Entrées analogiques

(Réponses à ce document : <https://drive.google.com/file/d/0B-7bGF-nF0FYcGNqZU5DQWNpRWc/view>)

1.La description d'un signal analogique, la différence entre un signal analogique et un signal digital

2. La description un convertisseur AD

3. Documenter les termes suivants:

a. La résolution

b. La plage de conversion ou tension de pleine échelle

c. L'échantillonnage

4. Les caractéristiques des entrées analogiques de la carte Spider (Tension d'entrée min/max, nombres des bits,

nombre d'entrées analogiques, connecteurs et pattes utilisées, temps de conversion...)

5. La description d'un port AnalogInput en précisant la fonction des paramètres d'initialisation (constructeur), les

références à ajouter au projet C#.

6. La description détaillée des méthodes de l'objet AnalogInput permettant d'acquérir les données d'un capteur.

7. Les propriétés de l'objet AnalogInput.

8. La description des autres méthodes utilisables avec l'objet AnalogInput.

9. Fournir un tableau contenant les canaux analogiques, les connecteurs utilisables en entrées/sorties analogiques,

un exemple de déclaration et d'initialisation de toutes les entrées/sorties analogiques disponibles

10. Un exemple de programme permettant de lire et d'afficher des valeurs analogique à l'aide de la carte spider

1. Les signaux analogiques utilisent les valeurs du signal, tandis que les signaux digitaux lisent uniquement le binaire.
2. Les convertisseurs AD convertissent une grandeur analogique en une grandeur numérique.  
   Le convertisseur à simple rampe marche comme suit :  
   Il récupère un signal analogique et mesure si il dépasse certaines valeurs définies, lorsqu’il dépasse ces valeurs il incrémente le chiffre binaire, ce qui permet de convertir petit à petit les valeurs en binaire.
3. **Définitions** :

**la résolution** : La résolution est définie par le nombre de bits dans le convertisseur plus il y en a plus la mesure convertie est précise.

**la plage de conversion**: Zone occupée en volt par le signal analogique.

**l’échantillonnage**: La fréquence de la mesure (analyse) d’un signal.

4) Pour l’AnalogInput nous avons 0-3.3v, pin 3-4-5, dans le branchement 9 et 10 avec une résolution de 12 bits.  
Pour ce qui est du AnalogOutput nous avons de 0-3.3v, pin 5, dans le branchement 9, avec une résolution de 10 bits.  
  
(<https://drive.google.com/a/eduge.ch/file/d/0B0H7gtlCU6KaTnhKZXJqNkY2czQ/view?usp=sharing>)

Ces circuits analogiques intégrés ne sont pas conçus pour être très précis. Pour une précision plus élevée, un ADC externe peut être ajouté en utilisant le bus SPI.

5) Référence à ajouter pour l’AnalogInput: using Microsoft.SPOT.Hardware;

Un analog input est une entrée analogique convertie au niveau de tension en une valeur numérique qui peut être stockée et traitée dans un ordinateur.  
  
Exemple d’initialisation d’AnalogInput:   
AnalogInput joyy = new AnalogInput(FEZSpiderII.Socket10.AnalogInput5);

6) Pour récupérer les données d’un AnalogInput nous utilisons la Méthode“.ReadRaw()”ou “ .Read()”. A savoir que la méthode “.ReadRaw()” sera plus rapide

7)

* **Offset** : retourne ou définit le décalage à appliquer à la lecture brute du capteur avant que la valeur soit retournée ( double)
* **Pin**: Retourne la broche GPIO du canal.
* **Precision**: Retourne la précision en bits du canal (Int).
* **Scale**: Retourne ou définit le facteur multiplicatif appliqué à la lecture brute du capteur avant que la valeur soit retournée (double). Valeur par défaut = 1.

(<https://msdn.microsoft.com/en-us/library/microsoft.spot.hardware.analoginput(v=vs.102).aspx>)

8)

**pinMode()**: Configure le pin spécifié pour se comporter en entrée ou en sortie.

**digitalWrite()**: Écrit une valeur Haute ou Bas à une broche numérique.

**delay()**: Met en pause le programme pour le temps (en millisecondes) spécifié comme paramètre.

**AnalogInOutSerial**: Lis une broche d’entrée analogique puis ces données pour diminuer ou illuminer une LED.

**AnalogWriteMega**: fait allumer et éteindre les LEDs un par un.

**Calibration**: On peut définir un maximum et un minimum pour les valeurs des capteurs analogiques attendus.

**Fading**: Utilise une sortie analogique pour estomper la LED.

**Smoothing**: Lisse les lectures multiples d’une entrée analogique.

9) tableau fournis par le prof.:

<https://drive.google.com/file/d/0B-7bGF-nF0FYV0Z4cU5SSnhpVXM/view>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  |  |  |  |  |
|  |  |  |  |  |

10) **exemple de programme:** <https://drive.google.com/a/eduge.ch/file/d/0B_q-2T0YFLLINDJrZVdDYmZWSnM/view?usp=sharing>

**cartouche (à remplir) et usings:**

/\* Programme NomProgramme

\* Nom (seulement les consonnes) prénom date Version NumVersion

\* Description ...

\*/

/\* Description matériel et librairies:

\*

\* Version: GHI SDK:2016.1.2.0; NETMF:4.3.8.1; Gadgeteer core:2.44.1100.0

\* Carte mère: Spider 2 Version firmware: 4.3.8.1 Version Loader:4.3.8.1 (MAJ: 4381)

\* Module utilisé Connecteur utilisé Fonction

\* usbClientDP 1 Connexion au PC

\* Button 2 Start/Stop

\* Joystick 1 "Mooving" LED in the LED strip

\* LED strip 1 Show the action of the Joystick

\*/

/\* Référence à ajouter:

\* GHI.Pins pour GHI.Pins.EMX ou ...

\* Microsoft.SPOT.Hardware pour Cpu.Pin et OutputPort

\*/

// Par défaut

using System;

using Microsoft.SPOT;

// A ajouter

using Microsoft.SPOT.Hardware; // Pour OutputPort

using GHI.Pins; // Pour GHI.Pins

using System.Threading; // Pour Thread.Sleep(1000)

using GHICard = GHI.Pins.FEZSpider; // Défini la carte principale utilisée

**exemples déclarations et utilisations :**

OutputPort LedSpider = new OutputPort(FEZSpiderII.DebugLed, false);

OutputPort LedBtn1 = new OutputPort(FEZSpiderII.Socket12.Pin4, false);

OutputPort LedBtn2 = new OutputPort(FEZSpiderII.Socket14.Pin4, false);

OutputPort LedLigne0 = new OutputPort(FEZSpiderII.Socket8.Pin3,

InputPort btn1 = new InputPort(G120.P2\_13, true, Port.ResistorMode.Disabled);

InputPort btn2 = new InputPort(FEZSpiderII.Socket14.Pin3, true, Port.ResistorMode.Disabled);

InputPort btnJoy = new InputPort(FEZSpiderII.Socket10.Pin3, true, Port.ResistorMode.Disabled);

AnalogInput joyx = new AnalogInput(FEZSpiderII.Socket10.AnalogInput4);

AnalogInput joyy = new AnalogInput(FEZSpiderII.Socket10.AnalogInput5);

AnalogInput lightSense = new AnalogInput(FEZSpiderII.Socket9.AnalogInput3);

AnalogInput entreeAnalogS10p3 = new AnalogInput(FEZSpiderII.Socket10.AnalogInput3);  
AnalogInput entreeAnalogS10p4 = new AnalogInput(FEZSpiderII.Socket10.AnalogInput4);  
AnalogInput entreeAnalogS10p5 = new AnalogInput(FEZSpiderII.Socket10.AnalogInput5);  
AnalogInput entreeAnalogS9p3 = new AnalogInput(FEZSpiderII.Socket9.AnalogInput3);  
AnalogInput entreeAnalogS9p4 = new AnalogInput(FEZSpiderII.Socket9.AnalogInput4);  
AnalogInput entreeAnalogS9p5 = new AnalogInput(FEZSpiderII.Socket9.AnalogInput5);  
AnalogOutput sortieAnalogS9p5 = new AnalogOutput(FEZSpiderII.Socket9.AnalogOutput)

intx = joyx.ReadRaw();

inty = joyy.ReadRaw();

light = lightSense.ReadRaw();