

Sujet du TP 5

Conditions de réalisation :

- Utilisation de microvision5 pour 8051 et de la carte de développement 8051F020DK.
- Utilisation d'une application « Terminal de commande » telle que « Putty » pour établir une liaison entre un microcontrôleur et un PC au travers d'une liaison série type RS232
- Documentation indispensable : Poly Fiche Technique du 8051F020 (extraits).
- Documentations utiles : Cours BSE
- Le point de départ est un projet Microvision BSE_TP5 (téléchargeable sur le E-campus). Ce projet contient plusieurs fichiers source (4 fichiers dont `startup.A51`). Ce projet est 100% fonctionnel.

Etape de préparation et de compréhension.

1. Installer le kit matériel 8051F020 constitué de la carte 8051, de son boîtier de débogage et de son bloc alimentation
2. Télécharger le projet BSE_TP5, le désarchiver, le compiler et faire exécuter le code sur la carte. Ce code exécute les tâches suivantes (cette application correspond à un corrigé du TP4):
 - Clignotement régulier de la LED branchée sur P1.6.
 - Mise en œuvre d'une liaison série sur l'UART0 configurée à 115200Bd, 8bits de données, pas de parité, 1 stop bit
 - Au démarrage du programme le message « TP5 – System OK » est envoyé vers le terminal de commande.



Les signaux électriques échangés avec la carte 8051F020 sont des niveaux CMOS 0-3,3V au maximum! Utilisation du +5V interdite Ceci est valable aussi pour les broches véhiculant des signaux analogiques

Etape de compréhension 1 – Configuration de la tension de référence VREF interne

1. Configurer le périphérique « Voltage Reference » afin d'obtenir la tension de référence interne sur la broche VREF. Cette configuration sera codée dans une fonction `Voltage_Reference_Init()` (le squelette de la fonction est donné) .
2. Vérifier sur cette broche que vous obtenez bien la tension escomptée.

Etape de compréhension 2 – Configuration de l'ADC0 –Premiers essais de conversion A/N

1. Configurer l'ADC0 pour qu'il soit en mesure d'exécuter des conversions sur la voie AIN0.7 en unipolaire.
2. L'excursion en tension des signaux d'entrée sur le convertisseur sera de 0-1V, aussi on adaptera le gain de l'étage d'entrée du convertisseur en conséquence.
3. Attention à la configuration de l'horloge CLK SAR qui pilote le convertisseur. Sa fréquence ne doit pas dépasser une valeur limite indiquée dans la doc.
4. Les conversions dans cette étape seront déclenchées de manière logicielle et le résultat de conversion sera justifié à droite dans les registres ADC0H-ADC0L.
5. Cette configuration sera codée dans une fonction `ADC0_Init_Etape2()` (le squelette de la fonction est donné) .
6. La procédure de test sera la suivante : toutes les secondes, dans le `while(1)` du main, on déclenchera une conversion analogique-numérique du signal appliqué sur AIN0.7 (tension continue ou très lentement variable) et le résultat de la conversion sera transmis au terminal de commande sous la forme : « Mesure N – ADC0H = XX - ADC0L = XX » (N variable affichant le numéro de la mesure) .
7. Vérifier que les valeurs transmises correspondent bien aux niveaux de tensions appliqués sur AIN0.7
8. Améliorer votre code en calculant et en affichant la valeur de la tension convertie. Le message à envoyer sera donc du type : « Mesure N – ADC0H = XX - ADC0L = XX – Vin en mV : yyyy », comme représenté ci-dessous :

Mesure 5 -ADC0H:0x03 - ADC0L:0xf4 - Vin en mV: 300

Exemple du printf qui a permis l'affichage ci-dessus :

```
printf (" Mesure %5u -ADC0H:%#2.2bx - ADC0L:%#2.2bx - Vin en mV: %#2.2u \n\r", CP_Mesure, ADC0H, ADC0L, Value_mV);
```

Etape de compréhension 3 – Configuration du DAC0 –Premiers essais de conversion N/A.

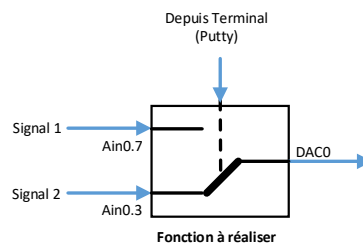
1. Configurer le DAC0 pour qu'il soit en mesure d'exécuter des conversions par logiciel (c'est l'écriture dans le registre DAC0H qui déclenche la conversion).
2. Cette configuration sera codée dans une fonction `DAC0_Init()` (le squelette de la fonction est donné) . Dans la configuration, on fera en sorte que les données 12 bits transmises au DAC soient les 12bits poids faibles du couple de registre DAC0H-DAC0L (calage à droite).
3. La procédure de test sera la suivante : toutes les secondes, dans le `while(1)` du main, on cherchera à produire une rampe de tension de 0V à Vref sur la broche de sortie DAC0 du convertisseur DAC0. Pour produire cette rampe, on appliquera successivement en entrée du DAC, toutes les valeurs possibles compte tenu de sa résolution (de 0 à $(2^{12}-1)$). La durée entre 2 conversions sera fixée à 10µs. A la fin de la rampe, on fera en sorte que la sortie DAC revienne à 0 volt.
4. Pour gérer la durée de 10µs, vous pourrez utiliser une fonction de temporisation `Delay_CLK22M_10micro` fournie dans le projet.
5. Visualisez la rampe sur l'oscilloscope. Vérifier sa linéarité.

Etape de compréhension 4 – Conversion A/N numérique périodique.


1. On souhaite désormais réaliser des conversions A/N périodiques.
2. Aussi, on codera une seconde fonction de configuration de l'ADC0 nommée `ADC0_Init_Etape4()`, pour que désormais la conversion soit déclenchée par un overflow du timer3, et d'autre part, la fin de conversion produira une interruption ADC0.
3. On souhaite avoir une fréquence d'échantillonnage de 10Khz. Aussi configurer le timer3 au travers de `Timer3_Init()` (le squelette de la fonction est donné) pour qu'il produise des overflows toutes les 100µs.
4. Coder la fonction d'interruption `ISR_ADC0`. Mettre en place un drapeau matériel sur P3.4 (VISU_INT_ADC0) qui permettra de visualiser l'exécution du programme d'interruption ADC0 et de mesurer sa récurrence.
5. Par ailleurs, pour vérifier que l'acquisition de signal (la voie Ain0.7) est correcte, on enverra sur la sortie DAC0, le résultat de conversion ADC0 obtenu à chaque interruption.
6. Vérifier sur l'oscilloscope que le signal injecté sur Ain0.7 (fréquence 100Hz) et le signal de sortie DAC0 (on travaille avec des signaux 0-1V d'excursion) sont quasiment identiques mais légèrement décalés temporellement. Mesurez ce décalage temporel et interprétez.
7. Pour un signal d'entrée à 1Khz sur Ain0.7, combien a-t-on de points de conversion sur une période (observation du signal DAC0) ?
8. Pour un signal de 19KHz en entrée, qu'observe-t-on en sortie du DAC0 ? Quelle est sa fréquence ? Conclure.

Etape de synthèse 5 – Sélection de voies analogiques.

1. Réaliser un sélecteur de voies analogiques pilotable depuis le terminal commande. Ainsi, 2 signaux analogiques différents (tension max 0-1V) seront appliqués sur les voies Ain0.7 et Ain0.3. Depuis le terminal de commande, on souhaite sélectionner quelle voie sera numérisée (conversion A/N) et ensuite envoyée sur DAC0. Un mode particulier permettra aussi d'envoyer alternativement sur DAC0 les conversions A/N faites sur Ain0.7 et Ain0.3.



2. Les 2 voies analogiques d'entrées sont les voies AIN0.7 et AIN0.3. Les niveaux de tension vont de 0 à 1V max.
3. La fréquence d'échantillonnage par voie est fixée à 10KHz.
4. Selon un ordre donné par le terminal de commande, on va recopier sur la sortie DAC0, soit l'entrée AIN0.7, soit l'entrée AIN0.3, soit recopier de manière alternée AIN0.7 et AIN0.3.
5. Les ordres transmis par la centrale de commande sont gérés de la manière suivante :
 - Le code est ajouté dans le `while(1)` du `main`
 - On scrute en permanence la réception d'un caractère.
 - Si un caractère (quelconque) est reçu, on envoie alors un message d'invite « Choix Voie : » afin de pouvoir changer de voie.
 - Si la réponse transmise est « 7 » alors on sélectionne la voie AIN0.7 pour la retransmettre sur DAC0 en continu (jusqu'au prochain changement de mode...).
 - Si la réponse transmise est « 3 » alors on sélectionne la voie AIN0.3 pour la retransmettre sur DAC0 en continu.
 - Si la réponse transmise est « 0 » alors on envoie sur DAC0 alternativement un échantillon du signal AIN0.7, puis un échantillon du signal AIN0.3 et ceci en continu.

	CPE Lyon – 4ETI Ver : 09/12/2022 14:09	Bases des systèmes embarqués
--	--	-------------------------------------

Validations - Sujet – TP5

Groupe :	Date :
-----------------	---------------

Etudiant1 :	Etudiant2 :
--------------------	--------------------

Etape Préparation	Validé par :	Heure :	Etape 1	Validé par :	Heure :
Validation 0 à 100%			Validation 0 à 100%		
Contrôle du fonctionnement de l'application transmise Commentaires éventuels :			Visualisation VREF Valeur de la tension Vref : _____ Commentaires éventuels :		

Etape 2	Validé par :	Heure :	Etape 3	Validé par :	Heure :
Validation 0 à 100%			Validation 0 à 100%		
Mise en œuvre de l'ADC0 : Valeur théorique ADC0H-ADC0L pour Vin = 1V : _____ Valeur théorique ADC0H-ADC0L pour Vin = 0V : _____ Valeur théorique ADC0H-ADC0L pour Vin = 0.75V : _____ Affichage ADC0L et ADC0H OK ? Affichage de la valeur mesurée en mV OK ? Commentaires éventuels :			Mise en œuvre du DAC0 Tension sur la sortie DAC0 pour DAC0H_DAC0L = 0xFFFF : _____ Tension sur la sortie DAC0 pour DAC0H_DAC0L = 0x3FFF : _____ Durée de la rampe ? _____ Valeur minimale de la rampe ? _____ Valeur maximale de la rampe ? _____ Commentaires éventuels :		

Etape 4	Validé par :	Heure :	Etape 5	Validé par :	Heure :
Validation 0 à 100%			Validation 0 à 100%		
Conversion A/N périodique Visu du drapeau matériel d'interruption ADC0 – OK ? Période mesurée des interruptions ADC0 : _____ Vérification : la sortie DAC0 « suit » le signal 100Hz injecté sur Ain0.7 – OK ? Pour un signal de 1Khz injecté sur Ain0.7, combien a-t-on de conversions A/N sur une période de ce signal ? _____ Pour un signal de 19Khz injecté sur Ain0.7, qu'observez-vous sur la sortie DAC ? quelle est sa fréquence ? Concluez.? _____ Commentaires éventuels :			Synthèse Visu affichage de l'invite « choix voie » à chaque appui sur une touche – OK ? Visu contrôle de la valeur transmise – OK ? Visu effective de la commutation des voies 3 et 7 – OK ? Visu effective du mode avec multiplexage des 2 voies - OK ? Commentaires éventuels :		

Synthèse	Validé par :	Heure :		Validé par :	Heure :
Validation 0 à 100%			Validation 0 à 100%		
Commentaires éventuels :			Commentaires éventuels :		

Contact : joly@cpe.fr