
UNIVERSITE PARIS 13

I.U.T. DE VILLETANEUSE

DÉPARTEMENT G.E.I.I.

12/09/2022 – 05/12/2022

JAF AIS Noumane

HECHMI Hazem

KONE Issiaka

BUT GEII TP B1

PROJET AGS



SOMMAIRE

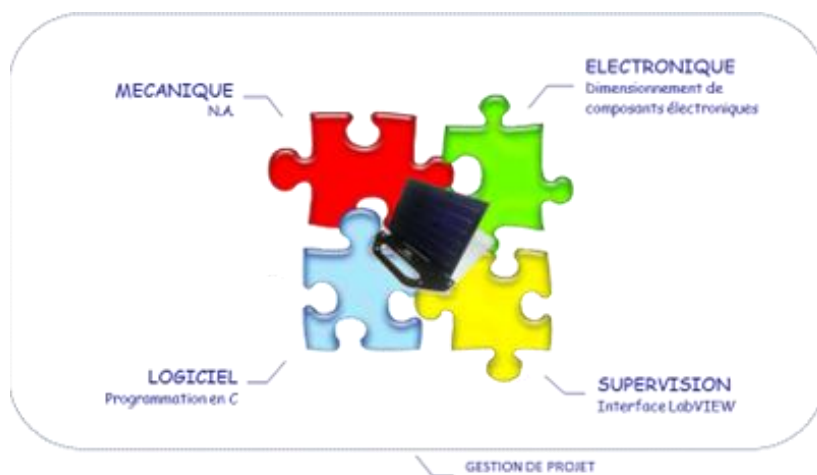
Introduction	1
1. Organisation du projet.....	2
2. Étude de l'interface de supervision.....	3
3. Étude de la carte µContrôleur et du logiciel embarqué	4
3.1 Étude de la carte électronique.....	4
3.2 Environnement de développement logiciel MPLAB X.....	6
4. Étude de la carte d'acquisitions analogiques	9
4.1 Étude du filtre d'entrée d'acquisition analogique tension PV.....	9
4.2 Étude de la résistance de charge du panneau solaire.....	14
4.3 Etude de la sonde de température	15
4.4 Fabrication de la carte électronique analogique	16
Conclusion	16
Annexe.....	17

Introduction

L'analyseur de gisement solaire est un outil d'aide à la décision. Il permet de relever en temps réel les informations essentielles.

Notre objectif : Réaliser une étude et fabriquer un démonstrateur photovoltaïque
Notre projet sera divisé en 3 parties :

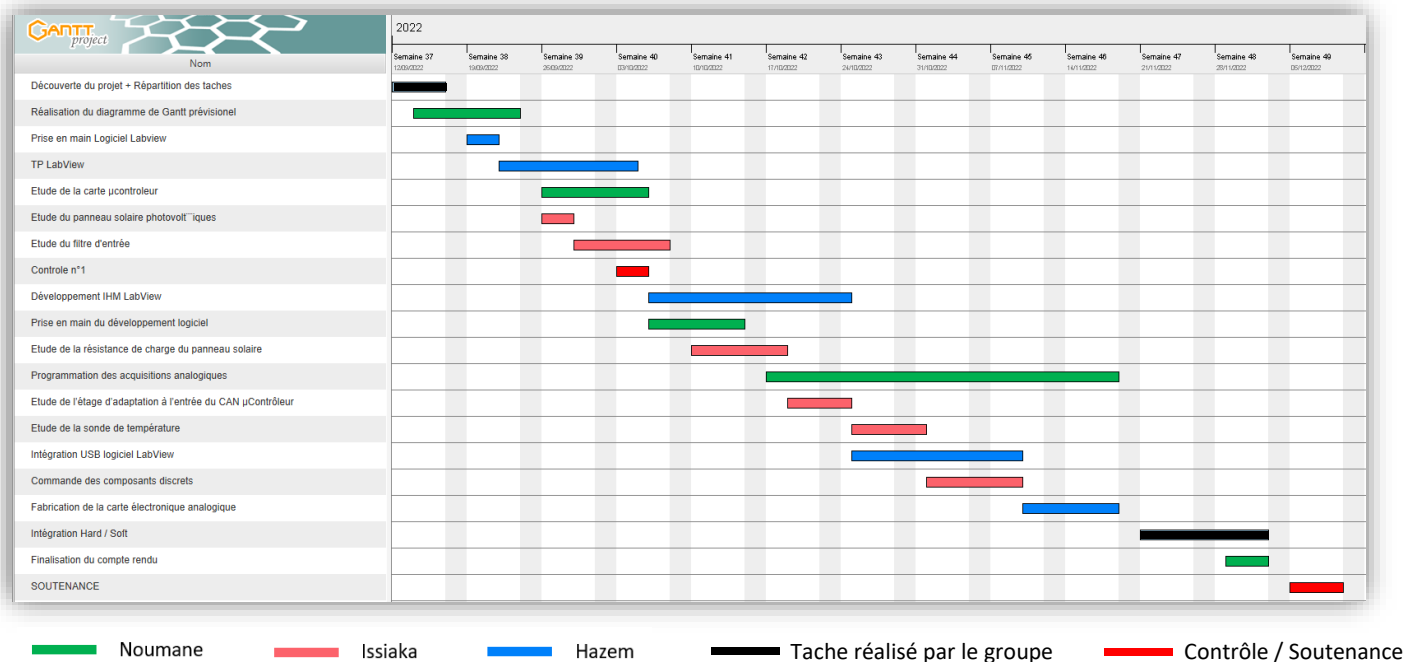
- Réalisation de l'interface de supervision LabView
- Réalisation de la carte µContrôleur et du logiciel embarqué MPLAB X
- Réalisation d'une carte électronique d'acquisition analogiques



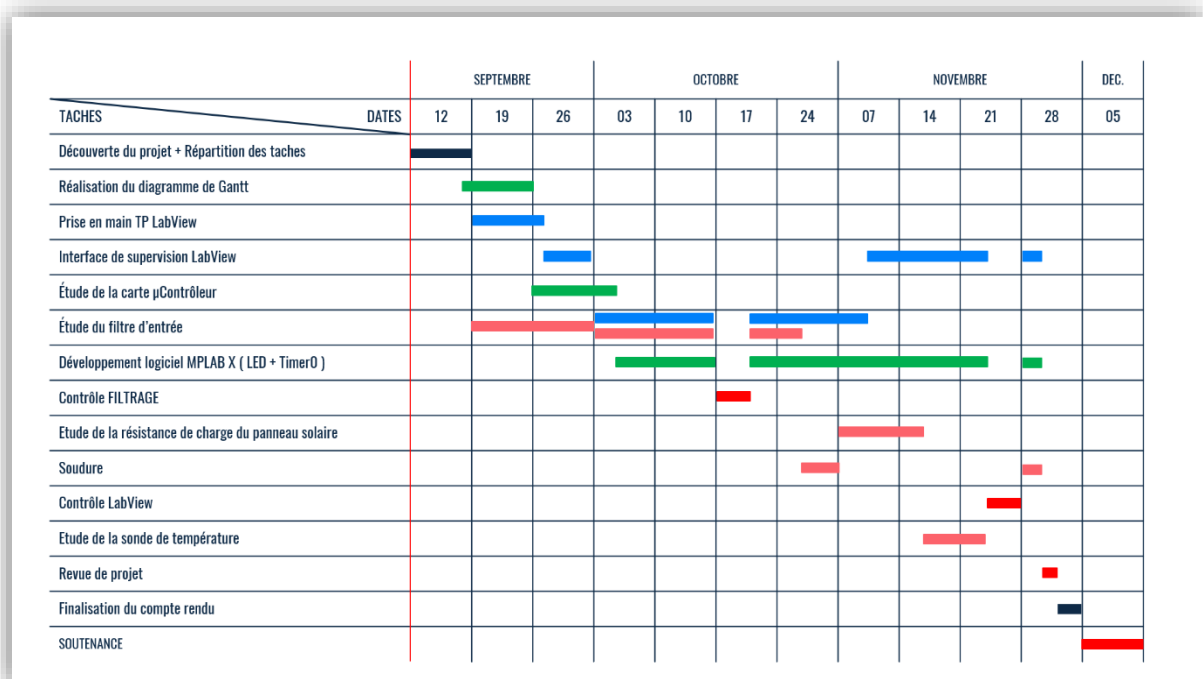
1. Organisation du projet

Pour la réalisation de l'analyseur de gisement solaire, nous avons au total 45 heures décomposés en 9 séances de 4 heures par semaine. La 10^{ème} séance sera consacrée à la soutenance. Au cours du projet, nous aurons 2 contrôles (filtrage et LabView) et une revue du projet. Notre groupe sera composé de 3 étudiants : Noumane JAF AIS, Issiaka KONE et Hazem HEHCMI.

Nous réalisons le diagramme de Gantt prévisionnel.

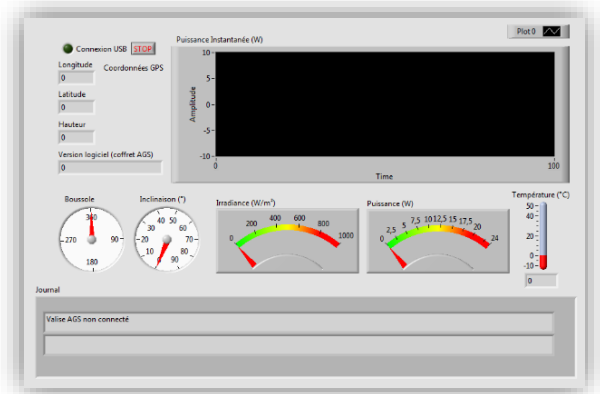


Voici le diagramme de Gantt des tâches réalisée :

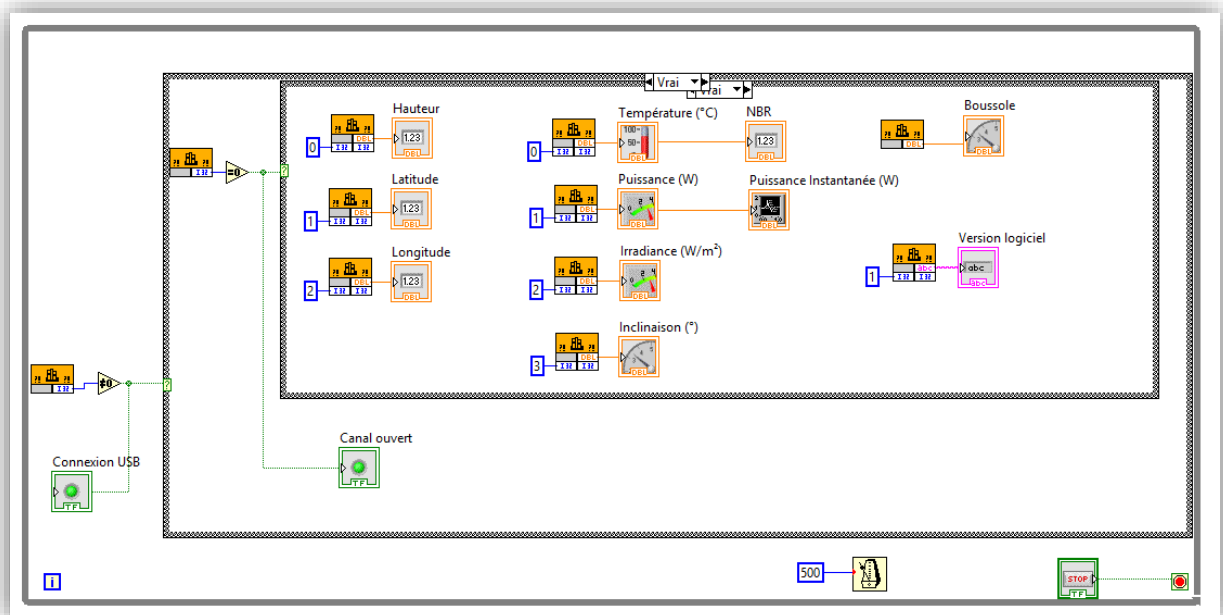


2. Étude de l'interface de supervision

La valise AGS est munie de nombreux capteurs électroniques. Ces capteurs sont aujourd'hui supervisés par le logiciel AGS. Les données issues de ces capteurs permettent ainsi de fiabiliser le calcul de productible d'une installation solaire photovoltaïque, et montrer en temps réel la production instantanée d'un panneau solaire photovoltaïque. Nous utiliserons le logiciel Labview pour permettre la compréhension des fonctionnalités.



Nous utiliserons également le fichier dll dans lequel sont intégrées des fonctions. Ces fonctions retournent une valeur précise en fonction de l'entrée. Par exemple, si l'on veut la valeur de la température, d'après la bibliothèque en annexe, il faut mettre la valeur « 0 » en entrée pour obtenir la valeur de température en sortie. Nous obtenons ce circuit dans la partie Bloc Diagramme du logiciel LabView.



Nous branchons la valise AGS à l'ordinateur pour pouvoir visualiser les valeurs des capteurs et nous obtenons le résultat dans la face avant du logiciel LabView.



Dans l'interface, nous pouvons constater que la valise est bien connectée et le canal est bien ouvert. Nous pouvons également visualiser les valeurs des capteurs de la valise : la température, la version du logiciel, la boussole, l'inclinaison, etc... La puissance et l'irradiance ne peut être défini car dans la pièce où nous avons mesuré, il y avait de la lumière artificielle. Il faut une source lumineuse du soleil pour avoir une puissance bien précise.

3. Étude de la carte μ Contrôleur et du logiciel embarqué

3.1 Étude de la carte électronique

1°) À partir du schéma de la carte μ Contrôleur fourni en annexe et de la documentation technique du *PIC18F46J50*, préciser quelle fonctionnalité du μ Contrôleur est utilisée des broches suivantes.

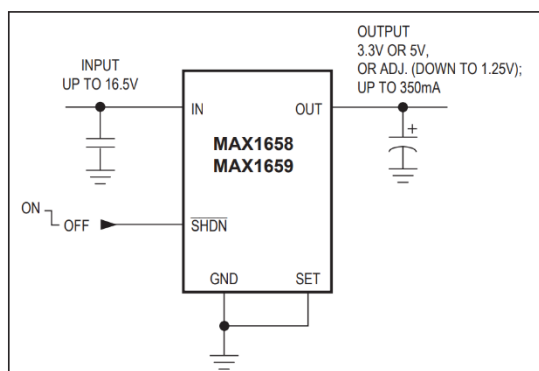
BROCHE	FONCTIONS
1	RC7 - Entrée / Sortie numériques PMA4 - Adresse du port maître parallèle RX1 - Réception asynchrone USART1 DT1 - Sortie/entrée de données synchrones SDO1 - Sortie de données
2	RD4 - Entrée / Sortie numérique
8	RB0 - Entrée / Sortie numérique AN12 - Entrée analogique 12 INT0 - Interruption externe 0
9	RB1 - Entrée / Sortie numérique AN10 - Entrée analogique 10
10	RB2 - Entrée / Sortie numérique AN8 - Entrée analogique 8

11	RB3 - Entrée / Sortie numérique AN9 - Entrée analogique 9 VPO - Émetteur-récepteur USB externe Sortie de données D+
19	RA0 - Entrée / Sortie numérique AN0 - Entrée analogique 0
20	RA1 - Entrée / Sortie numérique AN1 - Entrée analogique 1
21	RA2 - Entrée / Sortie numérique AN2 - Entrée analogique 2 VREF- - Entrée de la tension de référence
22	RA3 - Entrée / Sortie numérique AN3 - Entrée analogique 3 VREF+ - Entrée de la tension de référence
24	RA5 - Entrée / Sortie numérique AN4 - Entrée analogique 4
25	RE0 - Entrée / Sortie numérique AN5 - Entrée analogique 5
26	RE1 - Entrée / Sortie numérique AN6 - Entrée analogique 6
32	RC0 - E/S numérique T1OSO - Sortie de l'oscillateur du Timer1 T1CKI - Entrée d'horloge externe des Timer1/Timer3
36	RC2 - E/S numérique AN11 - Entrée analogique 11
38	RD0 - E/S numérique
39	RD1 - E/S numérique
42	RC4 - E/S numérique D- - Entrée/sortie de ligne négative du bus USB
43	RC5 - E/S numérique D+ - Entrée/sortie de ligne positive du bus USB
44	RC6 - E/S numérique

2°) Quelle est la tension d'alimentation du μ Contrôleur ?

La tension d'alimentation du μ Contrôleur est de 3,3V.

3°) Télécharger la documentation technique du composant *MAX1658*. Quelle est sa fonction ?



Le composant MAX1658 est un régulateur linéaire qui maximise la durée de vie des batteries en combinant des courants d'alimentation ultra-faibles et des tensions de sortie faibles.

Le composant MAX1658 est un régulateur de tension linéaire à faible chute.

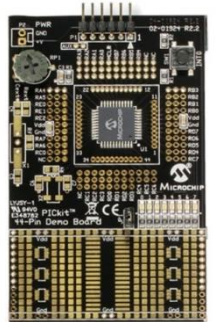
4°) Télécharger la documentation technique du composant *MAX3232CSE*. Quelle est sa fonction ?

Le composant MAX3232CSE est un émetteur récepteur double.

3.2 Environnement de développement logiciel MPLAB X

Le programme de la prise en main du logiciel MPLAB X consiste à faire clignoter la LED sur la carte Pickit 3.

Nous programmons le code en langage C.



Appel des bibliothèques

```
#include <stdio.h>
#include <stdlib.h>
#include "xc.h"
```

Initialisation de la sortie LED
Tous les LED sont éteint

```
int main() {
    TRISD = 0x00;
```

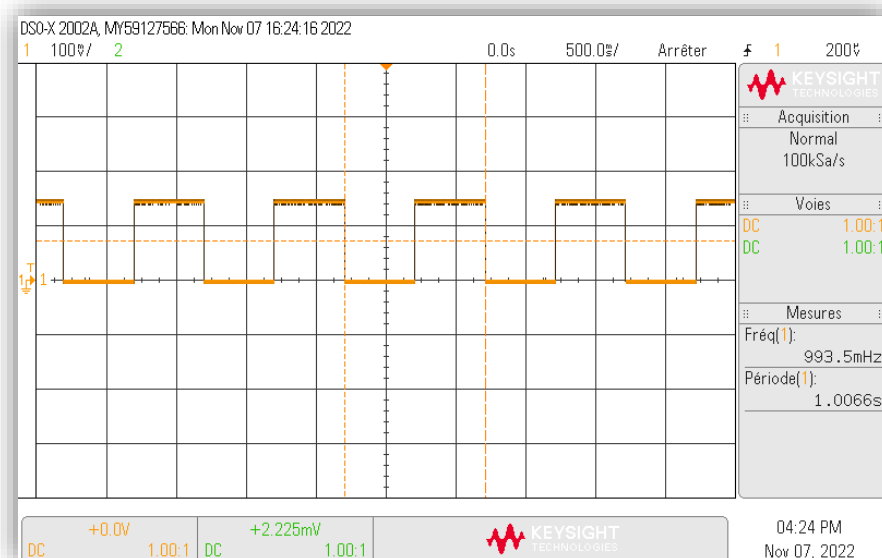
Boucle de temporisation

```
    int i;
    while(1)
    {
        for (i=0; i< 15000; i++){
            PORTDbits.RD7 = !PORTDbits.RD7;
        }
    }
```

Inverse l'état de la LED

```
    return (EXIT_SUCCESS);
}
```

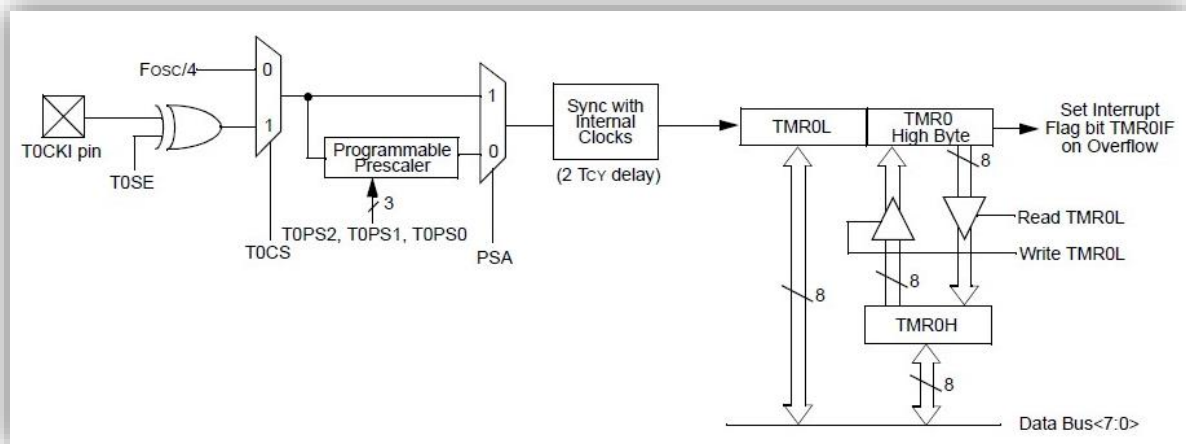
Nous utilisons la boucle For pour la temporisation, puis nous inversons le bit de la LED. Nous visualisons la tension de la LED sur l'oscilloscope.



Nous constatons que la fréquence est d'environ 1Hz et la période d'une seconde. La LED clignote toutes les 500ms.

Pour garantir une meilleure précision de notre période, nous allons utiliser une temporisation intégrée au µContrôleur nommé Timer0.

Le but du Timer en 16 bits est de compter de la valeur qu'on entre jusqu'à FFFFH. Lorsque le timer atteint FFFFH, le timer repasse à 0000H, à ce moment-là le bit de débordement passe à 1.



Dans le Timer0, nous avons 2 registres importants : **TOCON** et **INITCON**.

REGISTRE DE CONTROLE DU TIMER0 : TOCON

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TMR0ON	T08BIT	T0CS	T0SE	PSA	T0SP2	T0SP1	T0PS0
0	0	0	0	0	1	1	1

- Arrêt du Timer0
 - Configuration en 16 bits
 - Horloge interne (timer)
 - Incrémentation sur front montant
 - Affectation du Prescaler de 256
- ➔ TOCON = 0x07

Pour définir la valeur TMR0 en 16 bits, nous aurons 2 variables en 8 bits : TMR0H et TMR0L.

Nous avons une fréquence d'instructions de 1MHz. Donc, le temps d'un cycle est de 1µs.

$$T_{cycle} = \frac{1}{1\text{ MHz}} = 1\mu\text{s}$$

$$TMR0 = 2^{16} - \left(\frac{\text{délais}}{T_{cycle} \times \text{Prescaler}} \right)$$

$$TMR0 = 2^{16} - \left(\frac{0,1\text{ s}}{1\mu\text{s} \times 256} \right) = 65145 = 0xFE79$$

➔ TMR0H = 0xFE

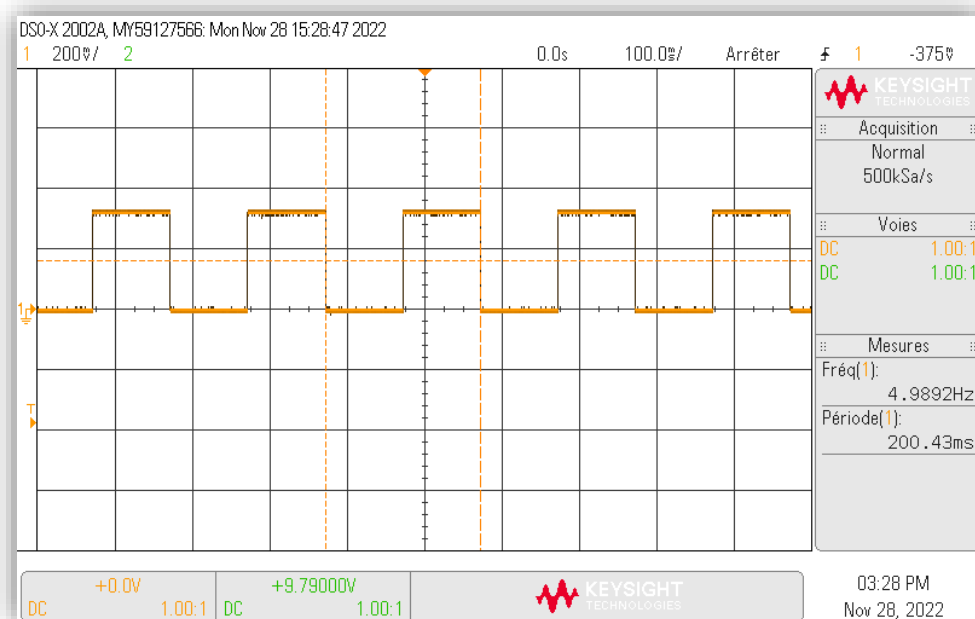
➔ TMR0L = 0x79

Nous réalisons le code en langage C en utilisant le Timer0.

```
#include <stdlib.h>
#include <stdio.h>
#include "xc.h"

void main (void)
{
    TRISD = 0; // Initialisation des LED en sortie
    PORTD = 0;
    TOCON = 0x07; // Initialisation du registre de contrôle
    TMR0H = 0xFF; // Initialisation du registre TMR0
    TMR0L = 0x9E;
    TOCONbits.TMR0ON = 1; // Démarrage du Timer0
    while(1)
    {
        if (INTCONbits.TMR0IF) // Si débordement = 1 alors
        {
            PORTDbits.RD7 = !PORTDbits.RD7; // Inverser l'état de la LED
            TMR0H = 0xFF; // Réinitialisation du registre TMR0
            TMR0L = 0x9E;
            INTCONbits.TMR0IF = 0; // Réinitialisation du débordement
        }
    }
}
```

Dans ce programme, nous utilisons bit de débordement inclut dans le registre INTCON. Lorsque le bit de débordement (TMR0IF) passe à 1, alors on arrête le timer et on inverse l'état de la LED. Nous visualisons la tension de la LED.

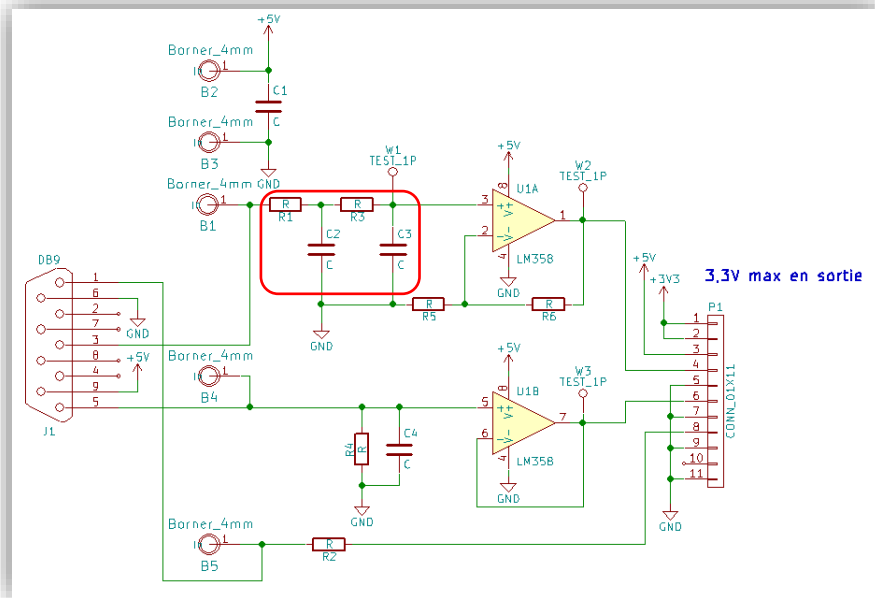


Nous constatons que la fréquence est d'environ 5 Hz et la période d'une seconde. La LED clignote toutes les 200 ms. Nous pouvons donc confirmer que le Timer0 est plus précis que les boucles de temporisation.

4. Étude de la carte d'acquisitions analogiques

4.1 Étude du filtre d'entrée d'acquisition analogique tension PV

Quelle est la fonction de ce filtre ?

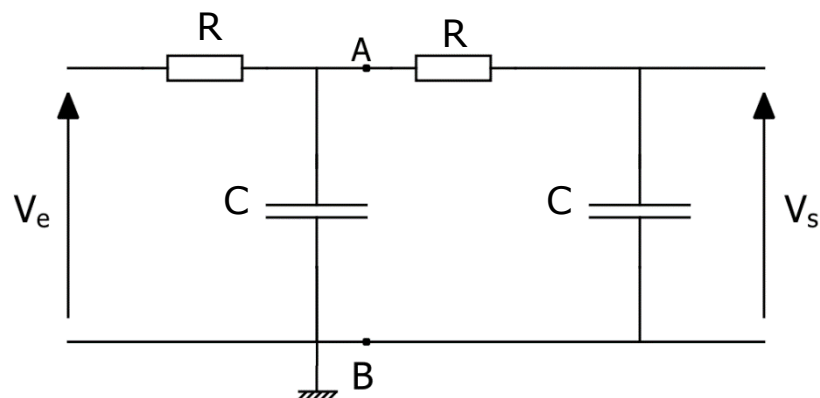


C'est un filtre passe bas de second ordre. Sa fonction est de laisser passer les basses fréquences tout en coupant les hautes fréquences.

Établir la fonction de transfert $T(j\omega)$ de ce filtre. Simplifier l'expression de la fonction de transfert en posant $R_1 = R_3 = R$ et $C_2 = C_3 = C$. Écrire alors la fonction de transfert sous la forme :

$$T(p) = \frac{\omega_0^2}{p^2 + 2m\omega_0 p + \omega_0^2}$$

Fonction de transfert filtre passe bas de second ordre



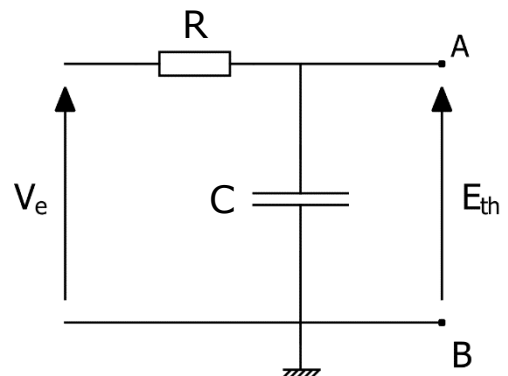
Utilisation du théorème de Thévenin

On commence par déterminer la tension équivalente du circuit :

$$E_{th} = V_e \times \frac{Z_c}{Z_R + Z_c}$$

$$E_{th} = V_e \times \frac{\frac{1}{jC\omega}}{R + \frac{1}{jC\omega}}$$

$$E_{th} = V_e \times \frac{1}{1 + jRC\omega}$$



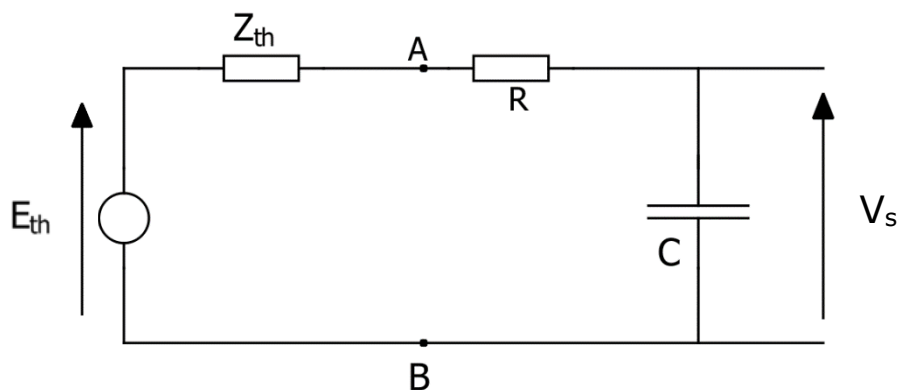
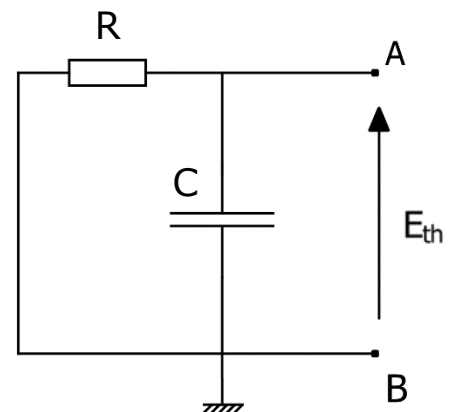
On calcule l'impédance équivalente du circuit :

$$Z_{th} = Z_R \parallel Z_C$$

$$Z_{th} = \frac{\frac{R}{jC\omega}}{R + \frac{1}{jC\omega}}$$

$$Z_{th} = \frac{\frac{R}{jC\omega}}{\frac{RjC\omega + 1}{jC\omega}}$$

$$Z_{th} = \frac{R}{1 + RjC\omega}$$



A partir de ce schéma équivalent, on va déterminer la fonction de transfert.

D'après le pont diviseur de tension :

$$V_s = \frac{\frac{1}{jC\omega}}{Z_{th} + R + \frac{1}{jC\omega}} \times E_{th} \quad \Leftrightarrow \quad V_s = \frac{\frac{1}{jC\omega}}{\frac{1 + jC\omega(Z_{th} + R)}{jC\omega}} \times E_{th}$$

$$V_s = \frac{1}{1 + jC\omega(Z_{th} + R)} \times E_{th} \quad \Leftrightarrow \quad V_s = \frac{1}{1 + jC\omega(Z_{th} + R)} \times V_e \times \frac{1}{1 + jRC\omega}$$

$$\frac{V_s}{V_e} = \frac{1}{1 + jC\omega \left(\frac{R}{1 + jRC\omega} + R \right)} \times \frac{1}{1 + jRC\omega} \quad \Leftrightarrow \quad \frac{V_s}{V_e} = \frac{1}{1 + jC\omega \left(\frac{R + R^2jC\omega + R}{1 + jRC\omega} \right)} \times \frac{1}{1 + jRC\omega}$$

$$\frac{V_s}{V_e} = \frac{1}{\frac{(1 + jRC\omega) + jC\omega(2R + R^2jC\omega)}{1 + jRC\omega}} \times \frac{1}{1 + jRC\omega}$$

$$\frac{V_s}{V_e} = \frac{1 + jRC\omega}{(1 + jRC\omega) + jC\omega(2R + R^2jC\omega)} \times \frac{1}{1 + jRC\omega}$$

$$\frac{V_s}{V_e} = \frac{1}{1 + jRC\omega + jC\omega(2R + R^2jC\omega)} \quad \Leftrightarrow \quad \frac{V_s}{V_e} = \frac{1}{1 + jRC\omega + 2jRC\omega + (jRC\omega)^2}$$

$$\Rightarrow \quad T = \frac{V_s}{V_e} = \frac{1}{1 + 3jRC\omega + (jRC\omega)^2}$$

Établir les expressions de ω_0 pulsation propre et m facteur d'amortissement de ce filtre ?

On divise par $(RC)^2$

$$\omega_0 = \frac{1}{RC}; p = j\omega$$

$$T = \frac{\frac{1}{(RC)^2}}{p^2 + \frac{3RC}{(RC)^2}p + \frac{1}{(RC)^2}}$$

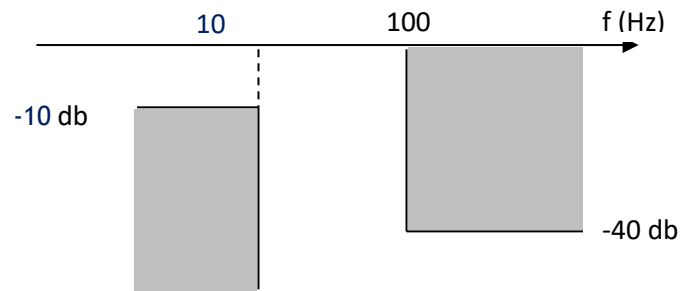
Pour trouver m :

$$2 m \omega_0 = \frac{3RC}{(RC)^2}$$

$$2 m \omega_0 = 3 \omega_0$$

$$m = \frac{3}{2}$$

La fonction de transfert de ce filtre doit répondre au gabarit suivant :



On donne $R = 47\text{k}\Omega$. Calculer alors la valeur de C permettant de satisfaire le gabarit ci-dessus.

Nous avons $C = 470\text{ nF} = 470 \times 10^{-9}\text{ F}$

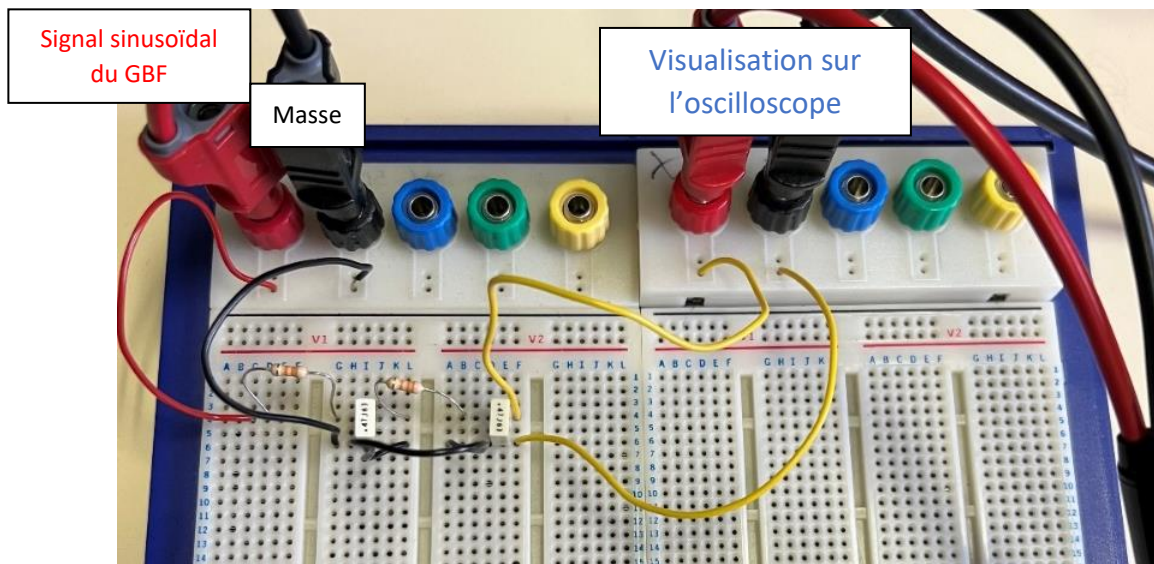
$$\omega_0 = \frac{1}{RC} \Leftrightarrow 2\pi f_0 = \frac{1}{RC} \Leftrightarrow 20\pi = \frac{1}{470 \times 10^{-9} \cdot R}$$

$$R = \frac{1}{20\pi \times 470 \times 10^{-9}} = 33\,862,75\,\Omega$$

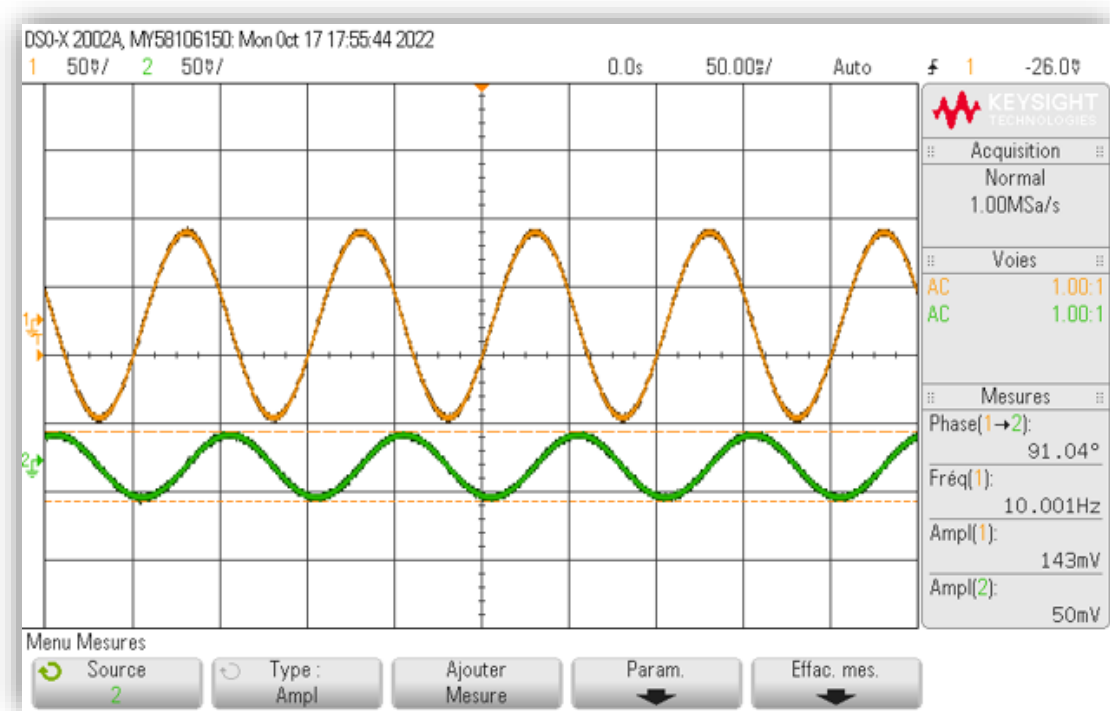
Nous obtenons donc $C = 470\text{ nF}$ et $R = 33\text{ k}\Omega$ car $33,8\text{ k}\Omega$ n'est pas une valeur normalisée.

Câbler le filtre sur une plaque d'essais et tracer la courbe de gain en fonction de la fréquence.

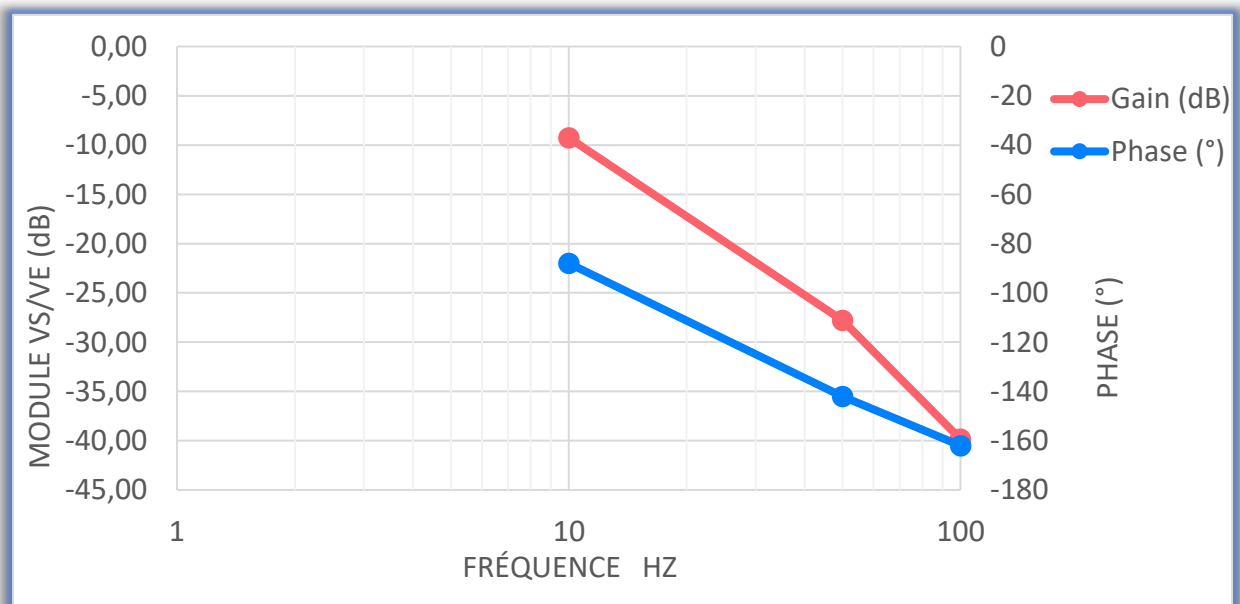
Nous réalisons le câblage sur une plaque d'essai avec une résistance $33,8\text{ k}\Omega$ et un condensateur de 470 nF . On reçoit un signal sinusoïdal de 10 kHz du GBF.



Nous visualisons sur l'oscilloscope le signal de sortie aux bornes du condensateur.

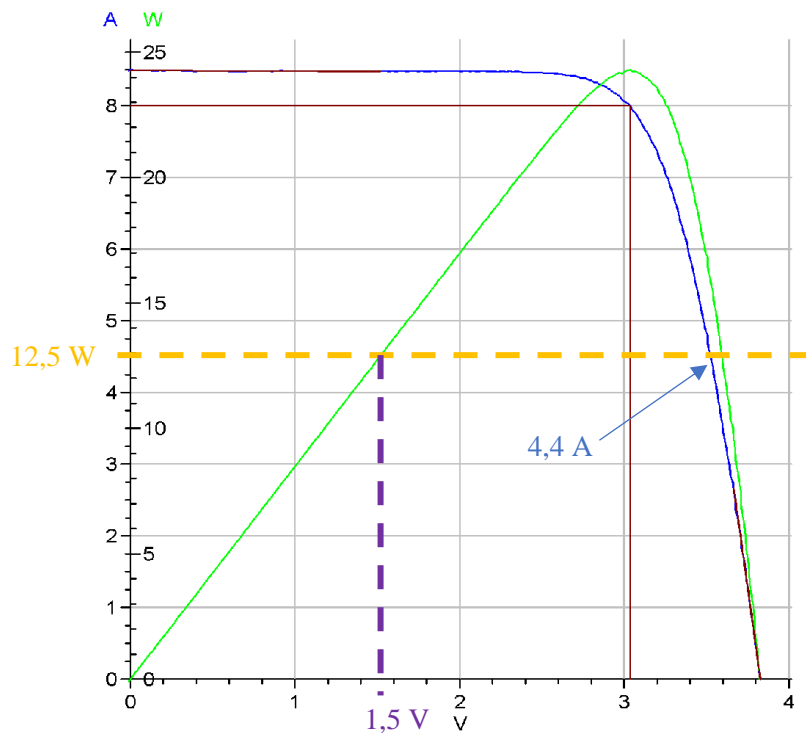


Nous récupérons la phase et la tension de sortie à 10Hz, 50Hz et 100Hz sur l'oscilloscope. Nous traçons la courbe de gain en fonction de la fréquence.



4.2 Étude de la résistance de charge du panneau solaire

Un relevé des performances du panneau solaire est fourni en annexe. Sur ce document figure différentes caractéristiques (courbe $I = f(v)$ et courbe de puissance). Déterminer à l'aide de ces caractéristiques, la puissance maximum que peut produire le panneau.



Puissance maximale = 25 W

À 12,5 W :

$V_{mp} = 1,5 \text{ V}$

$I_{mp} = 4,4 \text{ A}$

V_{mp} est la tension et I_{mp} le courant lorsque la puissance maximum est à 50% c'est-à-dire 12,5 W.

Une résistance de charge R montée sur la valise AGS permet de définir un point de fonctionnement du panneau. On dispose pour R d'une résistance de puissance du type RH25 dont les caractéristiques constructeur sont fournies en document ressource.

Quelle est la puissance maximale que peut dissiper ce type de résistance ?

La puissance maximale, que peut dissiper ce type de résistance est de 35 W.

Quelle est la puissance que peut dissiper cette résistance lorsqu'elle n'est pas montée sur un dissipateur ?

La puissance maximale que peut dissiper cette résistance lorsqu'elle n'est pas montée sur un dissipateur est de 12,5 W.

On souhaite limiter à 80% la puissance dissipée par R par rapport à la puissance établie à question 6.5.2b. Calculer alors cette nouvelle valeur de la puissance ?

80 % de 12,5 = 10 W La nouvelle puissance vaut 10 W.

4.3 Etude de la sonde de température

La valise AGS est équipée d'une sonde de température afin de mettre en évidence l'élévation de température du panneau solaire lorsqu'il est au soleil. Une sonde de température CTN Epcos est utilisée.

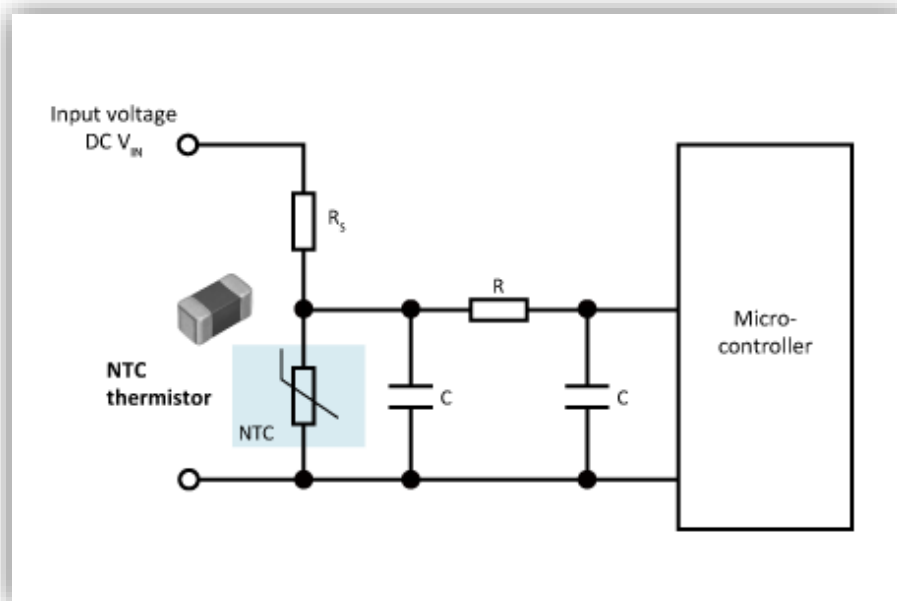
Analyser le fonctionnement et les caractéristiques de la CTN. Quel est le modèle équivalent du composant ?

La CTN EPCOS est une thermistance CTN pour la mesure de température. Elle est principalement utilisée dans les mesures de température de surface sur les boîtiers et les dissipateurs de chaleur.

CARACTÉRISTIQUES

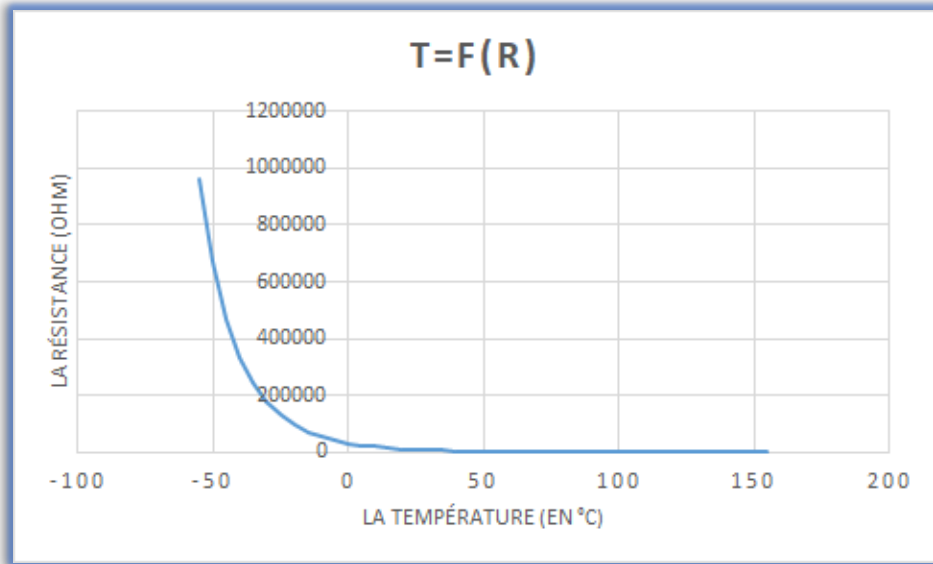
- Haute précision
- Montage aisé
- Bon couplage thermique par bague métallique
- Thermistance encapsulée dans un boîtier métallique
- Cordons isolés PTFE constitués de fil nickel argenté 30AWG et de fil Cu argenté 28 AWG
- Approuvé UL (E69802)
- Valeur de résistance à 25°C comprise entre 5kohm et 30Kohm
- Gamme de valeur Beta (valeur B25/100) de 3964K à 3988K
- Température d'utilisation de -55°C à 125°C
- Dissipation de puissance maximum de 150mW à 25°C
- Applications
- Industrie, Electronique Grand Public, Périphériques Portables, Détection et Instrumentation

Modèle équivalent CTN EPCOS



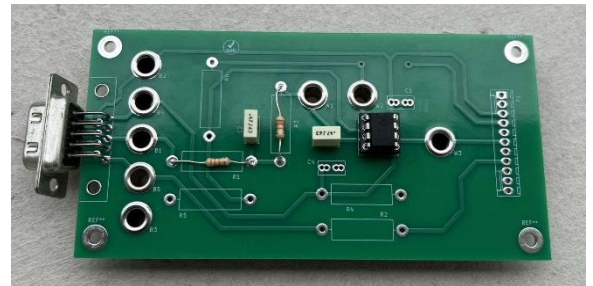
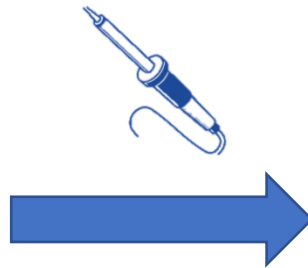
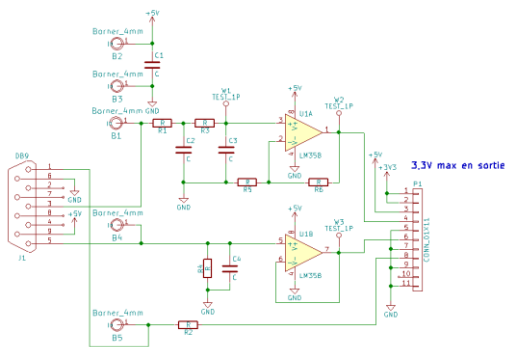
A partir du tableau de valeurs fourni par le constructeur, tracer la courbe $T=f(R)$ pour une valeur nominale de la CTN de 10 k Ω .

Nous traçons la courbe de la température en fonction de la résistance. Nous pouvons constater que plus la température augmente, plus la résistance diminue.



4.4 Fabrication de la carte électronique analogique

Pour finir, nous allons souder tous les composants sur un circuit imprimé.



Conclusion

Ce projet s'est révélé très productif. En effet, le respect des délais et le travail seront primordiaux pour le métier qu'on voudrait faire plus tard. De plus, ce projet nous a permis d'appliquer nos connaissances en électronique, en informatique embarquée. Mais aussi à câbler des composants électroniques sur une carte électronique. Nous avons également appris à faire de la soudure sur un circuit imprimé. Nous avons vraiment aimé réaliser ce projet et nous avons vraiment passé du temps dessus. Ce type de projet nous permet de nous rapprocher à notre futur métier.

Annexe

Fonction		Commentaire
present		teste si la carte est détectée sur le bus USB
	Paramètre	
	-	
	Retour	
	int	=0 si carte non détectée ≠0 sinon
open		ouvre le canal de communication USB
	Paramètre	
	-	
	Retour	
	int	=0 si ouverture réussie ≠0 sinon
close		ferme le canal de communication USB
	Paramètre	
	-	
	Retour	
	int	=0 si ouverture réussie ≠0 sinon
state		donne le dernier état de communication
	Paramètre	
	-	
	Retour	
	int	=0 si canal ouvert =1 si canal fermé * si erreur durant la communication
analog		fournit une valeur analogique
	Paramètre	
	int	=0 analogique demandé est la température =1 analogique demandé est la puissance =2 analogique demandé est l'irradiance =3 analogique demandé est l'inclinaison
	Retour	
	double	analogique demandé
compass		fournit une valeur du composant boussole
	Paramètre	
	-	
	Retour	
	double	direction par rapport au nord
gps		fournit une valeur du composant GPS
	Paramètre	
	int	=0 valeur GPS est l'altitude =1 valeur GPS est latitude =2 valeur GPS est longitude
	Retour	
	double	valeur GPS demandée
version		fournit les versions logicielles
	Paramètre	
	int	=0 version demandée est bootloader =1 version demandée est firmware
	Retour	
	const char *	version sous la forme :major.minor

Bibliothèque USB