Moodle Cours ▼ Aide ▼ Liens utiles ▼

Terminé le lundi 8 mars 2021,

Marquer la

question

Question 2

Note de 1,00

Marquer la

Terminer

sur 1,00

question

Question 3

Note de 1,00

Marquer la

**Terminer** 

sur 1,00

question

Question 4

Note de 1,00

Marquer la

Terminer

sur 1,00

question

Question **5** 

Note de 1,00

Marquer la

sur 1,00

question

Question **6** 

Note de -0,20

Marquer la

Terminer

sur 1,00

question

Question **7** 

Terminer

sur 1,00

question

Question **8** 

Note de 1,00

Marquer la

Terminer

sur 1,00

question

Question 9

Note de 1,00

Marquer la

Question 10

Note de 4,00

Marquer la

Question 11

Note de 1,00

Marquer la

Question 12

Note de 1,00

Marquer la

Question 13

Note de 1,00

Marquer la

Question 14

Note de 1,00

Marquer la

Question 15

Note de 1,00

Marquer la

Terminer

sur 1,00

question

Terminer

sur 1,00

question

Terminer

sur 1,00

question

Terminer

sur 1,00

question

un, lequel?

coder.

Terminer

sur 1,00

question

Terminer

sur 5,00

question

Terminer

sur 1,00

question

Note de 1,00

Marquer la

INF1900 - Projet initial de système embarqué Tableau de bord / Mes cours / INF1900 - Projet initial de système embarqué / Quiz / Quiz du 8 mars 18h30

**NAVIGATION DU TEST** 

3 ||

Afficher une page à la fois

Terminer la relecture

Examen

Commencé le lundi 8 mars 2021, **État** Terminé

Temps mis Note 17,80 sur 20,00 (89%) Question 1 En vous référant au code suivant, identifier l'affirmation qui est fausse? Terminer #include <avr/io.h> Note de 1.00 sur 1,00

DDRD = 0xff;unsigned long compteur=0; for(;;) compteur++; PORTD = compteur; PORTC = compteur >> 8; PORTB = compteur >> 16; PORTA = compteur >> 24; return 0;

int main()

DDRA = 0xFF;

DDRB = 0xFF; DDRC = 0xFF;

○ 1. Il est justifié d'utiliser un type unsigned long pour le compteur plutôt qu'un uint8\_t dans le cas présent. 2. La valeur du compteur repasse par zéro indéfiniment.

• 3. Il est compliqué de construire un circuit dans SimulIDE permettant de visualiser l'exécution de ce que produit ce code.

4. Je préfère ne pas répondre à la question pour éviter une perte de point.

• 6. Sans la directive d'inclusion de fichier tout au haut, il y aurait une erreur de compilation sur les identificateurs propres aux microcontrôleurs AVR comme les DDRx et PORTx. Les machines à états finis sont importantes pour contrôler des opérations avec un microcontrôleur. Toutes les affirmations qui suivent sont vraies, sauf

5. L'instruction « return 0 » ne devrait jamais être atteinte.

une, laquelle? 1. Dans SimulIDE, si toutes les entrées sont des boutons-poussoirs qu'on doit activer avec la souris, il devient pratiquement impossible de changer deux entrées en même temps.

code.

0 2. Le fait d'avoir un code structuré par un bon diagramme des états permettra l'ajout d'états supplémentaires plus facilement dans le futur si la machine doit être modifiée. O 3. On doit toujours décoder les sorties à partir de l'état présent et peut-être aussi des entrées. On peut le faire dans un « switch-case ». 4. Je préfère ne pas répondre à la question pour éviter une perte de point.

Une seule de ces affirmations est <u>fausse</u> concernant SimulIDE? 1. SimulIDE produit des fichiers binaires du circuit ayant l'extension .simide. 2. Je préfère ne pas répondre à la question pour éviter une perte de point. ○ 3. La composante « DEL bicolore » peut prendre la couleur rouge ou verte selon le sens de la chute de tension à ses bornes.

• 5. On doit absolument avoir recours aux interruptions pour la conception d'une machine à états finis.

O 6. Il est important d'avoir des noms d'état très significatifs dans une machine à états finis logicielle pour faciliter sa compréhension à la lecture du

5. Il faut placer la tension maximale souhaitée pour la conversion analogique / numérique sur Aref sur le ATmega324 dans le circuit si on utilise le convertisseur analogique / numérique. O 6. Sans être connectée au reste du circuit, la composante « lumière » peut servir à simuler la quantité de lumière émise dans l'environnement.

4. On peut inclure dans le circuit une source de tension fixe.

La minuterie et le PWM sont très utiles en situation de contrôle. Un élément <u>parmi les suivants est faux</u>, lequel? 1. Le pont-en-H utilisé en laboratoire est conçu pour prendre directement deux signaux PWM générés par le ATmega324 PA pour faciliter les connexions. O 2. En mode PWM phase correcte, le compteur incrémente, atteint le maximum et décrémente par la suite ce qui fait qu'on obtient une rampe

montante et une descendante. O 3. La documentation d'Atmel nous offre des formules mathématiques pour calculer la fréquence du PWM généré par une minuterie. 4. Avec un PWM de 50%, le signal passe autant de temps au niveau haut qu'au niveau bas.

 5. Je préfère ne pas répondre à la question pour éviter une perte de point. 6. La minuterie 1 est de 8 bits. Les minuteries 0 et 2 sont de 16 bits.

sauf une, laquelle? • 1. On décale toujours vers les bits de poids fort, jamais vers les bits de poids faible. 2. Je préfère ne pas répondre à la question pour éviter une perte de point.

Les opérations sur des bits sont fondamentales avec la programmation, surtout sur microcontrôleurs. Les affirmations qui suivent sont toutes vraies,

0 3. L'utilisation de noms de bits d'un registre, combinée à l'utilisation d'opérations sur des bits, permet d'augmenter la lisibilité du code lors de

l'affectation d'un registre de configuration par rapport à l'affectation d'une valeur binaire ou d'une valeur hexadécimale directement.

Gestion des

interruptions

SPI et I2C

UART

Minuterie

de 8bits

Minuterie asynchrone de 8 bits

Convertisseur an alogique/

. . .

numérique

○ 6. L'opération A = 0x03 & 0x01 produit le même résultat que A = 0x03 & 0x01, mais il suffirait que les valeurs numériques changent pour que ce ne soit pas toujours vrai de façon générale.

Bus des données (8 bits)

Compteur de

programme (PC)

Adressage

direct

À partir de la figure suivante:

Mémoire pour

le programme

Registre

d'instruction

Décodeur

d'instruction

Lignes d'entrées/

sorties parallèles

Quelle est l'affirmation <u>fausse</u> parmi les suivantes? :

 $\bigcirc$  4. Les expressions PIND == 0x04 et (PIND & 0x04) == 0x04 sont équivalentes.

○ 5. L'ALU est utilisé matériellement pour supporter les opérations sur des bits précisés dans le code.

Registres

contrôle

32 registres

de base (8 bits)

RAM

**EEPROM** 

d'état et de

Lignes de ALU contrô le Minuterie de 16 bits avec PWM

Adressage

indirect

0 1. On peut très bien utiliser certains périphériques sans nécessairement avoir besoin de gérer des interruptions. 2. Je préfère ne pas répondre à la question pour éviter une perte de point. ○ 3. Les opérations d'addition en C/C++ se réalisent dans l'unité arithmétique et logique (ALU). • 4. Les lignes de contrôle sur la figure mènent à des machines à états finis qui commandent l'exécution des opérations de tout le microcontrôleur. ○ 5. Un USART peut envoyer au convertisseur analogique / numérique une valeur analogique à convertir en numérique. O 6. Le déclenchement d'une routine d'interruption amènera l'exécution de la routine ISR correspondante et le compteur de programme PC changera pour pointer vers cette fonction en mémoire de programme. Le pont-en-H et le PWM sont importants dans les situations de contrôle. Toutes les affirmations qui s'y rapportent sont vraies, sauf une, laquelle

1. Je préfère ne pas répondre à la question pour éviter une perte de point.

4. Un signal PWM de 0% est un signal à tension nulle (0 Volt).

programmation qu'on effectue.

ATmega324.

vraies, <u>sauf une</u>, laquelle?

(reset).

• 6. Il est toujours bon d'utiliser un oscilloscope pour s'assurer du pourcentage de PMW réellement généré, qu'il le soit de manière logicielle ou matérielle.

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

○ 5. On doit placer une instruction #include <avr/interrupt.h> dans le code avant de pouvoir utiliser les interruptions.

signal se gère de la même façon, peu importe si on fait du PWM matériel ou logiciel.

• 2. On doit toujours utiliser un pont-en-H pour atténuer l'intensité d'une diode électroluminescente (DEL - LED).

valeurs aux registres de comparaisons (OCCRnX) de la minuterie tout en gardant le reste de la configuration inchangée.

 3. L'interruption doit toujours être écrite dans un fichier séparé du reste du code. 4. L'interruption a un caractère imprévisible puisqu'elle peut survenir à n'importe quel point du déroulement normal du code (« main » et fonctions appelées directement ou indirectement à partir du « main »).

Les mémoires EEPROM jouent un rôle important dans les microcontrôleurs et il faut comprendre leurs rôles. Toutes ces affirmations les concernant sont

1. Contrairement à la RAM, la mémoire EEPROM conserve ses données même si le microcontrôleur n'est plus alimenté ou subit un remise à zéro

1. Une routine d'interruption peut être appelée plusieurs fois durant le déroulement d'un programme ou une seule fois selon le réglage par

2. La documentation d'AVRLibC donne la liste de tous les types d'interruptions possibles avec leur identificateur pour un microcontrôleur comme le

O 3. En général, avec du PWM matériel et un pont-en-H, on vise à ce qu'un changement de vitesse se résume simplement à l'assignation de nouvelles

0 5. La broche D d'un pont-en-H est connectée directement avec une broche du microcontrôleur pour le contrôle du sens de rotation d'un moteur. Ce

○ 2. Il existe aussi des mémoires EEPROM externes accessibles par un protocole supporté par le ATmega324 même si elles n'ont pas été utilisées en laboratoire. ○ 3. La mémoire EEPROM du ATmega324 n'est pas très grande, 1024 octets (1 Kilo-octet) seulement.

• 4. On doit toujours initialiser la mémoire EEPROM même lorsqu'on ne l'utilise pas.

6. Le EEPROM n'a pas besoin d'avoir accès à des broches externes du ATmega324.

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

5. Je préfère ne pas répondre à la question pour éviter une perte de point.

Pour gérer un bouton-poussoir sur D2 par interruption :

possiblement entrer d'autres commandes :

6. Je préfère ne pas répondre à la question pour éviter une perte de point.

INTO\_vect \$ Commande Linux pour créer un répertoire : mkdir Composante dans SimulIDE pour tester rapidement si une sortie est à zéro ou un : probe

O 3. Dans SimulIDE, ouvrir le «Serial Monitor» permet de recevoir et visualiser directement les données émises en RS232 par un ATmega324.

1. On identifie une routine d'interruption non par son nom qui est toujours ISR(), mais par l'identificateur du type d'interruption en argument.

○ 4. Le mécanisme d'interruption perturbe le déroulement des activées normales du CPU et du programme alors que la scrutation implique des

○ 5. La scrutation peut éviter d'avoir recours à des variables globales, contrairement au mécanisme d'interruption, et est souvent plus simple à

(Int.Req.)

Canceler

O 2. On peut contrôler l'intensité d'une diode électroluminescente (DEL ou LED) avec utilisant OCnA ou OCnB en sortie à l'extrême droite sur la

0 2. La commande « git help log » est utile pour obtenir une description de la sous-commande log de git et comment elle peut être utilisée avec des

Comparator Ouput

0 2. Le qualificatif «volatile» est important à utiliser à la déclaration d'une variable globale utilisée dans une routine d'interruption.

• 3. Une routine d'interruption a avantage à prendre du temps pour s'exécuter, car elle exécute toujours du code important.

Pour envoyer une commande Linux entrée au terminal en arrière-plan et ainsi pouvoir regagner l'inviter de commande (prompt) pour

anti-rebond ♦

cg

Pour stabiliser les multiples contacts entre pièces métalliques d'un interrupteur mécanique et éviter les double lectures :

Le RS232 est un protocole utilisé par le ATmega324. Les affirmations qui suivent sont toutes vraies, sauf une, laquelle?

1. Avec la ATmega324, on spécifie si on utilise le bit de parité ou non en ajustant un registre de configuration du USART avant de transmettre quoi que ce soit. • 2. Le USARTO du ATmega324 partage des registres de configuration avec le timer1. Certains modes de fonctionnement sont communs.

○ 5. Pour envoyer ou recevoir les données avec l'extérieur, un registre à l'intérieur du USARTO décale les bits reçus ou à envoyer un à un. 6. Je préfère ne pas répondre à la question pour éviter une perte de point. Les mécanismes d'interruption et de scrutation nécessitent plusieurs aspects pour se concrétiser. On peut affirmer que tous ces énoncés sont vrais, sauf

activités prévues au déroulement normal du programme.

En considérant l'architecture de la minuterie timer 1 suivante ?

Figure 40. 16-bit Timer/Counter Block Diagram<sup>(1)</sup>

Count

Clear

6. Je préfère ne pas répondre à la question pour éviter une perte de point.

4. On peut assez facilement transmettre des chaînes de caractères en RS232.

Clock Select Direction Edge Detector воттом TOP ( From Prescaler ) Timer/Counter **TCNTn** OCnA (Int.Req.) Generation OCnB (Int.Req.) TOP DATABUS Values Waveform - OCnB Generation ( From Analog

→ ICFn (Int.Reg.)

**TCCRnB** 

• 1. Le seul rôle utile d'une minuterie est de générer des interruptions quand on en a besoin.

4. Je préfère ne pas répondre à la question pour éviter une perte de point.

3. Je préfère ne pas répondre à la question pour éviter une perte de point.

Edge Detector

O 3. La valeur de BOTTOM est toujours zéro alors que la valeur de TOP peut prendre diverses valeurs possibles.

Git peut s'utiliser avec les commandes Linux de base. Une des affirmations suivantes <u>n'est pas vraie</u>, laquelle?

○ 6. La commande Linux « pwd » permet de savoir où je suis dans l'arborescence de répertoires sous Linux.

• 1. Un « git commit » rend mes modifications visibles directement à tous les gens ayant accès au répertoire Git commun.

○ 4. Un « git log LISEZMOI.txt» permet de voir les commentaires associés aux différentes versions (commit) du fichier LISEZMOI.txt.

○ 5. Après avoir propagé mes modifications avec Git, les autres peuvent faire un « git pull » pour inclure mes changements dans leur entrepôt.

Control Logic

○ 5. TCCRnA et TCCRnB, même si la représentation simplifiée de la figure ne le montre pas, ont beaucoup à voir avec ce qui se passe dans la logique qui contrôle l'ensemble des opérations. O 6. On change assez peu fréquemment les valeurs de OCCRnA et OCCRnB. Ce sont plus des constantes qu'on écrit dans ces registres en pratique.

options plus précises.

**TCCRnA** 

Un seul élément parmi les suivants est faux, lequel?

figure.

Il y a des principes généraux de programmation à savoir avec le contexte particulier de la programmation du ATmega324 avec SimulIDE. Toutes les affirmations suivantes sont vraies, sauf une, laquelle?

pour faciliter la programmation. ○ 6. PORTC est en fait un registre à une adresse précise dans l'architecture du ATmega324.

En considérant cette figure, toutes les affirmations suivantes sont fausses, sauf une qui est vraie, laquelle?

 4. On doit relier un bouton-poussoir entre Vo et la masse (ground). 0 5. Vo est le signal idéal pour servir d'interruption. On devrait le brancher au port D2. 6. Je préfère ne pas répondre à la question pour éviter une perte de point.

la souris pour des opérations qui reviennent fréquemment en codant. Question 16 Note de 1,00 sur 1,00 Marquer la question Considérant ce point

● 1. On peut utiliser la librairie de programmation C++ STL sans problème avec le ATmega324. 2. Ajuster un registre de configuration d'un périphérique revient souvent à assigner les nouvelles valeurs (souvent des 1) seulement aux bits pertinents afin de ne pas affecter les bits du registre qui activent d'autres opérations ou modes de fonctionnement. 3. Je préfère ne pas répondre à la question pour éviter une perte de point.

• 4. L'éditeur VS Code possède des raccourcis clavier (keyboard shortcuts) qui permettent de travailler plus rapidement sans toujours avoir recours à

○ 5. Les positions des bits d'un registre de configuration sont précisées dans la librairie AVRLibC et leur nom est en accord avec la documentation

Aller à...

O 3. Le point Vo doit être connecté à la référence analogique du ATmega324PA.

Obtenir l'app mobile

<u>Politiques</u>

 1. Le point Vo peut être connecté à l'une ou l'autre des broches du port A. ○ 2. On doit relier une diode électroluminescente (DEL- LED) entre Vo et la masse (ground).

Terminer la relecture ◄ Présentation de la société technique Poly Hx

Connecté sous le nom « Antonin Tritz » (Déconnexion) <u>INF1900 (010)</u>