LO52 : Logiciels embarqués et communications courtes portées

Examen final automne 2012

Documents autorisés - Durée 2h

Rédigez la partie I sur la feuille de l'énoncé

Rédigez la partie IV sur une copie séparée.

Partie I : Communication courtes portées (5 points)

NOM	*PRENOM*

Un ordinateur portable reçoit les puissances de 9 BOX Wi-Fi 802.11g ci-dessous en dBm. Les BOX B, G et I sont en co-canal entre elles sur le canal 1. Les BOX A, C et H sont en co-canal entre elles sur le canal 6. Les BOX D, E et F sont en co-canal entre elles sur le canal 11. Les interférences en canal adjacent sont prises en compte à 70% du 1er au 4ème canal adjacent inclus.

A	B	C	D	E	F	G	H	I
-120	-84	-104	-83	-102	-95	-76	-80	-108

Pour initier un transfert sur une BOX l'ordinateur :

1. Doit recevoir une puissance supérieure à -85dBm.
2. Ne doit pas recevoir un autre signal co-canal supérieur de plus de 6dB.
3. Doit avoir un SINR supérieur à 9dB en co-canal pour un bruit N= - 120dBm

Quels sont les BOX qui remplissent le critère 1? Les critères 1 et 2? Les critères 1 et 2 et 3?

	Réponse
Critère 1	
Critère 1 et 2	
Critère 1, 2 et 3 **(justifier par le calcul)**	

Partie II : Optimisation de code (5 points)

EXERCICE 1 :

Soit le code intermédiaire à 3 adresses suivant :

(1) r1 = a
(2) r2 = b
(3) r3 = c
(4) r4 = r1 * r1
(5) r5 = r2 * r3
(6) r6 = r4 + r5
(7) r7 = r3 * r3
(8) r8 = r6 + r7
(9) r9 = r1 * r2
(10) r10 = r1 * r3
(11) r11 = r9 + r10
(12) r12 = r2 * r3
(13) r13 = r11 + r12
(14) r14 = r8 * r13

1. Donner le graphe d'interférence de ce code
2. Appliquer l'algorithme d'affectation de registres pour déterminer le nombre de registres physiques nécessaires pour l'exécution de ce code
3. Donner une solution pour économiser l'utilisation d'un registre physique

EXERCICE 2 :

Soit le programme suivant :

(1) y = x+2 (e1)
(2) z = x*4 (e2)
(3) d = z+x (e3)
(4) if (d<z) goto (9)
(5) x = y+3 (e4)
(6) y = z+x (e3)
(7) a = x*z (e6)
(8) goto (13)
(9) y = y+3 (e4)
(10) if (y<0) goto (13)
(11) x = x*z (e6)
(12) t = z+x (e3)
(13) y = y+3 (e4)
(14) b = z+x (e3)
(15) c = z+y (e5)

1. Donner le graphe de flot de contrôle de ce programme
2. Définir les ensembles des expressions produites, supprimées et disponibles en début et fin de bloc, pour chaque bloc du programme
3. Indiquer quelles sont les expressions qui peuvent être simplifiées et le gain en nombre d'instructions éliminées pour chacune d'elles.
4. Appliquer l'algorithme d'élimination des sous-expressions communes et donner le code résultant.

Partie III : Zygbee (5 points)

Question de cours

1. Expliquer la différence entre Zigbee, XBee et 802.15.4
2. Qu'est-ce que le ZigBee ?
3. Écrire les commandes AT nécessaires pour faire communiquer deux puces XBEE en P2P.

Exercice

On vous propose en utilisant le matériel de TP de construire une voiture télécommandée (Figure 1). Le châssis a déjà été construit et on se propose d'utiliser deux moteurs L293D. Un pour tourner les roues avant et l'autre pour faire avancer ou reculer la voiture. Pour faire fonctionner un moteur L293D, il faut 3 PINS:

- Enable 1 : Permet de gérer la puissance du moteur
- Input 1: Permet de tourner dans le sens anti-horaire
- Input 2 : Permet de tourner dans le sens horaire.

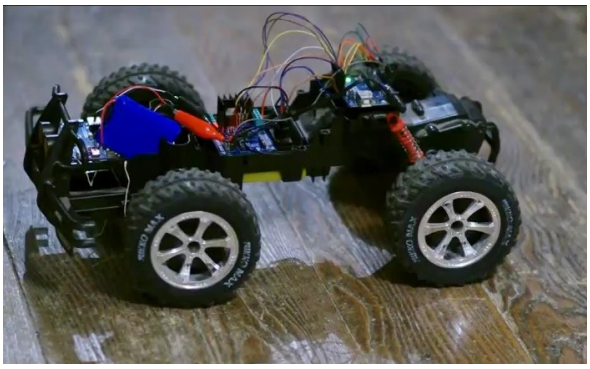

Figure 1. Voiture télécommandée

Voici un exemple de code permet de gérer le moteur du montage de la figure 2 :

```
int motor1Pin1 = 3; // pin 2 (Input 1) du L293D
int motor1Pin2 = 4; // pin 7 (Input 2) du L293D
int enablePin = 9;  // pin 1 (Enable 1) du L293D

void setup() {
  // set all the other pins you're using as outputs:
  pinMode(motor1Pin1, OUTPUT);
  pinMode(motor1Pin2, OUTPUT);
  pinMode(enablePin, OUTPUT);

  // Mettre la broche Enable a high comme ca le moteur tourne
  digitalWrite(enablePin, HIGH);
}

void loop() {
  // Le moteur tourne dans un sens
  digitalWrite(motor1Pin1, LOW);  // mettre pin 2 a 293D low
  digitalWrite(motor1Pin2, HIGH); // mettre pin 7 a L293D high

  delay( 3000 ); // Attendre 3 secondes

  // Le moteur tourne dans l'autre sens
  digitalWrite(motor1Pin1, HIGH); // Mettre pin 2 a L293D high
  digitalWrite(motor1Pin2, LOW); // Mettre pin 7 a L293D low

  delay( 3000 ); // Attendre 3 secondes
}
```

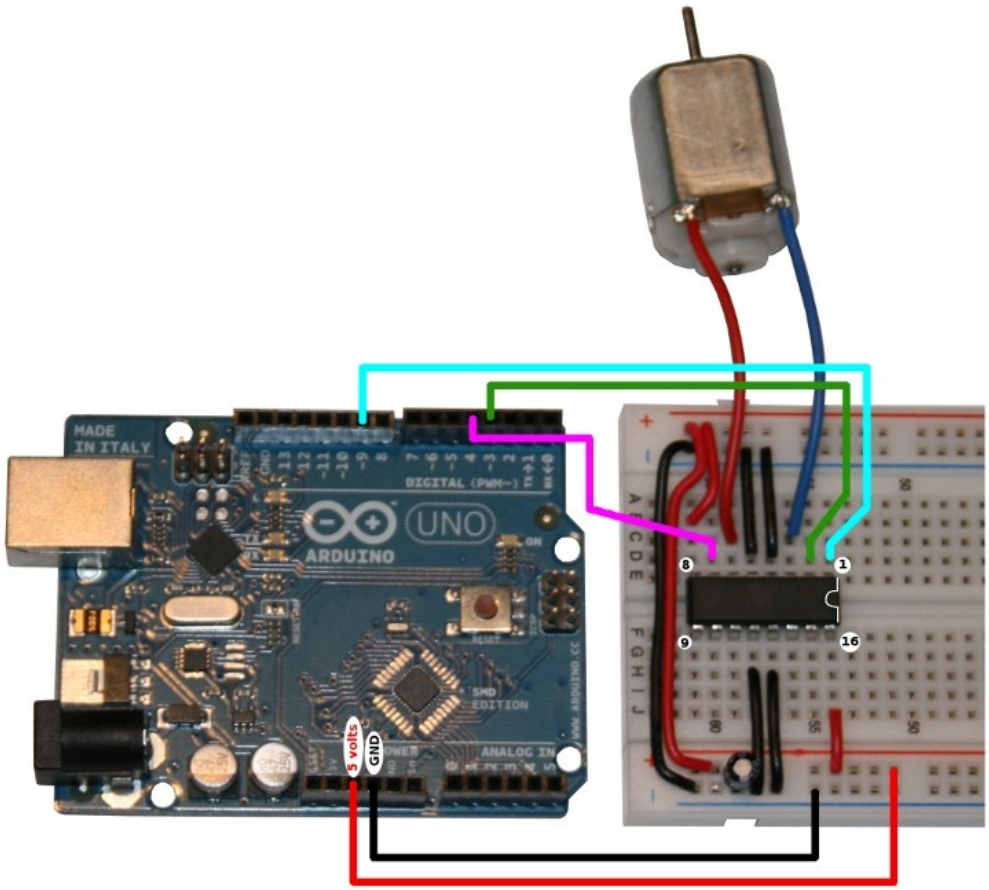

Figure 2. Exemple de montage du moteur sur l'arduino

Travail demandé :

1. Lister les composants nécessaires (On considérera le bloc moteur comme un composant) en utilisant les notations I0 à I5 et O0 à O5 pour les noms des pins.
2. Réaliser le schéma du montage et décrivez votre protocole de communication entre les deux arduinos .
3. Écrire le code de la télécommande .
4. Écrire le code de gestion des moteurs (coordinateur) .

Partie IV : Android (5 points)

Exercice 1 : QCM (2 points)

Pour cet exercice, il n'y a qu'une seule bonne réponse par question. Vous n'avez à indiquer que le choix final sans le texte. Exemple: Question 1 : A, Question 2 : B ...

1 – Quel est le nom de la **dernière version** d'Android ?

A – Ice Cream Sandwich
B – Jelly Bean
C – Jelly Beam
D – Bender

2 – On souhaite télécharger les sources de la version d'**Android 4.0.3 release 2.1**. Laquelle des propositions suivantes correspond aux deux commandes repo à exécuter ?

A – repo init -u https://android.googlesource.com/platform/manifest -b android-4.0.1_r1;repo sync

B – repo sync -u https://android.googlesource.com/platform/manifest -b android-4.0.3_r2.1;repo init

C – repo init -u https://android.googlesource.com/platform/manifest -b android-4.0.3_r2.1;repo sync

D – repo init -u https://android.googlesource.com/platform/manifest -b android-4.0.3_r2.1;repo download

3 – Quelle commande permet d'afficher l'historique d'un projet git ?

A – git history
B – git -h
C – git diff
D – git log

4 – Quelle commande permet de créer une branche git nommée lo52_final ?

A – repo branch lo52_final
B – repo start lo52_final
C – repo -b lo52_final
D – repo branch -c lo52_final

5 – Quelle commande fastboot permet de récupérer la valeur de la variable serialno ?

A – fastboot getvar serialno
B – fastboot get serialno
C – fastboot var serialno
D – fastboot -g serialno

6 – On veut afficher les logs de type **WARNING** de l'activité **NetworkController**. Laquelle des propositions suivantes correspond à la commande à exécuter ?

A – fastboot logcat -s NetworkController:W

B – adb logcat NetworkController:W

C – adb logcat -s NetworkController:I

D – adb logcat -s NetworkController:W

7 – Quelle commande permet de **choisir** le type de cible à produire lors d'une compilation d'Android ?

A – source build/envsetup.sh
B – make
C – mm
D – lunch

8 – Lors de l'ajout d'un composant à la plateforme Android, une variable doit être ajoutée/modifiée dans la définition du produit, laquelle ?

A – PRODUCT_PACKAGES
B – PRODUCT_PACKAGE

C – PRODUCT_BRAND D – DEVICE_PACKAGE_OVERLAYS

Exercice 2 : Questions de cours (2 points)

1 – Vous voulez ajouter une fonctionnalité X à votre noyau, configurable à travers une option. Quelles sont les différents **types de valeurs** que peut avoir votre option ?

2 – Quelle est la méthode de compilation utilisée pour produire un noyau pour plateformes embarquées ainsi que les deux variables nécessaires à ce type de compilation ?

3 – Vous venez de finir la configuration optimisée de votre noyau pour votre système Android lui permettant de démarrer le plus rapidement possible. Quelles commandes permettent de sauvegarder cette configuration en une configuration par défaut nommée **android_lo52_final_defconfig** ?

4 – Quelle est la différence entre un driver configuré en statique ou en module au niveau du noyau Linux ?

5 – Donnez 2 grandes modifications apportées par Android au noyau Linux et définissez les en une phrase.

Exercice 3 : (1 point)

Vous travaillez sur un projet appelé **lo52_final_panda** consistant en la création d'un nouveau « device » au sens Android, nommé **pandalo52final** et ayant pour vendeur **utbm**. Ce nouveau « device » doit hériter de la configuration déjà existante pour le produit panda. Décrivez les différentes étapes à suivre pour créer ce nouveau « device » et ainsi pouvoir y ajouter vos propres modifications par la suite.