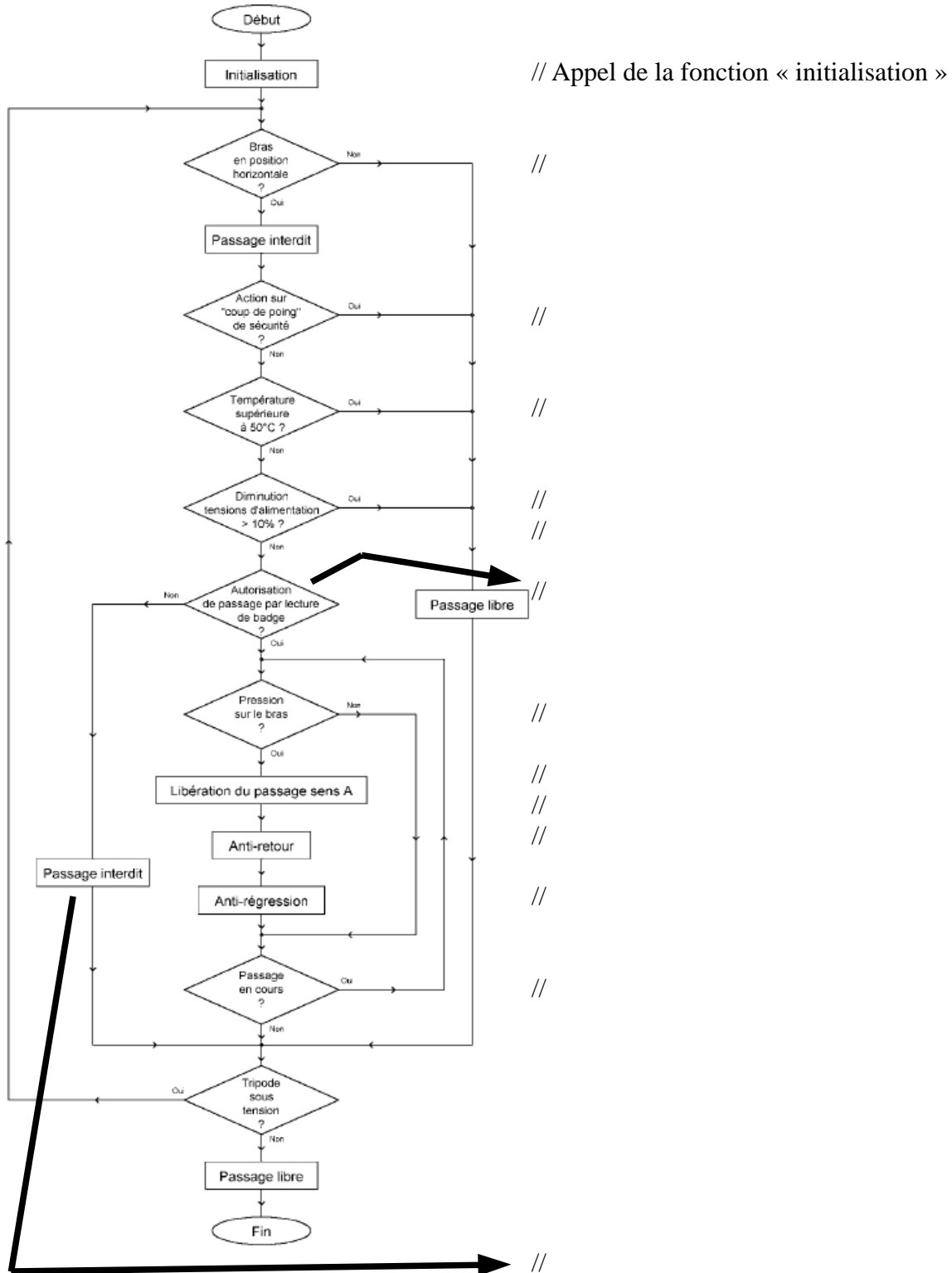


Étude de FS4-5 : GESTION DES SIGNAUX D'ACTION

1. Correspondance entre l'algorithme de fonctionnement (page 57) et l'algorithme du programme principal (page 66)



2. A l'aide de la définition des variables d'entrée du programme, déterminer les informations des capteurs contenues dans la variable STATUS ?

3. A quoi correspond l'information SENS3 ? Quelle est sa valeur lorsque le tripode est au repos ?

4. Etude de la suite d'instructions :

```
VALEUR_STATUS = STATUS;
```

```
VALEUR_STATUS = VALEUR_STATUS & 0x04;
```

A l'aide de cette page Internet : http://membres.lycos.fr/dancel/c/c30_30.htm

déterminer la valeur de VALEUR_STATUS après la deuxième instruction, si STATUS vaut 0xA9 (N.B. : 0x signifie que la valeur qui suit est exprimée en hexadécimal)

5. En s'inspirant de l'algorithme du programme principal, convertir l'algorithme de la fonction « test_lecteur_A » en algorithme.

6. Faire de même pour la fonction « anti_retour ».

7. Faire de même pour la fonction « anti_regression ».

6. Faire de même pour la fonction « passage_interdit ».

Étude de FS4-7 : MÉMORISATION DES PROGRAMMES ET DES DONNÉES

1. Rappeler le rôle de cette fonction.

2. A l'aide de la documentation constructeur sur les composants de cette structure, déterminer le composant qui mémorise le programme et celui qui mémorise les données. Justifier votre réponse en faisant référence au type de mémoire utilisé (RAM, ROM, etc).

3. La mémoire « programme » insérée sur notre tripode est-elle modifiable ? Si oui, comment ?

4. A l'aide des documentations constructeur des mémoires, déterminer la capacité de chacune d'elles exprimée en kilo-octets.

Vérifier par calcul que le nombre de fils d'adresses correspond à cette capacité.

Accès aux mémoires :

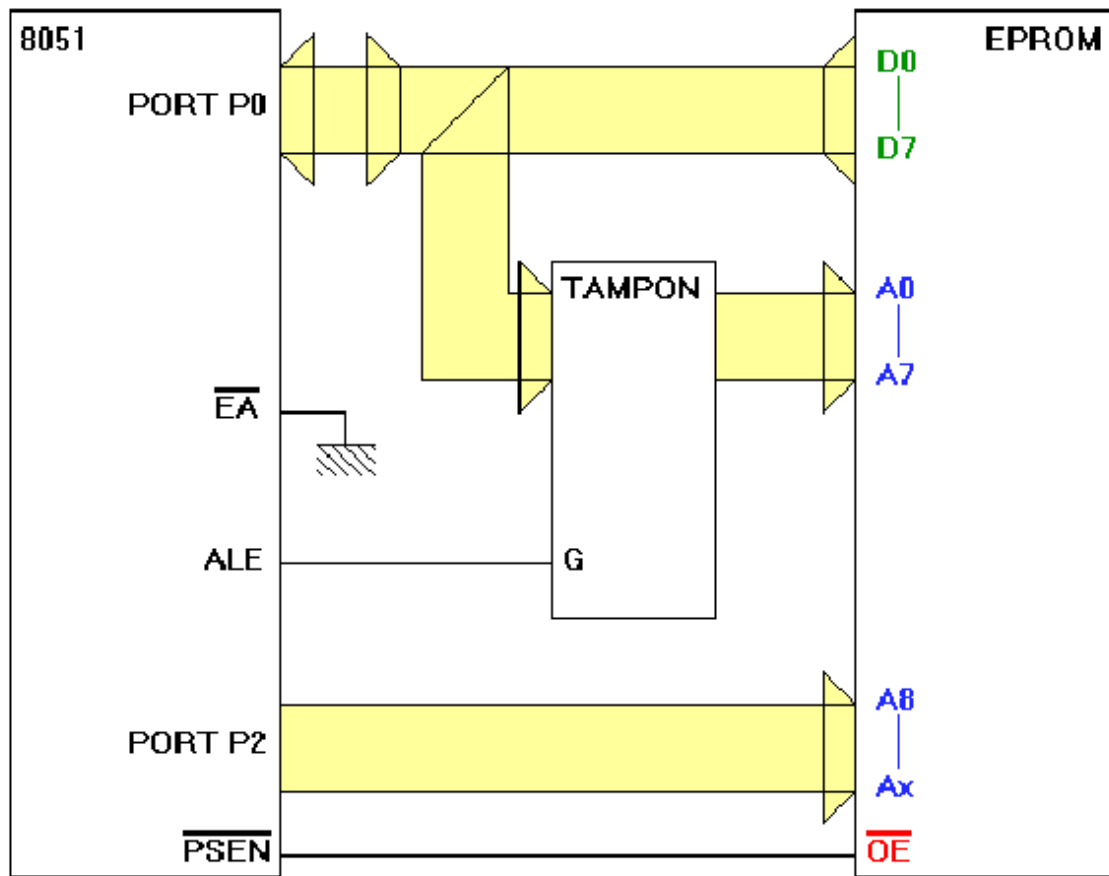
5. Comment le microcontrôleur différencie-t-il la mémoire « programme » de la mémoire « données » ?

6. Sur la page suivante, on trouve un extrait de la page Internet

<http://electroremy.ifrance.com/elec-info-eprom.html>

où l'on explique le principe d'accès du microcontrôleur à la mémoire programme.

Ci dessous je vous indique comment faire fonctionner un microcontrôleur 8051 avec le programme stocké dans une EPROM comme ROM externe.



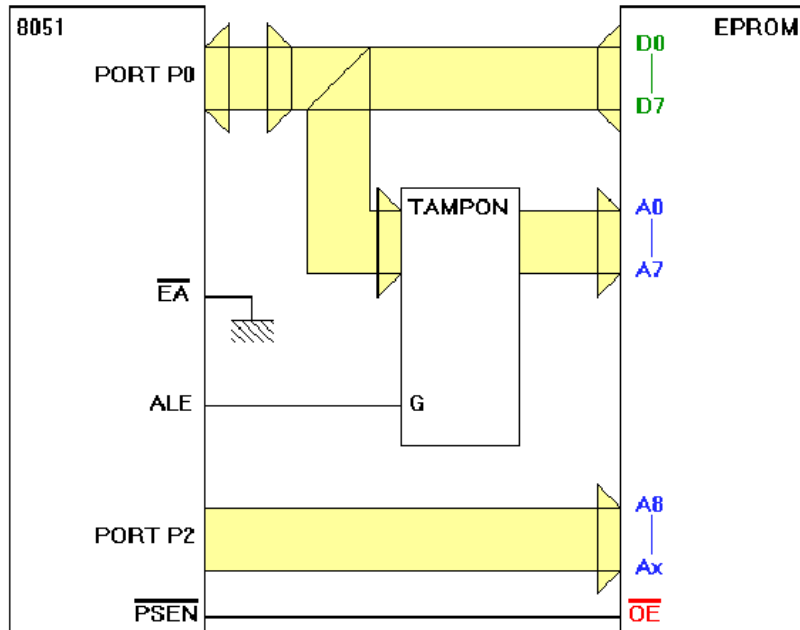
La broche /EA mise à la masse indique au microcontrôleur qu'il doit chercher l'ensemble du programme dans la ROM externe. Si /EA est mis à +5V, le microcontrôleur va seulement considérer la ROM externe comme une extension de mémoire, ce qui veut dire que le début du programme sera cherché dans la ROM interne.

La lecture d'un octet contenu dans la ROM externe se déroule comme suit :

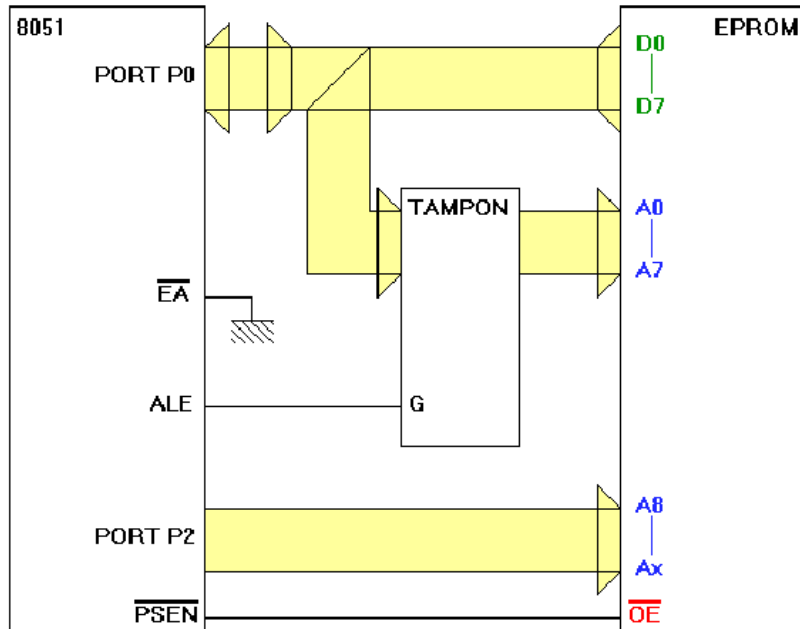
- l'octet bas (l'adresse est sur 16 bits) de l'adresse est émis sur le port P0;
- une impulsion est envoyée sur ALE pour recopier l'état du port P0 dans le tampon;
- l'octet haut de l'adresse est alors émis sur le port P2;
- alors PSEN active les sorties de l'EPROM et l'octet est lu sur P0.

Si on numérote de 1 à 4 les quatre étapes décrites ci-dessus, compléter les quatre schémas suivants en coloriant les connexions actives et en fléchant le sens de circulation des informations. On ajoutera les commentaires ou les légendes nécessaires à la compréhension de ces schémas.

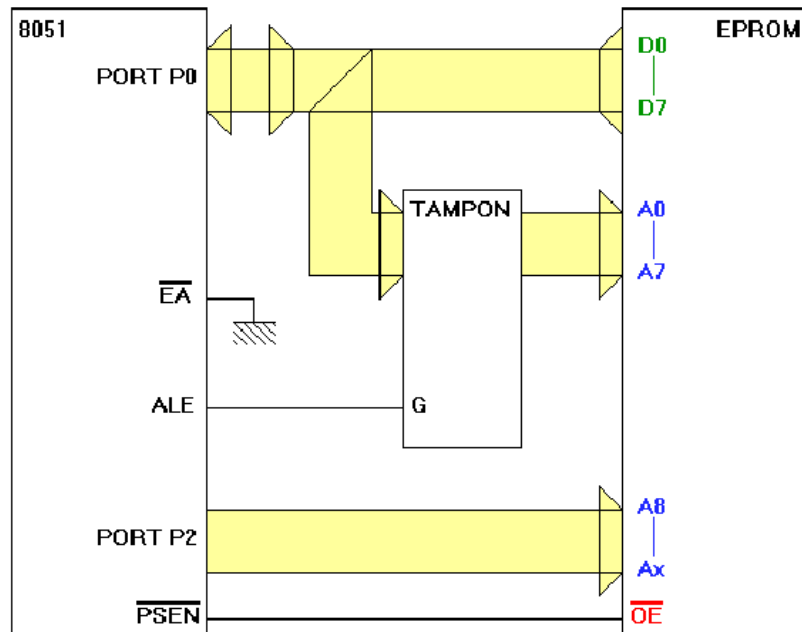
1ère étape : l'octet bas de l'adresse est émis sur le port P0.



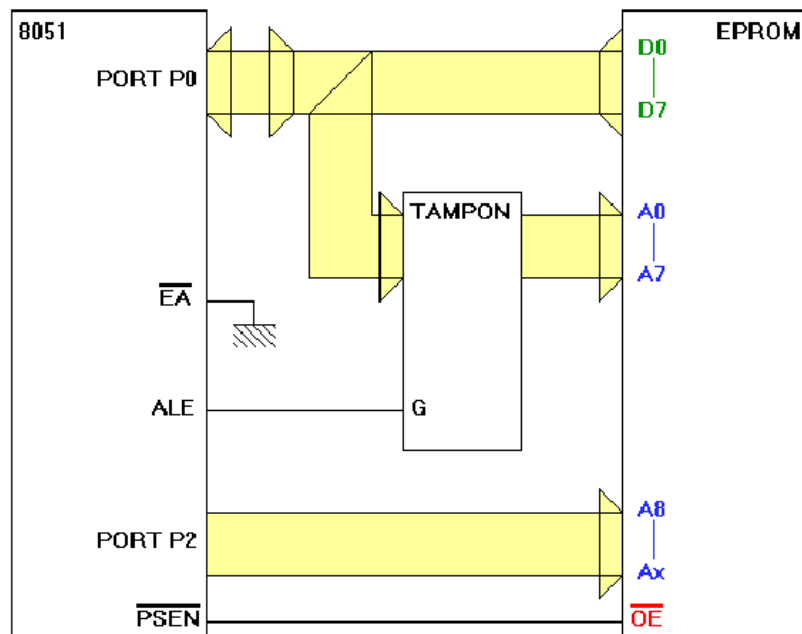
2ème étape : une impulsion est envoyée sur ALE pour recopier l'état du port P0 dans le tampon.



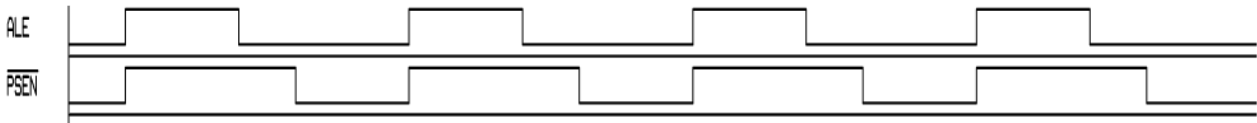
3ème étape : l'octet haut de l'adresse est émis sur le port P2.



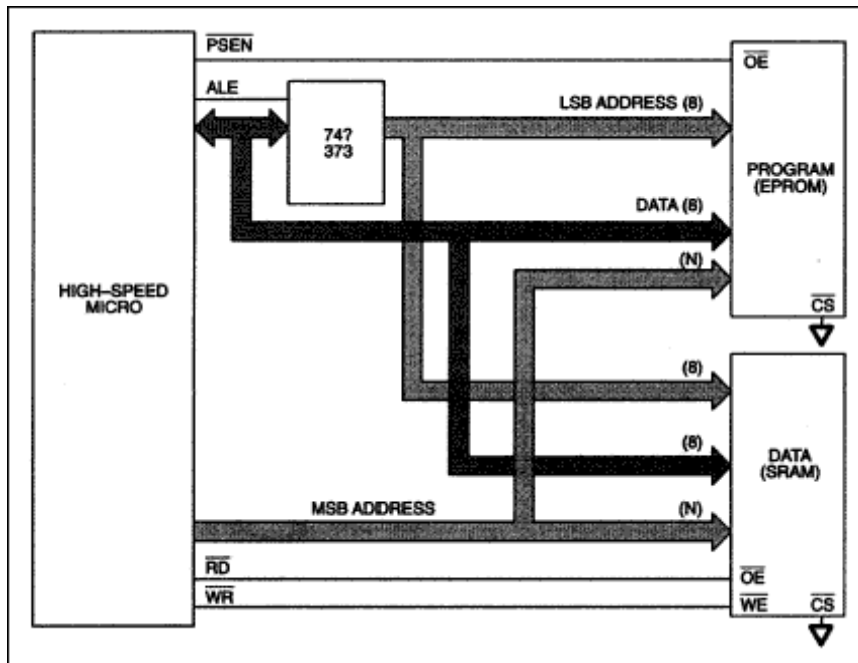
4ème étape : PSEN active les sorties de l'EPROM et l'octet est lu sur P0.



7. Sur les chronogrammes de lecture dans la mémoire « programme », repérer les quatre étapes décrites ci-dessus :



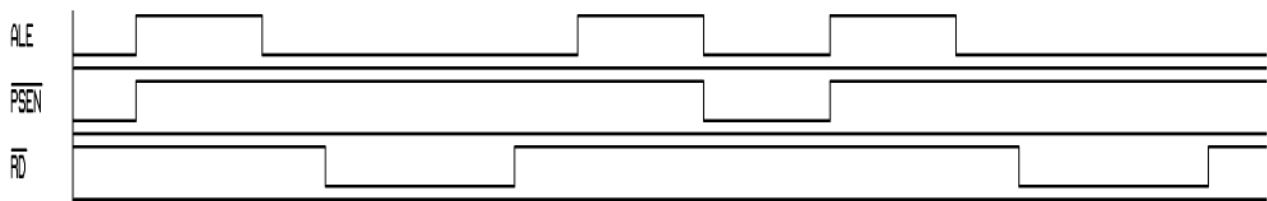
On peut représenter l'architecture « mémoire » complète du microcontrôleur par le schéma suivant :



source : http://www.maxim-ic.com.cn/appnotes.cfm/an_pk/89

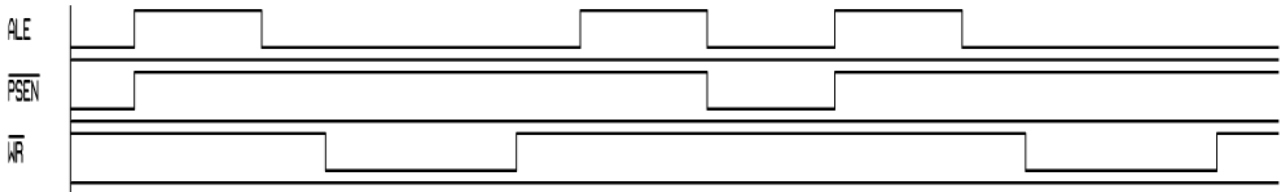
8. Décrire les quatre étapes de **lecture** dans la mémoire de données.

9. Sur les chronogrammes de lecture dans la mémoire « données », repérer ces étapes :



10. Décrire les quatre étapes **d'écriture** dans la mémoire de données.

11. Sur les chronogrammes d'écriture dans la mémoire « données », repérer ces étapes :



Étude de FS4-8 : MÉMORISATION DES PARAMÈTRES

Objectifs :

- Déterminer le rôle et les caractéristiques des différents composants de FS4-8.
- Relever expérimentalement une trame.
- Vérifier que la trame présentée dans le dossier est conforme au protocole I²C
- Vérifier que la trame relevée est conforme au protocole I²C

1. Rappeler le rôle de cette fonction.

2. A l'aide de la documentation technique du circuit intégré U12,

- a) déterminer la taille mémoire de ce composant, exprimée en Ko (kilo-octets) ;
- b) déterminer le rôle des entrées A0, A1 et A2 ;
- c) déterminer le rôle de l'entrée WP ? A quel état logique est-elle fixée ? Pourquoi a-t-on mis cette entrée dans cet état ?

3. Relevé expérimental d'une trame de communication entre la mémoire U12 et le microcontrôleur :

Montage :

La mémoire U12 est montée sur la petite carte I²C. Le microcontrôleur est celui du rack51. Connecter la carte I²C sur le rack51 au bout de deux cartes rallonges. Retirer le cavalier WP.

Placer une sonde (Voie 1) de l'oscilloscope numérique sur le point SCL et l'autre sonde (Voie 2) sur le point SDA. Connecter une masse de l'oscilloscope au 0V de la carte.

Brancher le rack51 à l'ordinateur à l'aide du câble série RS232.

Régler l'oscilloscope comme suit :

- * 0V de SCL sur l'axe médian ; 0V de SDA à un carreau du bas.
- * calibre 2V/division pour les deux voies.
- * calibre 50µs pour la base de temps
- * déclenchement sur un front descendant de SDA

Faire valider votre montage et vos réglages par le professeur.

Brancher le rack51 sur le secteur et le mettre sous tension.

Désactiver le déclenchement et vérifier que les deux signaux sont au niveau logique « 1 » (environ 5V). Si ce n'est pas le cas, appeler le professeur avant de continuer.

Préparation du logiciel :

Copier le dossier I2C qui se trouve sur le serveur Bepelec (dossier ressource/files/tripode/travaux de M. Lerouvillois) dans un dossier de votre poste de travail.

Ouvrir le logiciel Wedit32 ; fermer tout projet qui s'ouvrirait automatiquement.

Dans le menu « projet », ouvrir le projet i2c.prj qui se trouve dans le dossier I2C de votre poste.

Modification du programme :

Nous allons modifier le programme pour pouvoir relever les signaux qui correspondent à l'écriture de la donnée 08h à l'adresse 0Dh (voir page 39 du dossier).

Dans la fenêtre du projet, cliquer sur le petit + pour faire apparaître les fichiers du projet.

Ouvrir le fichier i2c.c

Dans ce programme, repérer la première accolade ouvrante : {

Placer juste après le bout de programme suivant :

```
BUS_LIBRE ();      // Vérification de la libération du bus.
```

```
START ();         // Début de message.
```

```
EMISSION (ECRITURE);
```

```
ACK ();
```

```
EMISSION (VALEUR_ADRESSE[13]);
```

```
ACK ();
```

```
EMISSION (VALEUR_EMISE[13]);
```

```
ACK ();
```

```
STOP ();          // Fin de message
```

Exécution de ce bout de programme :

Cliquer sur l'outil « Build » pour compiler et transférer le programme dans le rack51.

Lorsque le programme réapparaît, il a des petits points verts dans la marge gauche. Cliquer sur le point vert de la deuxième instruction « START(); » afin de placer un point d'arrêt du programme à cet endroit.

Cliquer sur l'outil « Go ». Une trame doit s'afficher sur l'oscilloscope numérique.

Faire vérifier votre trame par le professeur.

Imprimer la totalité de la trame en deux fois. Reconstituer la trame complète sur papier par découpage et collage.

4. Etude théorique du protocole I²C

Objectif : Etudier les échanges de données entre la mémoire U12 et le microcontrôleur.

a) Historique

Le bus I²C (Inter Integrated circuits Communication) a été développé et mis au point par Philips au début des années 80 afin d'assurer les échanges de données entre les circuits d'une même carte pour le matériel grand public, donc d'en réduire les coûts de fabrication.

Le bus I²C se décline en 3 versions :

Version	Standard	Fast	High-speed
Adressage	7 bits + R/W	9 bits + R/W	9 bits + R/W
Fréquence d'horloge max	100 KHz	400 KHz	3,4 MHz

A l'aide du document constructeur de la mémoire, déterminer la version utilisée par ce composant.

b) A ce jour, un grand nombre de fondeurs se sont portés acquéreurs de la licence I²C, et de ce fait, la liste des circuits disponibles possédant une interface I²C est très importante :

- Microcontrôleurs,
- Mémoires RAM, EEPROM,
- Horloges temps réel,
- Circuits audio (contrôle de volume, equalizer, ...),
- CAN et CNA,
- Ports E/S, etc.

Dans la liste ci-dessus, quels sont les composants du tripode concernés par le bus I²C ?

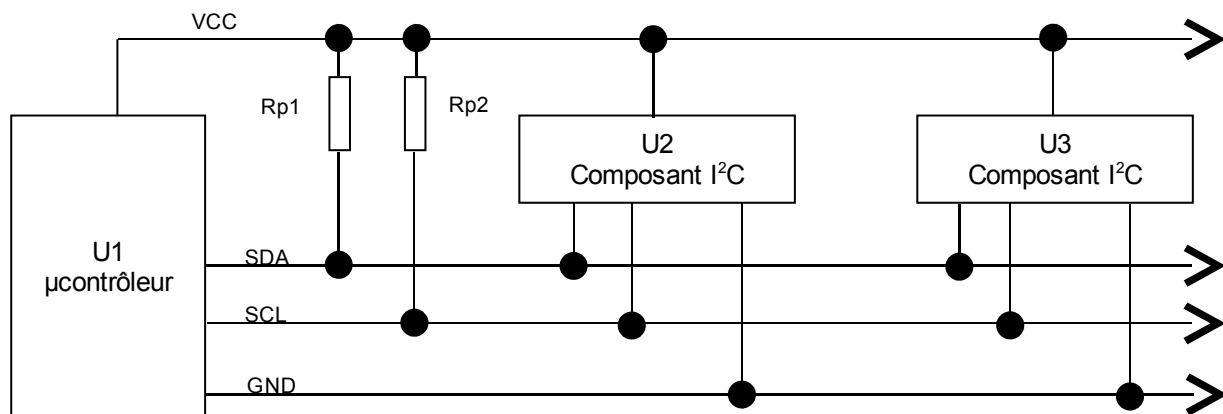
c) Architecture du bus I²C

Seulement 3 fils sont nécessaires pour la mise en œuvre du bus I²C :

SDA : fil de données (Serial Data)

SCL : fil d'horloge (Serial Clock)

GND : référence des potentiels.



Les lignes SDA et SCL sont de type collecteur ouvert, d'où la présence des résistances Rp1 et Rp2. A quelles résistances correspondent-elles dans le tripode ?

d) Terminologie utilisée

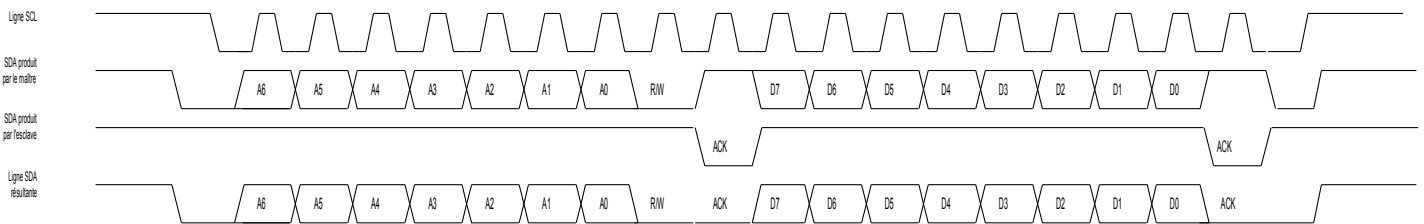
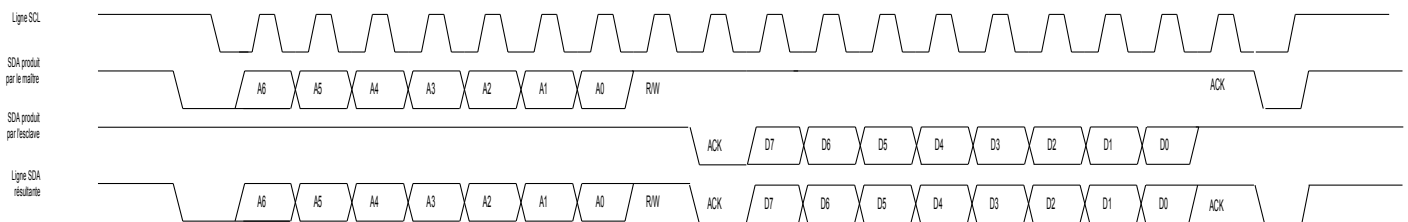
Les échanges de données sur le bus se font entre un maître et un esclave.

Le maître : Seul le maître peut prendre la parole sur le bus. Il produit le signal d'horloge et gère le protocole d'échange, le plus souvent grâce à une interface intégrée.

Quel composant joue le rôle de maître dans le tripode ?

L'esclave : Lorsqu'il est adressé (interrogé), il répond au maître.

Quel composant joue le rôle d'esclave dans le tripode ?

e) Règles de transfert des informationsTrame I²C en mode écritureTrame I²C en mode lecture

Les données sont transmises sur un format de 8 bits avec le MSB en tête.
Colorier en bleu le MSB d'une donnée sur une trame ci-dessus.

Une impulsion d'horloge sur SCL est générée à chaque fois qu'un bit de donnée est transféré.
Le changement d'état de la ligne SDA ne peut se faire que si la ligne SCL est au NL0 (niveau logique 0).

Sur une trame ci-dessus, repasser en rouge un changement d'état de la donnée et le niveau logique de SCL correspondant.

Une donnée est réputée valide lorsque la ligne SDA est stable alors que la ligne SCL est au NL1 (niveau logique 1).

La prise en compte de la donnée se fait sur le front descendant de l'horloge SCL.

Sur une trame ci-dessus, repasser en vert un bit de donnée valide et le front de SCL qui permet de la prendre en compte.

f) Le protocole de communication

Le protocole de communication peut se décomposer selon la trame de base suivante :

- Prise de parole sur le bus
- Envoi de la condition de départ de la communication
- Spécification du mode lecture ou écriture
- Adressage de l'esclave
- Envoi d'un acquittement de l'esclave vers le maître
- Envoi ou réception de la donnée
- Acquittement de fin de communication
- Envoi d'une condition d'arrêt de la communication

La prise de parole :

Avant de débiter une communication, le maître doit s'assurer que les lignes SDA et SCL

sont toutes les deux au repos, soit au NL1, pendant une durée de $4,7\mu\text{S}$.

Sur une trame ci-dessus, repérer entre pointillés une phase de repos.

La condition de départ :

Elle est obligatoire, et se traduit par un passage au NL0 de la ligne SDA , suivi par un passage au NL0 de la ligne SCL.

Il doit s'écouler au moins $4,7\mu\text{S}$ entre le changement d'état de la ligne SDA et celui de SCL.

Sur une trame ci-dessus, repérer entre pointillés une condition de départ.

Adressage de l'esclave:

En mode standard, l'adresse de l'esclave est transmise sur 7 bits.

Sur une trame ci-dessus, repérer entre pointillés l'adresse de l'esclave.

Spécification du mode :

Le niveau logique de ce que l'on pourrait appeler le 8^{ème} bit d'adresse va spécifier si le composant adressé va recevoir une donnée (écriture) ou doit retourner une donnée (lecture).

Sur une trame ci-dessus, repérer entre pointillés le bit de lecture/écriture.

Acquittement d'adresse:

Cette procédure (Acknowledgement) est effectuée par l'esclave pour indiquer au maître qu'il existe bien un composant à l'adresse envoyée.

Le maître maintient la ligne SDA au NL1 alors que l'esclave la force au NL0. Ce bit d'acquittement (ACK) est validé par une impulsion d'horloge SCL.

Sur une trame ci-dessus, repérer entre pointillés un acquittement d'adresse.

Envoi ou réception :

Si il s'agit d'une lecture, le maître maintient la ligne SDA au NL1, et c'est l'esclave qui la force au NL0 en fonction du poids du bit transmis.

Si il s'agit d'une écriture, l'esclave applique un NL1 sur SDA, et le maître gère les niveaux selon le poids des bits.

Sur une trame ci-dessus, repérer entre pointillés la transmission d'un octet de donnée.

Acquittement de donnée :

Il s'agit maintenant d'un acquittement (ACK) relatif à une donnée, et deux cas sont à envisager :

- Si la donnée a été lue par le maître, celui-ci maintient la ligne SDA au NL1. L'esclave n'a pas à accuser le fait qu'il ait terminé l'envoi des données.
- Si la donnée a été envoyée par le maître, la procédure d'acquittement est identique à celle de l'acquittement d'adresse.

Sur une trame ci-dessus, repérer entre pointillés un acquittement de donnée.

La condition de fin :

La condition de stop est l'inverse de la condition de start. De ce fait, cela implique que les lignes SDA et SCL soit préalablement mises au NL0.

Le maître passe la ligne SCL au NL1, puis après un temps minimum de $4\mu\text{S}$, il passe la ligne SDA au NL1.

Sur une trame ci-dessus, repérer entre pointillés une condition de fin.

5. Vérifications de trames

- a) A l'aide de l'étude faite ci-dessus, vérifier que les signaux donnés dans le dossier page 39 sont bien ceux de l'écriture de la donnée 08h à l'adresse 0Dh de la mémoire série U12.
- b) Faire de même avec les signaux relevés à l'oscilloscope dans l'expérimentation.
- c) Faire de même avec les signaux donnés dans le dossier page 39 et concernant la lecture de la donnée 08h à l'adresse 0Dh.

Étude de FS4-9 : ADAPTATION AU BUS I²C

Objectif : Déterminer le rôle des composants de FS4-9

1. Rappeler le rôle de cette fonction.
2. Rappeler le rôle des résistances R12 et R13.
3. A l'aide de la page 14 de la documentation de Philips intitulée « The I2C-bus and how to use it », indiquer le rôle des résistances R51 et R52.

Étude de FS4-3 : TENSION DE RÉFÉRENCE

Objectifs :

- *Vérifier expérimentalement que la structure remplit sa fonction.*
- *Justifier le choix de D14.*
- *Donner le rôle des condensateurs.*

1. Rappeler le rôle de cette fonction.

2. Expérimentation :

a) Mettre la carte élève sous tension et mesurer le potentiel de la sortie de FS4-3. La structure remplit-elle sa fonction ?

b) Diminuer progressivement la tension d'alimentation de la carte et observer la tension de sortie de FS4-3. En dessous de quelle tension d'alimentation, la tension de sortie n'est-elle plus conforme à ce qu'elle doit être ?

3. A l'aide de la documentation technique du composant D14, justifier la valeur obtenue en sortie de la structure en fonctionnement normal.

4. Indiquer le rôle des condensateurs de la structure.

Étude de FS4-2 : POLARISATION

Objectifs :

- *Vérifier expérimentalement que la structure remplit sa fonction.*
- *Vérifier par calcul les tensions des sorties de la structure.*
- *Donner le rôle des condensateurs.*

1. Rappeler le rôle de cette fonction.

2. Mettre la carte élève sous tension et mesurer les potentiels des sorties de FS4-2
La structure remplit-elle sa fonction ?

3. Justifier par calcul ces tensions de sorties.

4. Indiquer le rôle des condensateurs de la structure.

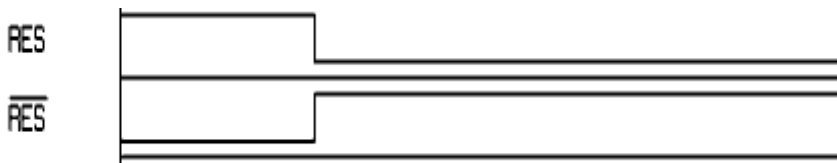
5. Pourquoi le microcontrôleur a-t-il besoin de ces tensions ? Quelle décision doit-il prendre et dans quels cas ?

Étude de FS4-1 : INITIALISATION

Objectifs :

- Relever à l'oscilloscope les signaux RES, RES\ et RESIN\ à la mise sous tension et après action sur SW1.
- Vérifier l'exactitude de ces signaux avec le dossier ressource et la documentation technique de U9.
- Définir le rôle de la structure R7/C16

1. Rappeler le rôle de cette fonction.
2. On souhaite vérifier le comportement suivant à la mise sous tension :



- a. Connectez les sorties RES et RES\ aux voies 1 et 2 de l'oscilloscope numérique. Réglez l'oscilloscope pour qu'il se déclenche à la mise sous tension de la carte.
 - b. Mesurez le temps qui sépare la mise sous tension du changement d'état des deux sorties.
 - c. Imprimez l'oscillogramme.
3. On souhaite vérifier le comportement suivant lors d'un appui sur SW1 :



- a. Connectez l'entrée RESIN\ et la sortie RES aux voies 1 et 2 de l'oscilloscope numérique. Réglez l'oscilloscope pour qu'il se déclenche sur le front descendant de RESIN\
 - b. Mesurez la durée de l'impulsion de la sortie RES lorsqu'on fait un appui bref sur SW1.
 - c. Imprimez l'oscillogramme.
4. A l'aide de la documentation technique du composant U9, calculez la durée théorique du Reset. Comparez cette durée avec celles mesurées aux questions 2b et 3b.
 5. Etude du circuit R7/C16
 - a. Lorsqu'on appuie sur SW1, quelle est la valeur de la tension $V_{RESIN\}$?
 - b. Lorsqu'on relâche SW1, combien faut-il de temps pour que la tension $V_{RESIN\}$ atteigne la valeur 5V ?
 - c. Si on considère que la limite entre les niveaux logiques 0 et 1 se trouve à 2,5V, combien faut-il de temps pour que la tension $V_{RESIN\}$ atteigne cette limite une fois qu'on a relâché SW1 ?
N.B. : On pourra s'aider de cette page pour répondre à ces deux dernières questions : <http://www.planete-sciences.org/robot/wikibot/index.php/Condensateur>
 - d. Quel est le rôle de ces deux composants ?

