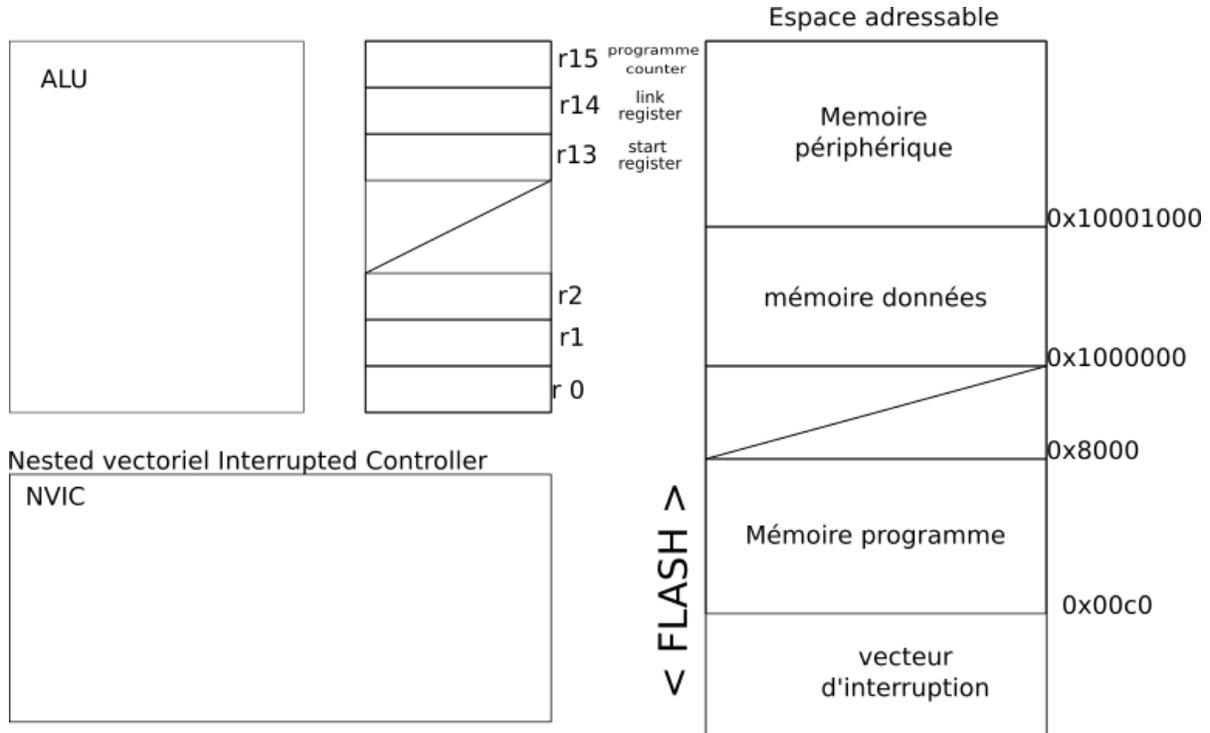
- Longue durée prioritaire (affichage LCD ..)
- Courte périodique et prioritaire (échantillonnage)
- Rares (liaison série ..)

4.1 Fonctionnement des processeur: le cas du Cortex M0+

- faible consommation électrique
- 56 instructions , 16 registre internes
- Contrôleur d'interruption NVIC
- SysTick timer

<http://www.infocenter.arm.com>

4.2 fonctionnement de la CPU : schéma interne.



4.3 les exceptions

- met en pause le programme pour exécuter un programme annexe,
- Exception matérielle : interruption
- 32 exception externe possible pour le cortex M0+ .

*

5 I2C et LPC804

3 protocoles classiques :

- Serial
- I2C
- SPI

Serial	SPI	I2C
Asynchrone	Synchrone	
Pas d'émulation soft ¹ point à point	émulation soft point à point	émulation soft BUS (adresse pour chaque périphérique I2C)

¹nécessité d'une aide physique (timer ...)

5.1 Technologie utilisé pour les bus :

Les sorties doivent être à collecteur ouvert (pas de court circuit). La vitesse du bus est limitée par la commutation de la résistance, moins efficace mais plus simple qu'un circuit à état/haute impédance

5.2 Trame I2C

cf le poly distribué en cours

6 CAN et CNA

Objectif: interfacer des dispositifs analogique et numérique

6.1 Signal analogique / signal numérique

La numérisation d'un signal fait intervenir:

- Un échantillonnage
- Une quantification

attention au théorème de Shannon.

Différents formats numériques audio (codage des données avec ou sans pertes sur plus ou moins de bits)

6.2 Convertisseur Numérique analogique

$\bullet V_A \sqrt{\frac{R}{2}}$ (A) to [R, l=R] ++(2,0) node(B) {•} node[above]{V₂} (B) to [R, l=R] ++(2,0) node(C) {•} node[above]{V₁} (C) to [R, l=R] ++(2,0) node(D) {•} node[above]{V₀}

