

Version	Date	Rédacteur	Modifications / Commentaires
1.0	25/05/12	DARRAS L.	Création du document

Table des matières

I.	Introduction et contexte.....	2
II.	Schéma général	2
A.	Partie capteurs.....	2
B.	Partie alimentation	2
III.	Circuit imprimé et boîtier	3
IV.	Exemples d'intégration de cette carte dans un système	3
A.	Exemple 1	3
B.	Exemple 2	4
V.	Applications	5
A.	Avec enregistreur d'acquisition Controlord Gigalog S :	5
B.	Avec enregistreur d'acquisition Campbell CR1000 :	5
VI.	Conclusion.....	5
VII.	Annexes	6
A.	Schéma carte électronique :	6
B.	Programme CR1000 :	7

I. Introduction et contexte

Le besoin était de pouvoir enregistrer des courants de consommation d'un matériel, ou des courants de charge d'une batterie.

Aucun datalogger ne permet d'enregistrer ce type de mesure (courant), car en général, ce sont des tensions et températures que ces dataloggers permettent de mesurer.

L'idée de ce projet est donc de réaliser une carte électronique permettant de convertir un courant en tension pour pouvoir l'enregistrer dans un datalogger.

De plus, l'idée est de :

- pouvoir convertir 4 courants sur une même carte électronique.
- pouvoir placer cette carte électronique dans un petit boîtier pour placer sur un rail-din.
- Pouvoir alimenter cette carte en 12V non régulé issu d'une batterie, ou en 5V régulé.

II. Schéma général

Dans cette carte électronique, il y a 1 partie alimentation, et 1 partie capteurs.

A. Partie capteurs

Un capteur de courant Allegro ACS712 ou ACS714 a été trouvé et semble correspondre à nos besoins.

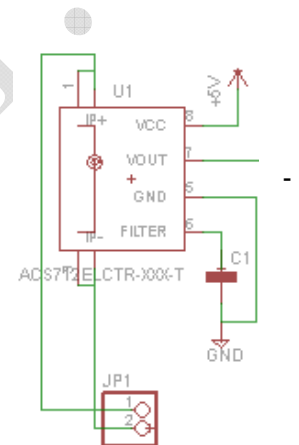
Il s'agit d'un capteur de courant pouvant mesurer plusieurs plages de courant :
5/+5A, -20/+20A, -30/+30A.

Ce capteur doit être alimenté en 5V, et sa plage de sortie est comprise entre :

- o Pour ACS712-5A: 1.5V(-5A) et 3.5V(+5A).
- o Pour ACS712-20A: 0.5V(-20A) et 4.5V(+20A).
- o Pour ACS712-30A: 0.5V(-30A) et 4.5V(+30A).

Il permet de mesurer du courant continu ou alternatif, et sa plage de température est -40/+150°C.

C1, C2, C3, C4, sont des capacités de filtrage, et peuvent valoir 1µF par exemple.



B. Partie alimentation

Dans la partie alimentation, on doit pouvoir, soit alimenter cette carte directement à partir d'une batterie 12V, soit à partir d'une tension 5V régulée.

Alimenter directement en 5V peut permettre :

- o D'utiliser une alimentation 5V plus stabilisée que celle issue du régulateur prévu sur la carte.
- o D'utiliser une alimentation 5V issue du régulateur d'une autre carte, permettant ainsi la cascade de plusieurs cartes électroniques de capteurs de courant, avec une seule régulation 12V/5V de référence.

Par contre, si aucune tension 5V régulée n'est disponible, il faut utiliser directement la tension 12V d'une batterie pour la convertir en 5V régulé :

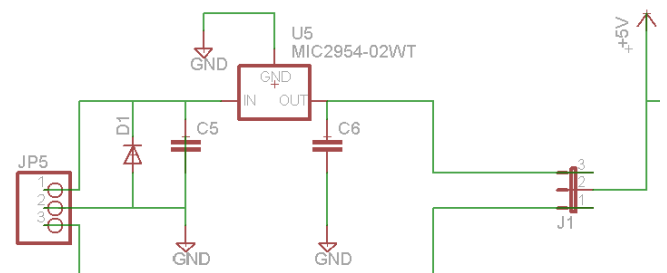
Le régulateur 12V/5V utilisé est le Micrel MIC2954.

Connecteur JP5 :

- o Broche 1 : +12 -> 30V.
- o Broche 2 : GND.
- o Broche 3 : +5V.

La diode D1 est une diode TransZorb afin d'éviter les surtensions.

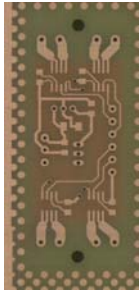
La capacité C5 n'est pas indispensable, et la capacité C6 sert de filtrage et peut valoir 2.2µF par exemple.



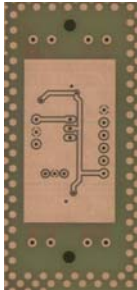
Le support de cavalier J1 nécessite un cavalier pour alimenter le +5V de la carte, soit par le 12V d'une batterie et du régulateur 12V/5V, soit par le +5V d'alimentation d'entrée de la carte.

III. Circuit imprimé et boîtier

Le routage de la carte a été effectué via le logiciel Eagle (gratuit pour cette taille réduite de cartes en 2 couches) :



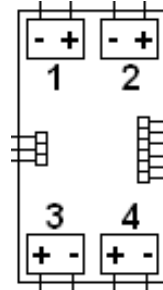
Top



Bottom



photo carte + boîtier rail-din



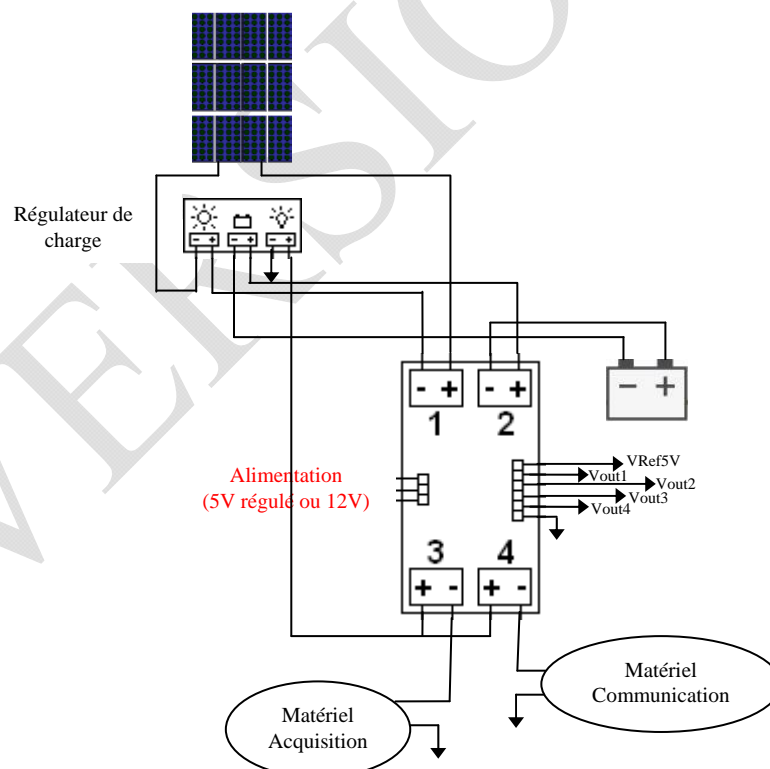
La taille de la carte est de 70mm*27mm. Cette carte peut être fixée :

- par vis via les 2 trous représentés en noir sur les photos ci-dessus.
- dans un boîtier pouvant être installé sur Rail-din :

IV. Exemples d'intégration de cette carte dans un système

A. Exemple 1

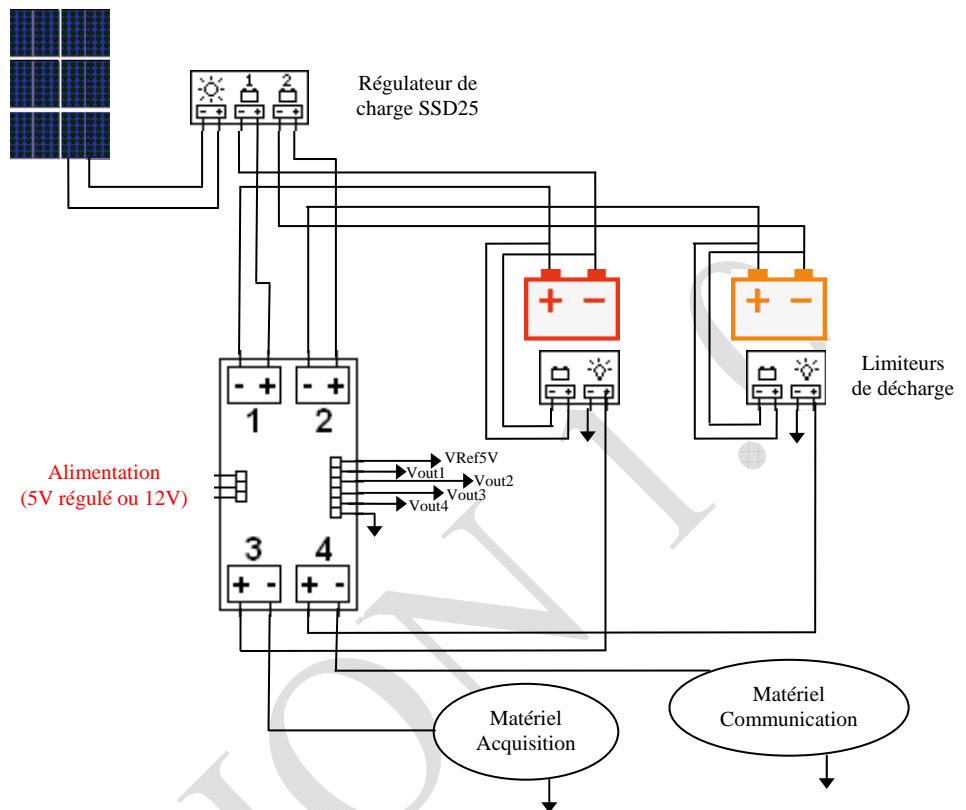
Exemple d'une alimentation solaire simple avec charge d'une seule batterie.



N.B. : en cas d'utilisation d'une pile à combustible ou d'une éolienne en complément de l'énergie solaire, il est possible de ne pas enregistrer le courant entre les panneaux solaires et le régulateur, et plutôt utiliser cette entrée 1 pour enregistrer le courant de charge de la batterie issue de cette 2^{ème} source.

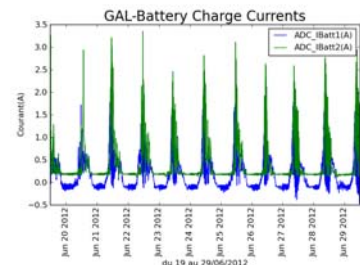
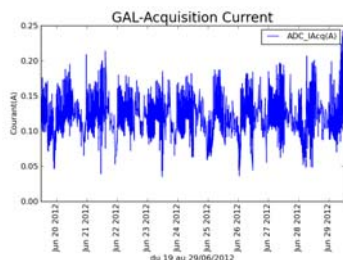
B. Exemple 2

Exemple d'une alimentation solaire avec charge de 2 batteries distinctes pour la partie « Acquisition » et la partie « Communication »..



Une telle installation permet ainsi ensuite un affichage dans le temps des courbes de courants mesurés :

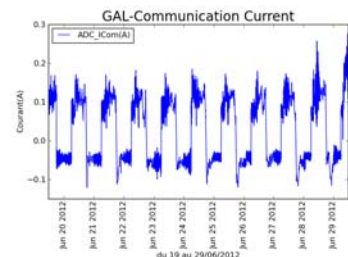
Courants de charge des batteries (Mesure courants entrées 1&2)->



<- Courant de consommation du matériel d'acquisition (Mesure courant entrée 3)

Courant de consommation du matériel de communication (Mesure courant entrée 4) :

N.B. : pour la courbe affichée, le matériel de communication n'est en fait alimenté que dans une plage horaire fixe afin de limiter la Consommation ->



N.B. :

Pour les entrées 1 et 2, le composant utilisé est le ACS712-20A pour enregistrer un courant de charge de forte intensité. Vout1 et Vout2 peuvent varier de 0.5V à 4.5V pour représenter la plage -20A à +20A.

Pour les entrées 3 et 4, le composant utilisé est le ACS712-5A pour enregistrer un simple courant de consommation. Vout3 et Vout4 peuvent varier de 1.5V à 3.5V pour représenter la plage -5A à +5A.

V. Applications

A. Avec enregistreur d'acquisition Controlord Gigalog S :

Les 4 sorties des capteurs doivent être connectées à une entrée analogique **aX** du gigalog configurée pour lire une tension de 0 à 5V.

Si le capteur de courant utilisé correspond à la plage de mesure -5A/+5A, la tension 1.5V correspond à un courant de -5A, et 3.5V correspond à une tension de +5A.

C'est la valeur réelle en tension qui est enregistrée à la fréquence sélectionnée.

Afin d'enregistrer également les valeurs réelles en courant, il est possible de sélectionner une entrée virtuelle **aY** (avec $Y > 15$) :

Si on considère **aR** l'entrée analogique de la tension 5V de référence, et **aX** l'entrée analogique de la tension

L'expression de l'entrée virtuelle doit être configurée à :

- Pour capteur courant -5A/+5A : $aY = (2 \cdot aX - aR) \cdot 5/2$.
- Pour capteur courant -201/+20A : $aY = (2 \cdot aX - aR) \cdot 5$.

Et le calcul : 1000 fois, et une précision de #.###.

Le résultat de l'entrée est exprimé ainsi en Ampères.

B. Avec enregistreur d'acquisition Campbell CR1000 :

Cet enregistreur dispose d'une sortie 5V régulée. C'est donc avec cette tension que sera alimentée la carte électronique.

5 entrées analogiques simples seront utilisées : 1 pour relire la tension 5V de référence pour s'assurer de la tension de référence, et 4 pour lire les tensions de sortie des 4 capteurs de courant sur la carte électronique.

Le programme à charger dans le datalogger CR1000 est décrit en Annexe.

Il permet ainsi d'afficher les mesures ainsi -> :

RecNum	241
TimeStamp	12 11:29:45
Batt_Volt_Avg	13,70
Ref5V_mVolt_Avg	5 008,00
CurSensor1_mVolt	2 516,00
CurSensor2_mVolt	2 520,00
CurSensor3_mVolt	2 503,00
CurSensor4_mVolt	2 502,00
CurSensor1_mAmp	1,31
CurSensor2_mAmp	2,48
CurSensor3_mAmp	0,08
CurSensor4_mAmp	0,00

Et de créer une table suivante :

```
"TOA5", "CR1000_LABO", "CR1000", "E4668", "CR1000.Std.24", "CPU:CR1000_LABO.CR1", "2993", "Table1"
"TIMESTAMP", "RECORD", "Batt_Volt_Avg", "Ref5V_mVolt_Avg", "CurSensor1_mVolt_Avg", "CurSensor2_mVolt_Avg", "CurSensor3_mVolt_Avg", "CurSensor4_mVolt_Avg", "CurSensor1_mAmp_Avg", "CurSensor2_mAmp_Avg", "CurSensor3_mAmp_Avg", "CurSensor4_mAmp_Avg"
"TS", "RN", "Volts", "Volts", "mVolts", "mVolts", "mVolts", "mVolts", "mA", "mA", "mA", "mA"
" ", " ", "Avg", "Avg", "Avg", "Avg", "Avg", "Avg", "Avg", "Avg", "Avg", "Avg"
"2012-05-23 11:26:00", 0, 13.7, 5008, 2516, 2520, 2503, 2502, -4.228, -2.724, -1.372, 0.514
"2012-05-23 11:27:00", 1, 13.7, 5008, 2516, 2520, 2503, 2502, 2.428, 2.701, 0.305, 0.836
"2012-05-23 11:28:00", 2, 13.7, 5008, 2516, 2520, 2503, 2502, -0.816, 3.931, 0.781, 0.025
"2012-05-23 11:29:00", 3, 13.7, 5008, 2516, 2520, 2503, 2502, 1.309, 2.477, 0.082, -0.003
"2012-05-23 11:30:00", 4, 13.7, 5008, 2516, 2520, 2503, 2502, 1.086, 1.806, 0.305, 0.668
"2012-05-23 11:31:00", 5, 13.7, 5008, 2516, 2520, 2503, 2502, 0.638, -0.431, 0.138, 0.668
"2012-05-23 11:32:00", 6, 13.7, 5008, 2516, 2520, 2503, 2501, 0.414, 2.925, 1.088, -1.401
```

VI. Conclusion

Lorsque ce capteur devra être alimenté par une tension autre que du 5V régulé, la tension de référence 5V issue du régulateur Micrel MIC2954 est stable, mais peut varier entre 4.7V et 5V selon les cartes électroniques.

Il sera peut être préférable dans une version ultérieure de la carte, de disposer d'un potentiomètre afin d'ajuster sur chaque carte la tension de référence à 5V avec le plus de précision possible.

A. Schéma carte électronique :



B. Programme CR1000 :

```
'CR1000
'Created by CRBasic Editor (2.8)

'Declare Currents Multiplier Constants
' for -20A/+20A sensors, the voltage range is 0.5V/4.5V, so 2V represents 20A, and so
the multiplier is : 10.
' for -5A/+5A sensors, the voltage range is 1.5V/3.5V, so 1V represents 5A, and so
the multiplier is : 5.
Const CurSensor1MultVoltToAmp = 10
Const CurSensor2MultVoltToAmp = 10
Const CurSensor3MultVoltToAmp = 5
Const CurSensor4MultVoltToAmp = 5

'Declare Currents Offset Constants
Const CurSensor1Offset_mAmp = 0 ' offset à définir apres test avec courant nul
Const CurSensor2Offset_mAmp = 0 ' offset à définir apres test avec courant nul
Const CurSensor3Offset_mAmp = 0 ' offset à définir apres test avec courant nul
Const CurSensor4Offset_mAmp = 0 ' offset à définir apres test avec courant nul

'Declare battery voltage measure
Public Batt_Volt ' tension batterie 12V, en V
'Declare currents sensor measures
Public Ref5V_mVolt ' tension référence 5V, en mV
Public CurSensor1_mVolt ' tension sortie capteur 1 entre 0 et 5V, en mV
Public CurSensor1_mAmp ' courant capteur 1 calculé, en mA
Public CurSensor2_mVolt ' tension sortie capteur 2 entre 0 et 5V, en mV
Public CurSensor2_mAmp ' courant capteur 2 calculé, en mA
Public CurSensor3_mVolt ' tension sortie capteur 3 entre 0 et 5V, en mV
Public CurSensor3_mAmp ' courant capteur 3 calculé, en mA
Public CurSensor4_mVolt ' tension sortie capteur 4 entre 0 et 5V, en mV
Public CurSensor4_mAmp ' courant capteur 4 calculé, en mA

.....

' Units des valeurs mesures apparaissant dans le fichier de sortie
Units Batt_Volt=Volts
Units Ref5V_mVolt=Volts
Units CurSensor1_mVolt=mVolts
Units CurSensor2_mVolt=mVolts
Units CurSensor3_mVolt=mVolts
Units CurSensor4_mVolt=mVolts
Units CurSensor1_mAmp=mA
Units CurSensor2_mAmp=mA
Units CurSensor3_mAmp=mA
Units CurSensor4_mAmp=mA

'Define Data Tables - definition des valeurs a enregistrer dans fichier Table1
DataTable(Table1,True,-1)
DataInterval(0,1,Min,1) ' intervalle d'enregistrement des donnees, 1min
Average(1,Batt_Volt,FP2,False)
Average(1,Ref5V_mVolt,FP2,False)
Average(1,CurSensor1_mVolt,FP2,False)
Average(1,CurSensor2_mVolt,FP2,False)
Average(1,CurSensor3_mVolt,FP2,False)
Average(1,CurSensor4_mVolt,FP2,False)
Average(1,CurSensor1_mAmp,FP2,False)
Average(1,CurSensor2_mAmp,FP2,False)
Average(1,CurSensor3_mAmp,FP2,False)
Average(1,CurSensor4_mAmp,FP2,False)
```

EndTable

'Main Program

BeginProg

SDMSpeed(50)

Scan(1,Sec,1,0) ' Definition frequence d'acquisition

'Default Datalogger Battery Voltage measurement Batt_Volt:
Battery(Batt_Volt)

'5V Ref measurement Ref5V_Volt:
VoltSe(Ref5V_mVolt,1,mV5000,1,True,0,_60Hz,1,0)

'Current Sensor 1 measurement CurSensor1_Volt
VoltSe(CurSensor1_mVolt,1,mV5000,2,False,0,_60Hz,1,0)

'Current Sensor 1 Conversion in mA
CurSensor1_mAmp = CurSensor1Offset_mAmp + (CurSensor1_mVolt - (Ref5V_mVolt/2))*CurSensor1Range_Amp

'Current Sensor 2 measurement CurSensor1_Volt
VoltSe(CurSensor2_mVolt,1,mV5000,3,True,0,_60Hz,1,0)

'Current Sensor 2 Conversion in mA
CurSensor2_mAmp = CurSensor2Offset_mAmp + (CurSensor2_mVolt - (Ref5V_mVolt/2))*CurSensor2Range_Amp

'Current Sensor 3 measurement CurSensor3_Volt
VoltSe(CurSensor3_mVolt,1,mV5000,4,True,0,_60Hz,1,0)

'Current Sensor 3 Conversion in mA
CurSensor3_mAmp = CurSensor3Offset_mAmp + (CurSensor3_mVolt - (Ref5V_mVolt/2))*CurSensor3Range_Amp

'Current Sensor 4 measurement CurSensor4_Volt
VoltSe(CurSensor4_mVolt,1,mV5000,5,True,0,_60Hz,1,0)

'Current Sensor 4 Conversion in mA
CurSensor4_mAmp = CurSensor4Offset_mAmp + (CurSensor4_mVolt - (Ref5V_mVolt/2))*CurSensor4Range_Amp

'Call Data Tables and Store Data
CallTable(Table1)
PortsConfig(&B000000111,&B000000000)
NextScan

EndProg