

Polytechnique Montréal

Département de génie informatique et génie logiciel

INF1900: Projet initial de système embarqué

Évaluation individuelle sur Moodle

Session hiver 2023

Lundi 20 février 2022, 18h30

Enseignant : Jérôme Collin, ing., M. Sc. A.

Directives:

- l'évaluation est sur 20 points et est évaluée sur Moodle;
- la pondération pour la session est de 30%;
- il y a toujours un seul choix valide parmi les cinq proposés pour les questions de 1 à 15. Ne pas répondre à la question ne fait perdre aucun point. Une bonne réponse donne 1.0 point. Une mauvaise réponse fait perdre 0.2 point;
- la question 16 demande de faire 5 associations. Un point pour chaque bonne association. Aucune perte de point pour une mauvaise association;
- la durée est de 50 minutes.

Question 1 (1 point) [Pour évaluation de la qualité de l'ingénieur 2]

La mécanique d'une interruption doit être bien maîtrisée pour programmer un code qui opère correctement. Quelle est l'affirmation fausse parmi les suivantes? :

- 1) Il faut prévoir une variable globale si la routine d'interruption doit retourner de l'information au reste du déroulement du code.
- 2) Il faut prévoir le moment à partir duquel les interruptions pourront être prises en charge et faire les bons réglages dans le code avant cette étape.
- 3) Il faut sélectionner le bon identificateur d'interruption pour le type d'interruption à gérer pour le périphérique en question.
- 4) Il faut prévoir les cas où la même interruption se produit une seule fois ou si elle peut survenir à nouveau dans le déroulement du programme.
- 5) Il faut prévoir l'endroit dans le code principal pour ne rien faire de valable en attendant qu'une interruption se produise.
- 6***) Je préfère ne pas répondre à la question pour éviter une perte de point.

Note: Choix possible qui revient systématiquement dans toutes les autres questions dans cette évaluation même s'il n'est pas répété ici inutilement. Il est toujours ajouté sur Moodle Examen pour les questions de 1 à 15 inclusivement.

Question 2 (1 point)

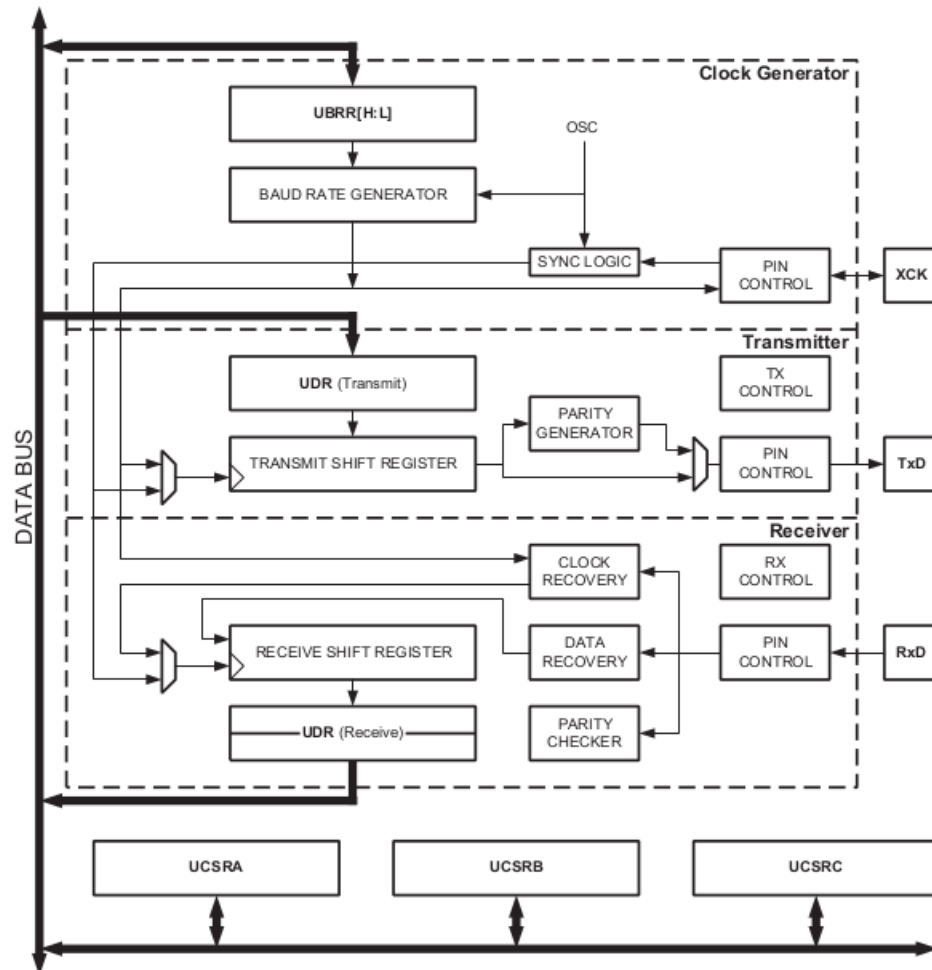
Il n'est pas nécessaire de connaître tout le fonctionnement d'une carte à microcontrôleur utilisée en laboratoire, mais il faut tout de même en savoir un minimum. Par rapport à la carte mère utilisée en laboratoire, une seule de ces affirmations est fausse, laquelle ?

- 1) Le microcontrôleur ATmega8 gère l'interface USB et programme le ATmega324PA.
- 2) Il y a 4 ports de 8 bits sur des broches IDC au haut de la carte mère, chacun ayant une combinaison de broches Vcc et GND accolées à droite aux broches du port.
- 3) Il n'y a que le connecteur USB qui peut alimenter électriquement la carte mère.
- 4) Il y a une diode électroluminescente qui donne l'état du microcontrôleur ATmega8.
- 5) Il y a deux boutons-poussoirs, un libre d'usage pouvant être relié à D2 par un cavalier (*jumper*) sur IntEN, et un autre pour la remise à zéro (*reset*) du ATmega234PA sans cavalier associé à ce bouton-poussoir.

Question 3 (1 point)

Le RS232 est un moyen de communication simple et peu coûteux. La figure qui suit présente un diagramme fonctionnel du USART supportant le protocole dans le ATmega324PA. Toutes les affirmations suivantes qui s'y rattachent sont vraies, sauf une, laquelle?

Figure 19-1. USART block diagram ⁽¹⁾.



- 1) En langage C/C++, les opérateurs de décalage vers la gauche et vers la droite («>>» et «<<<») sont effectués matériellement dans les «SHIFT REGISTER».
- 2) TxD et RxD se trouvent sur le port D en tant que broches sur la carte mère ultimement.
- 3) Comme la vitesse de transmission ou de réception en RS232 peut varier et est indépendante de celle du microcontrôleur, il faut la préciser avec UBRRH et UBRRL.
- 4) On a des unités séparées pour la réception et la transmission ce qui fait qu'on peut faire les deux au même moment sans impact significatif sur la performance.
- 5) On précise des paramètres du protocole en plaçant des valeurs dans UCSRA, UCSRB et UCSRC.

Question 4 (1 point) [Pour évaluation de la qualité de l'ingénieur 2]

Le code qui suit ressemble à un exemple présenté en laboratoire. Qu'est-ce que l'on peut affirmer à propos de ce code qui est faux (une seule réponse) ?

```
#include <avr/io.h>
int main()
{
    DDRA = 0xFF;
    DDRB = 0xFF;
    DDRC = 0xFF;
    DDRD = 0xFF;
    unsigned long compteur=0;

    for(;;) {
        compteur++;
        PORTD = compteur;
        PORTC = compteur >> 8;
        PORTB = compteur >> 16;
        PORTA = compteur >> 24;
    }
    return 0;
}
```

- 1) Pour que ce code présente de l'intérêt, on peut déduire que la variable compteur doit en être une de 32 bits.
- 2) Clairement, on n'utilise pas le bouton-poussoir usager de la carte avec ce code.
- 3) L'inclusion de `avr/io.h` (première ligne) fait en sorte que les identificateurs `DDRx` et `PORTx` sont définis et peuvent être utilisés adéquatement dans ce code.
- 4) Utiliser le multimètre pour mesurer la tension sur les broches du port D n'est vraiment pas utile lorsque ce code roule sur le microcontrôleur.
- 5) Ce code pourrait être plus efficace s'il était reprogrammé en utilisant des interruptions.

Question 5 (1 point) [Pour évaluation de la qualité de l'ingénieur 12]

On peut utiliser une minuterie du ATmega324PA en divers modes selon les besoins. Une affirmation parmi les suivantes est fausse, laquelle ?

- 1) Le mode normal compte de zéro vers la valeur maximale.
- 2) Le mode CTC est comme le mode normal, mais la valeur du compteur retombe à zéro après comparaison réussie.
- 3) Le mode PWM utilise le compteur en incrémentant et en décrémentant.
- 4) Le mode fast-PWM est deux fois plus vite que le mode PWM environ.
- 5) Le mode CDW est pour une mode décrémentant vers zéro uniquement.

Question 6 (1 point)

En considérant l'extrait de code suivant vu en laboratoire, un seul élément parmi les suivants est faux, lequel?

```
void partirMinuterie ( uint16_t duree ) {  
    gMinuterieExpiree = 0;  
    TCNT1 = 'modifier ici' ;  
    OCR1A = duree;  
    TCCR1A = 'modifier ici' ;  
    TCCR1B = 'modifier ici' ;  
    TCCR1C = 0;  
    TIMSK1 = 'modifier ici' ;  
}
```

- 1) Le fait que la variable gMinuterieExpiree débute par un g minuscule est une convention dans le code qui suit une bonne pratique visant à indiquer qu'elle est globale.
- 2) Il se pourrait que la valeur du paramètre duree soit sur 8 bits uniquement dans certains cas.
- 3) On voudra probablement initialiser TCNT1 à zéro dans ce code.
- 4) Matériellement, TCNT1 et OCR1A serviront d'entrées à un circuit combinatoire d'égalité dans la minuterie.
- 5) TCCR1A et TCCR1B sont des variables volatiles déclarées plus haut dans ce fichier.

Question 7 (1 point)

Un multimètre est un appareil de mesure de base. Toutes les affirmations suivantes le concernant sont vraies, sauf une, laquelle ?

- 1) On peut s'en servir pour mesurer une tension continue comme évaluer le voltage d'une pile par exemple.
- 2) Un test de continuité suppose que la résistivité est pratiquement nulle ou, au contraire, très élevée entre deux points considérés.
- 3) Il faut toujours utiliser les deux sondes de l'appareil, la noire et la rouge pour prendre les mesures.
- 4) Le multimètre mesure la pression électrique d'un composant.
- 5) L'avantage du test de continuité est qu'un son est émis lorsqu'il y a continuité électrique ce qui permet d'éviter d'avoir à regarder l'afficheur de l'appareil pour prendre l'information.

Question 8 (1 point)

Le pont-en-H joue un rôle fondamental pour le contrôle des moteurs à courant continu du robot. Toutes les affirmations qui s'y rapportent sont vraies, sauf une, laquelle ?

- 1) Le pont-en-H fonctionne de façon séquentielle puisqu'il est une machine à états finis.
- 2) Il est important pour la carte mère et le pont-en-H de partager une même masse électrique (*ground*) pour donner aux signaux de contrôle une référence commune.
- 3) La carte mère et le pont-en-H sont alimentés par des tensions différentes.
- 4) Un oscilloscope permet d'observer le pourcentage d'un signal PWM et son effet sur la vitesse des roues. Par contre, un multimètre est suffisant pour déboguer un problème avec le pont-en-H.
- 5) Le circuit du pont-en-H du robot utilisé en laboratoire est en fait constitué de deux ponts-en-H puisqu'il y a deux moteurs à courant continu sous le robot.

Question 9 (1 point) [Pour évaluation de la qualité de l'ingénieur 12]

On insiste sur des règles de qualité du code. Toutes les affirmations sur les règles qui suivent sont vraies, sauf une qui est fausse, laquelle ?

- 1) L'indentation devrait être régulière et être de 2 à 4 espaces.
- 2) Une ligne ne devrait pas dépasser 80 caractères.
- 3) Une variable devrait toujours débuter par une lettre majuscule et chaque changement de mot devrait commencer aussi par une majuscule.
- 4) On devrait éviter d'écrire `for (; ;)` et privilégier `while (true)`.
- 5) Le nom des constantes doit être en majuscules avec une séparation par le caractère souligné («`_`») entre chaque mot.

Question 10 (1 point) [Pour évaluation de la qualité de l'ingénieur 2]

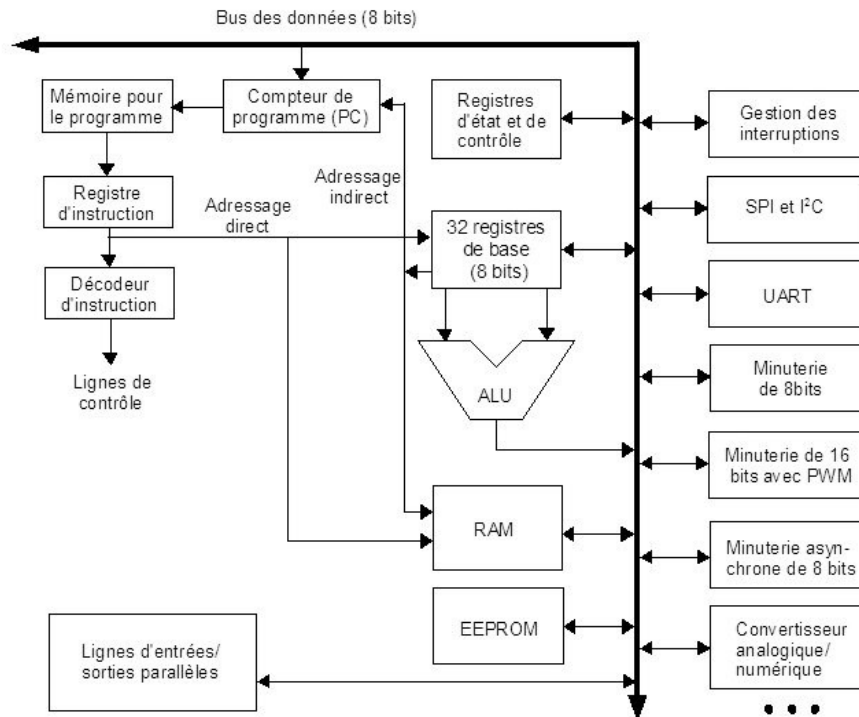
En langage C/C++, quel est la valeur de la variable `result` à la fin en hexadécimal ?

```
uint8_t result = 0x17 ;
result = 0xF749 || result ;
result = ( result >> 1 ) & 0xFE ;
result = ~result;
```

- 1) 0xD1.
- 2) 0x01.
- 3) 0xFF.
- 4) 0xEE.
- 5) 0x00.

Question 11 (1 point) [Pour évaluation de la qualité de l'ingénieur 2]

En vous référant à la figure suivante de l'architecture générale d'un AVR de la série Mega, identifier l'affirmation qui est fausse ?



- 1) Le compteur de programme changera de valeur de façon importante au moment de gérer une interruption.
- 2) Les lignes d'entrées/sorties parallèles font référence aux PORTx, PINx et DDRx.
- 3) Les lignes de contrôle mènent vers des machines à états finis matérielles qui vont contrôler l'ensemble des opérations nécessaires à la réalisation des instructions même si elles ne sont pas montrées sur la figure.
- 4) On peut utiliser un seul des périphériques de la partie de droite de la figure à la fois.
- 5) La gestion des données à placer dans la RAM et des registres de base est surtout de la responsabilité du compilateur lorsqu'on programme en C/C++.

Question 12 (1 point)

Il y a une mémoire externe EEPROM sur la carte mère et elle est à l'extérieur du ATmega324PA. Les affirmations qui suivent sont toutes vraies, sauf une, laquelle ?

- 1) Cette mémoire est différente de la mémoire EEPROM interne au ATmega324PA.
- 2) Cette mémoire fait 512 kilobits.
- 3) Elle est accédée par protocole I²C, aussi appelé TWI.
- 4) La librairie AVRLibC n'offre pas d'interface directe vers les puces externes, seulement pour les ressources internes de l'AVR. C'est pourquoi on distribue une classe C/C++ qui permet son utilisation de façon simple.
- 5) C'est une mémoire qui permet d'augmenter la grandeur de la mémoire flash interne de ATmega324PA pour accommoder des programmes plus longs.

Question 13 (1 point)

Le phénomène de rebond introduit au début du cours nécessite un traitement très particulier. Toutes les affirmations sur ce problème sont vraies, sauf une, laquelle ?

- 1) Il s'agit surtout d'un phénomène qui touche les interrupteurs mécaniques.
- 2) Le phénomène se produit, car il y a contact entre des plaques métalliques flexibles qui rebondissent l'une sur l'autre avant que le contact se stabilise.
- 3) Le signal d'entrée prend typiquement la forme de quelques petites impulsions avant que le signal se stabilise.
- 4) Le problème se corrige par un délai lors d'une scrutation ou d'une interruption, au choix, dans le code.
- 5) Le rebond peut se produire aussi en sortie, particulièrement pour une DEL (diode électroluminescente).

Question 14 (1 point) [Pour évaluation de la qualité de l'ingénieur 12]

Le PWM (Pulse-Width Modulation) est très utile avec le robot. On peut affirmer que tous ces énoncés le concernant sont vrais, sauf un, lequel ?

- 1) Généralement, on évalue le PWM en pourcentage.
- 2) Le PWM produit matériellement par une minuterie permet de dégager le CPU du microcontrôleur de le générer de façon logicielle.
- 3) Un signal PWM se connecte sur une des broches «E» du pont-en-H pour contrôler la vitesse de moteurs.
- 4) Souvent, en mode PWM, on est limité avec les minuteries du ATmega324PA à quelques fréquences fixes de PWM, mais ceci n'est pas un inconvénient majeur puisque le contrôle des moteurs n'est pas sensible à la fréquence de PWM lorsque celle-ci est dans les quelques kilohertz.
- 5) Le PWM varie en amplitude si la carte est alimentée par une tension plus élevée.

Question 15 (1 point) [Pour évaluation de la qualité de l'ingénieur 12]

Pour bien développer du code avec Git en équipe, il faut respecter son fonctionnement. Une des affirmations suivantes n'est pas vraie, laquelle ?

- 1) On fait un `git add` sur un fichier une fois et on a plus besoin de le faire par la suite.
- 2) La commande `git status` . permet de voir les fichiers qui ne sont pas pris en charge par Git dans le répertoire courant.
- 3) Une commande Linux `ls -a` permet de voir le répertoire `.git` et de déduire la présence de la racine Git dans mon compte local. On a avantage à ne pas modifier quoi que ce soit dans ce répertoire pour que Git fonctionne correctement.
- 4) La commande `git log` permet de voir les commentaires de toutes les versions laissées par les personnes ayant envoyé des modifications au fichier au serveur (avec `git push`).
- 5) Un `git commit` sur un fichier que je fais présentement est une commande dont les autres personnes ayant accès à l'entrepôt Git peuvent voir l'effet directement.

Note: correction d'une erreur – il y a deux réponses possibles dans les choix proposés pour cette question.

Question 16 (5 points)

Pour chacun des cinq énoncés, associer le bon terme en choisissant parmi les choix proposés.

Choix possibles:

- | | |
|-----------------|------------------------|
| 1 – on / off | 11 – swith case |
| 2 – cli / sei | 12 – ISR () |
| 3 – set / unset | 13 – enum / enum class |
| 4 – while | 14 – Broche |
| 5 – DDRx | 15 – PINx |
| 6 – _delay_ms() | 16 – dynamic cast |
| 7 – const | 17 – #define |
| 8 – uint8_t | 18 – EIMSK |
| 9 – char [] | 19 – int16_t |
| 10 – En / Dis | 20 – INT0 |

A) Type pour représenter les états une machine à états finis logiciels en C/C++ : _____

B) Type pour représenter une variable C/C++ ayant une largeur naturelle de mot correspondante à celle du ATmega324PA : _____

C) Pour contrôler la direction d'un port : _____

D) Nécessaire en C/C++ pour définir F_CPU : _____

E) Pour permettre ou bloquer les interruptions dans des sections de code : _____