

# Examen QCM – Développement des Applications Embarquées

## Réponses complètes avec justifications

---

### Question 1 : Avantage principal de l'Edge Computing

**Options :** A, B, C, D

**Réponse :** A

**Justification :** Selon les recherches sur l'Edge Computing, cette approche permet d'exécuter une intelligence embarquée directement sur les dispositifs avec une latence ultra-faible, une consommation énergétique réduite (milliwatts) et une diminution de l'utilisation de la bande passante. Elle élimine la nécessité de transmettre les données brutes vers le cloud. Des mesures montrent que 58 % des utilisateurs peuvent atteindre un serveur Edge en moins de 10 ms, contre seulement 29 % pour un serveur cloud proche. Cela confirme que l'Edge Computing permet un traitement local des données, réduisant la latence et la dépendance à Internet.

Les options B, C et D sont incorrectes.

---

### Question 2 : Rôle du microcontrôleur vs microprocesseur

**Options :** A, B, C, D

**Réponse :** A

**Justification :** Un microcontrôleur intègre sur une seule puce tous les éléments nécessaires à un système informatique : cœur processeur, mémoire RAM et mémoire programme (ROM/EEPROM). À l'inverse, un microprocesseur nécessite des mémoires externes connectées via un bus externe.

Les options B, C et D sont incorrectes.

---

### Question 3 : Importance de la phase de spécification

**Options :** A, B, C, D

**Réponse :** D

**Justification :** La phase de spécification permet de définir les exigences fonctionnelles du système, c'est-à-dire ce que le système doit faire avant de déterminer comment il sera réalisé. Les options A, B et C correspondent à des phases ultérieures.

---

#### Question 4 : Signification de Carry-Across et Carry-Over

**Options :** A, B, C, D

**Réponse :** A, B

**Justification :**

- **Carry-Across** : réutilisation d'un élément d'un système voisin ou similaire.
  - **Carry-Over** : réutilisation d'un élément d'une version précédente du système.
- Les options C et D inversent les définitions.
- 

#### Question 5 : Différence GPIO vs PWM sur Raspberry Pi

**Options :** A, B, C, D

**Réponse :** B

**Justification :** Les broches GPIO gèrent des signaux numériques (HIGH/LOW), tandis que le PWM permet de simuler un signal analogique en faisant varier le rapport cyclique d'un signal numérique. Cela permet de contrôler l'intensité d'une LED ou la vitesse d'un moteur.

---

#### Question 6 : Rôle de la broche GND sur Arduino

**Options :** A, B, C, D

**Réponse :** B

**Justification :** La broche GND représente la référence 0V et permet de fermer le circuit électrique en assurant le retour du courant.

---

#### Question 7 : Technique pour simuler un signal analogique

**Options :** A, B, C, D

**Réponse :** C (PWM)

**Justification :** Le PWM (Pulse Width Modulation) simule un signal analogique en modulant la largeur des impulsions d'un signal numérique.

---

#### Question 8 : Montage LED – Circuit correct ?

**Options :** A, B, C, D

**Réponse :** D

**Justification :** Le montage correct est : GPIO → LED → Résistance → GND. Si le GND est connecté directement à la LED sans résistance, le circuit est incorrect.

---

#### Question 9 : Montage LED clignotante

**Options :** A, B, C, D

**Réponse :** B

**Justification :** Une LED qui clignote rapidement est généralement contrôlée par un signal PWM à fréquence élevée.

---

### Question 10 : Rôle des broches numériques Arduino (0 à 13)

Options : A, B, C, D

Réponse : C

**Justification :** Les broches numériques permettent d'envoyer et de recevoir des signaux numériques via les fonctions `pinMode()`, `digitalWrite()` et `digitalRead()`.

---

### Question 11 : Fonction `pinMode()`

Options : A, B, C, D

Réponse : B

**Justification :** La fonction `pinMode(pin, mode)` configure une broche en entrée ou en sortie.

---

### Question 12 : Fonction `analogRead()` avec un potentiomètre

Options : A, B, C, D

Réponse : C

**Justification :** `analogRead()` lit une tension analogique et retourne une valeur entre 0 et 1023.

---

### Question 13 : Initialisation des broches LED RVB

Options : A, B, C, D

Réponse : B (`pinMode()`)

**Justification :** Les broches contrôlant une LED RVB doivent être configurées en sortie à l'aide de `pinMode()`.

---

### Question 14 : Broche A sur le montage Raspberry Pi Pico

Réponse : A (GND1)

---

### Question 15 : Broche B sur le montage

Réponse : B (GP15)

---

### Question 16 : Fonction de la broche Vsys

Réponse : D

**Justification :** Vsys est la ligne d'alimentation principale du Raspberry Pi Pico.

---

### Question 17 : Tension Vbus sur Raspberry Pi

Réponse : B

**Justification :** Vbus correspond à la tension USB, généralement 5V.

---

## Question 18 : Connexion des broches sur breadboard

**Réponse : B**

**Justification :** Les trous d'une même ligne de 5 sont connectés en parallèle.

---

## Question 19 : Utilisation des broches GPIO Raspberry Pi

**Réponse : D**

**Justification :** Les broches GPIO permettent la lecture et l'écriture de signaux numériques.

---

## Question 20 : Différence UART vs I2C

**Réponse : B**

**Justification :** UART est asynchrone, tandis que I2C est synchrone avec une ligne d'horloge.

## Tableau récapitulatif

Question	Bonne réponse	Concept clé
1	A	Edge Computing – faible latence
2	A	Microcontrôleur intégré
3	D	Spécification fonctionnelle
4	A, B	Carry-Across / Carry-Over
5	B	GPIO vs PWM
6	B	GND – référence 0V
7	C	PWM
8	D	Montage LED correct
9	B	LED et PWM
10	C	Broches numériques
11	B	pinMode()
12	C	analogRead()
13	B	Initialisation LED RVB
14	A	GND1 Pico
15	B	GP15
16	D	Vsys
17	B	Vbus USB
18	B	Breadboard parallèle
19	D	GPIO lecture/écriture
20	B	UART vs I2C