

## Question 1

Terminer

Note de 1,00  
sur 1,00

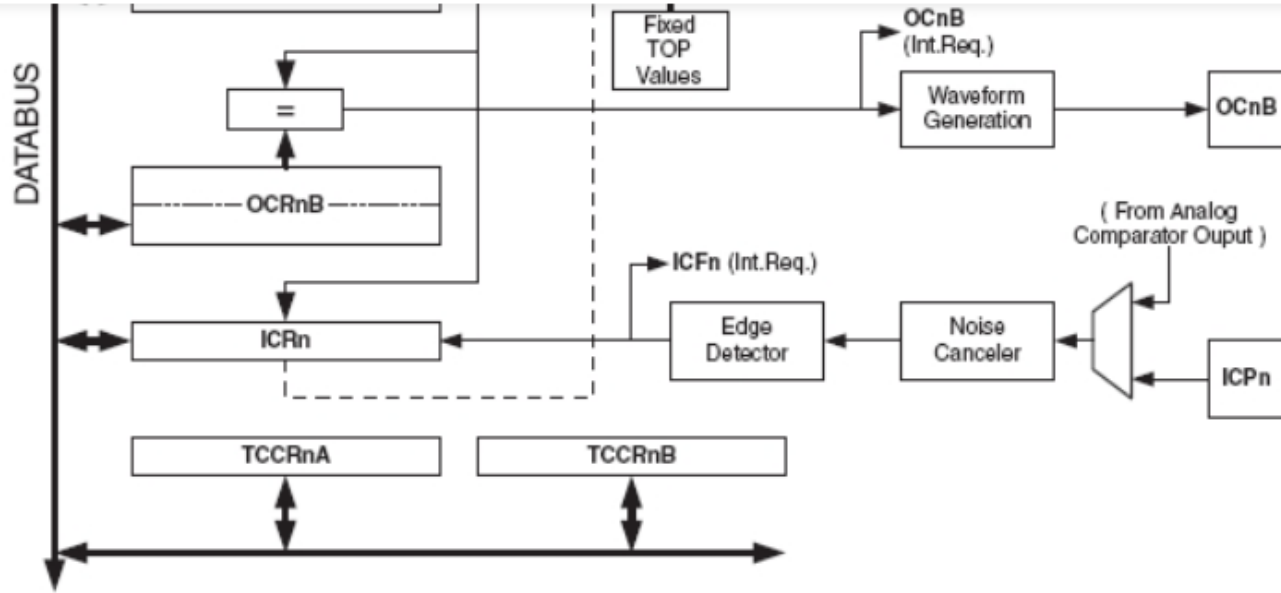
🚩 Marquer la  
question

Une seule de ces affirmations est vraie concernant SimulIDE, laquelle ?

- ☐ 1. On a pas besoin d'ajuster la fréquence d'horloge du ATmega324 dans SimulIDE puisqu'elle peut être déduite du programme directement.
- ☐ 2. L'outil Git est intégré à SimulIDE pour éviter d'avoir à utiliser des commandes textuelles.
- ☐ 3. Il est impossible d'arrêter et la simulation une fois qu'elle est démarrée dans l'interface usager.
- ☒ 4. On peut facilement savoir si une broche ou un fil est à zéro ou à un en ajoutant une composante «probe» dans le circuit.
- ☐ 5. Le RS232 reste un moyen facile de programmer un microcontrôleur avec le simulateur.
- ☐ 6. Je préfère ne pas répondre à la question pour éviter une perte de point.

Marquer la question

Un seul élément parmi les suivant est faux, lequel?



Un seul élément parmi les suivant est faux, lequel?

- ☐ 1. Généralement, TCNTn voit sa valeur changée beaucoup plus souvent que OCRnA ou OCRnB.
- ☐ 2. Même si la figure n'est pas complète sur cet aspect, TCCRnA et TCCRnB sont les registres qui permettent le choix du mode de fonctionnement de la minuterie.
- ☐ 3. Lorsqu'on désire générer un signal PWM constant, on ajuste des registres de cette unité et on a plus besoin de les réajuster par la suite.
- ☒ 4. Le ATmega324 possède quatre autres minuteries à peu près semblables à celle-ci, mais elles sont de 8 bits seulement.
- ☐ 5. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 6. Le nombre d'interruptions différentes que peut servir un microcontrôleur comme le ATmega324PA est fixe et connu.

Question 3

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Les mémoires jouent un rôle crucial dans les microcontrôleurs et il faut comprendre le rôle des plus importantes. Toutes ces affirmations les concernant sont vraies, sauf une, laquelle ?

- ☐ 1. Un registre de configuration d'un périphérique peut, à la rigueur, être vue comme une mémoire puisqu'il retient des valeurs qu'on peut aller relire et qu'il est accessible à une adresse précise.
- ☐ 2. Un des avantages du EEPROM interne est qu'on n'a pas besoin d'une puce externe pour stocker des valeurs de façon persistante.
- ☒ 3. Les données stockées dans le EEPROM sont toujours sur 32 bits, même si le processeur a une architecture de 8 bits à la base.
- ☐ 4. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 5. Le fichier de registres (*register file*) du CPU du microcontrôleur forme aussi une mémoire puisque les registres y retiennent des valeurs. Par contre, ces registres sont gérés par le compilateur et on évite de les modifier explicitement dans le code soi-même comme programmeur en langage C.
- ☐ 6. La plus grande différence entre un EEPROM par rapport à la RAM est que le EEPROM conserve ses valeurs lorsque le microcontrôleur démarre ou redémarre (remise à zéro - reset) alors que la RAM est réinitialisée.

Question 4

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

La conversion analogique/numérique est un aspect important des microcontrôleurs. Toutes les affirmations suivantes sont vraies sauf une qui est inappropriée, laquelle ?

- ☐ 1. Une photorésistance placée en série avec une résistance peut former un capteur analogique pouvant mesurer la quantité de lumière ambiante.
- ☐ 2. La conversion analogique/numérique permet d'abord et avant tout d'associer des valeurs de tension à des valeurs numériques.
- ☐ 3. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 4. On pourrait se servir d'une source de tension dans SimulIDE pour simuler le travail d'un capteur analogique.
- ☒ 5. L'ajustement de la référence analogique modifie les valeurs de voltage à convertir, pas les valeurs numériques résultant de la conversion.
- ☐ 6. Une conversion avec le ATmega324PA produit toujours un résultat numérique sur 10 bits.

Question 5

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Les opérations sur des bits sont fondamentales avec la programmation, surtout avec les microcontrôleurs. Les affirmations qui suivent sont toutes vraies, sauf une, laquelle ?

- ☐ 1. Par définition, la valeur 1 est interprétée comme vraie en langage C/C++ et 0 comme faux.
- ☐ 2. Les opérations sur des bits sont pratiquement nécessaires lorsqu'on manipule uniquement une partie d'un port de 8 bits du ATmega324PA pour éviter de modifier les autres autres parties du port en question.
- ☒ 3. Les opérations sur des bits fonctionnent sur le ATmega324PA uniquement parce qu'elles sont supportées par la librairie AVRLibC.
- ☐ 4. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 5. Un registre qui subit un décalage à droite de 1 bit suivi d'un décalage à gauche de 1 bit peut avoir une valeur différente après ces deux décalages successifs par rapport à sa valeur au départ.
- ☐ 6. Après un complément à un de la valeur 0x00 sur 8 bits, on se retrouvera avec la valeur 0xFF.

Question **6**

Terminer

Note de 5,00  
sur 5,00

🚩 Marquer la  
question

Pour chacun des cinq énoncés (1 point chacun), choisir le bon terme ou concept parmi les choix proposés dans le menu déroulant.

Pour atténuer progressivement l'intensité d'une diode électroluminescente (DEL).

réduction du pourcentage de PWM ▾

Est une commande Linux.

cd ▾

Est une résistance ajustable.

potentiomètre ▾

Utile dans SimulIDE pour mesurer et s'assurer de la valeur de la référence analogique.

voltmètre ▾

Problème en entrée associé au bouton-poussoir.

rebond ▾

Question **7**

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Git est un outil de développement logiciel qui doit être bien utilisé. Un élément parmi les suivants est faux, lequel ?

- ☐ 1. Un « git pull » me permet de voir la correction d'un travail pratique du cours par les chargés de laboratoire une fois la correction terminée.
- ☐ 2. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 3. Il est bon de faire une commande « make clean » avant de faire un « git add » pour éviter d'inclure des fichiers produits par la compilation du code.
- ☒ 4. Une commande « git attempt » permet de faire un test pour vérifier si un conflit pourrait se produire sur un fichier dans mon répertoire.
- ☐ 5. Introduire un fichier .gitignore bien ajusté permet d'éviter que Git prenne en charge des fichiers qui ne devraient jamais se retrouver sous son contrôle.
- ☐ 6. On peut toujours faire un « git rm » sur un fichier pour le retirer du contrôle de Git, mais il restera toujours dans l'entrepôt pour pouvoir restaurer les versions passées de ce fichier au besoin.



Question 8

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Pour contrôler un bouton-poussoir par interruption, il faut un code bien ajusté. Les affirmations qui suivent sont toutes vraies, sauf une, laquelle ?

- ☐ 1. Prévoir l'ajustement d'un registre pour s'assurer du type de changement de niveau logique sur signal d'interruption (signal montant ou descendant, etc.)
- ☒ 2. S'assurer qu'il s'agit de la seule interruption utilisée dans le système.
- ☐ 3. S'assurer que le diviseur de tension constitué d'un bouton-poussoir et d'une résistance soit relié au port D2 ou un autre port permettant une interruption externe.
- ☐ 4. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 5. Associer le bon type d'interruption en paramètre de la routine ISR.
- ☐ 6. Avoir dans le début du code une directive `#include <avr/interrupt.h>` .

Question 9

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Une des affirmations suivantes n'est pas vraie concernant Git, laquelle ?

- ☐ 1. La commande « git log » permet de voir les commentaires associés aux versions d'un fichier.
- ☐ 2. L'option -m à la commande « git commit » permet d'ajouter un commentaire qui pourra être consulté et qui sera associé à cette version pour fins de documentation.
- ☒ 3. Une commande « git history » permet de voir les dernières commandes git effectuées pour être capable de les répéter dans le même ordre plus facilement par la suite.
- ☐ 4. Une commande « git pull » permet de récupérer les modifications les plus à jour dans ma copie locale. Ceci peut créer des conflits. Donc, il vaut mieux exécuter cette commande assez régulièrement pour éviter de trop nombreux et complexes conflits et maintenir sa propre copie assez à jour.
- ☐ 5. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 6. Les branches dans git sont plus complexes à gérer, mais isolent un peu plus mes modifications locales ce qui diminue le risque de conflits directs par la suite. La fusion avec le reste du code devient un peu plus complexe cependant et doit être faite avec doigté.

Question **10**

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Toutes ces affirmations concernant les signaux PWM sont vraies, sauf une, laquelle?

- ☐ 1. Peuvent être générés par une minuterie.
- ☒ 2. Se contrôlent à partir d'un bouton-poussoir.
- ☐ 3. Sont utiles au contrôle de la vitesse des moteurs avec un pont-en-H.
- ☐ 4. Peuvent être visualisés dans SimulIDE avec un oscilloscope virtuel.
- ☐ 5. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 6. Se mesurent souvent en pourcentage.

Question **11**

Terminer

Note de -0,20  
sur 1,00

🚩 Marquer la  
question

Les machines à états finis sont importantes pour contrôler des opérations avec un microcontrôleur. Toutes les affirmations qui suivent sont vraies, sauf une, laquelle ?

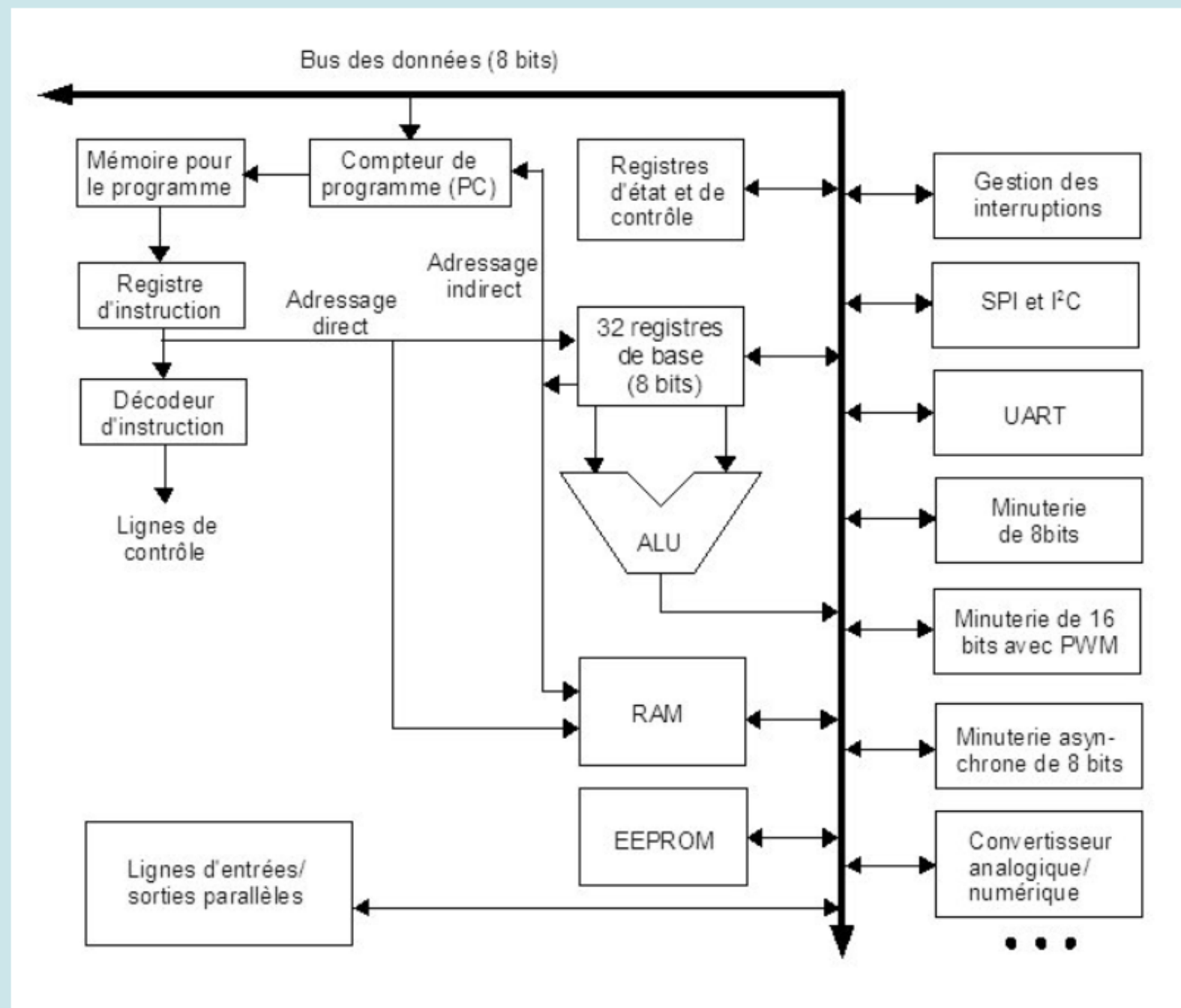
- ☐ 1. Il est impératif d'avoir un état initial au départ pour éviter de débiter dans n'importe quel état inconnu.
- ☒ 2. Il serait possible de produire une machine à états finis dans SimulIDE qui serait faite de portes et de bascules et qui seraient connectées aux broches d'un ATmega324 pour interagir avec lui.
- ☐ 3. Une machine à états bien établie facilite la lecture du code pour une autre personne.
- ☐ 4. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 5. Une machine à états finis demande toujours l'utilisation d'un masque sur des opérations sur des bits d'un registre pour être implémentée correctement.
- ☐ 6. Il est utile d'avoir une variable qui représente l'état de la machine et qu'elle puisse prendre les valeurs possibles d'un type énuméré (enum) qui liste les différents états possibles.

## Question 12

Terminer

Note de 1,00  
sur 1,00Marquer la  
question

À partir de la figure suivante:



Quelle est l'affirmation fausse parmi les suivantes? :

Quelle est l'affirmation fausse parmi les suivantes ?

- ☐ 1. Une programmation de la puce ATmega324 n'affectera pas le contenu de la mémoire EEPROM.
- ☒ 2. Le convertisseur analogique/numérique génère un signal analogique en sortie après conversion des données numériques en mémoire.
- ☐ 3. Comme les minuteries et autres périphériques sont inactifs dans un programme au départ à moins de les activer explicitement, on peut déduire qu'une remise à zéro replace les registres de configuration des périphériques à leurs valeurs par défaut.
- ☐ 4. Une programmation de la puce ATmega324 amène une remise à zéro et donc le début d'une nouvelle exécution du programme. La mémoire RAM et donc affectée elle aussi.
- ☐ 5. Le bus des données est interne et ne sort pas directement sur des broches de la puce vers l'extérieur.
- ☐ 6. Je préfère ne pas répondre à la question pour éviter une perte de point.

Question **13**

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Le RS232 est un protocole utilisé par le ATmega324PA. Les affirmations qui suivent sont toutes vraies, sauf une, laquelle ?

- ☐ 1. On peut opter pour deux types d'interfaces avec SimulIDE, une connecte vers un vrai port RS232 physique du PC et l'autre vers une console bleue où on voit les données envoyées et reçues.
- ☐ 2. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 3. Le USART0 utilise les deux derniers bits du port D du ATmega324PA.
- ☒ 4. Il y a un mécanisme en place dans un USART pour redemander la transmission des octets mal reçus.
- ☐ 5. On peut transmettre 7, 8 ou 9 bits utiles à la fois avec ce protocole.
- ☐ 6. Le registre UDR est là où on envoie ou on récupère les données reçues ou envoyées par RS232.

Question **14**

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Le ATmega324PA supporte plusieurs types d'interruptions. Toutes les affirmations suivantes sont vraies, sauf une, laquelle ?

- ☐ 1. ISR peut faire en sorte qu'on doive recourir à des variables ayant le qualificatif «volatile ».
- ☐ 2. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 3. On ne trouve pas, à parcourir le code, un appel explicite à une fonction ISR. L'électronique du microcontrôleur appelle directement la fonction au bon moment.
- ☒ 4. La fonction ISR retourne toujours un entier de 8 bits uint8\_t.
- ☐ 5. Un paramètre en argument de la fonction ISR caractérise quel type d'interruption est prise en charge.
- ☐ 6. Une fonction ISR a avantage à être écrite pour qu'elle s'exécute rapidement.



Question **15**

Terminer

Note de 1,00  
sur 1,00

🚩 Marquer la  
question

Le pont-en-H permet de faire tourner des moteurs à courant continu. Toutes les affirmations qui s'y rapportent sont vraies, sauf une, laquelle ?

- ☐ 1. En général, avec un oscilloscope branché au pont-en-H, on s'intéresse plus à la largeur des pulses que de la fréquence de l'onde.
- ☐ 2. On peut inverser le sens de rotation d'un moteur en inversant les fils à ses bornes, donc en inversant les signaux provenant du pont-en-H.
- ☐ 3. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 4. Le pont-en-H permet d'augmenter le courant par rapport à ce qu'une carte mère peut normalement fournir.
- ☒ 5. Si on utilise le timer1 pour générer du PWM de manière matérielle, on peut faire sortir les signaux sur les broches de notre choix.
- ☐ 6. Dans SimulIDE, un circuit avec un pont-en-H et un microcontrôleur qui génère du PWM de façon matérielle n'a pas à être modifié si on choisit plutôt de générer du PWM de façon logicielle plus tard.

Question **16**

Terminer

Note de 1,00

sur 1,00

🚩 Marquer la question

Les mécanismes d'interruption et de scrutation nécessitent plusieurs aspects pour se concrétiser. On peut affirmer que tous ces énoncés sont vrais, sauf un, lequel ?

- ☐ 1. On ne peut pas avoir deux fonctions ISR pour le même type d'interruption dans le même programme.
- ☐ 2. Je préfère ne pas répondre à la question pour éviter une perte de point.
- ☐ 3. On ne peut ajouter des paramètres à la fonction ISR nous-même autres que ceux déjà présents.
- ☒ 4. Il est impossible d'utiliser la fonction `_delay_ms()` dans du code avec interruption.
- ☐ 5. La scrutation est une initiative de la partie CPU pour vérifier l'état d'un signal ou d'un périphérique.
- ☐ 6. Comme il est possible que les interruptions modifient des variables globales, il devient possible dans le reste du code de vérifier les changements de valeurs de celles-ci.