

MPLABX TP01 : INITIATION À LA SIMULATION

Objectifs : Etre capable de simuler un programme à l'aide de MPLABX.

1. CAHIER DES CHARGES

On souhaite simuler un programme avec MPLABX avant de l'implanter sur la cible (PIC16F1575)

Le programme consiste à faire clignoter une LED (RC1) toutes les secondes et de tester l'appui sur un bouton poussoir en entrée RC0, cet appui entraîne l'allumage d'une LED sur le port RC2.

2. CONFIGURATION

En vous aidant de MCC (microchip code configurator) configurer les entrées/sorties du PIC.

Module	Function	Direction	Pin No.	13	12	11	4	3	2	10	9	8	7	6	5
ADC1	ADCACT	input													
	ANx	input													
	VREF+	input													
CWG	CWGA	output													
	CWGB	output													
DAC	DACOUT	output													
	VREF+	input													
EUSART	RX	input													
	TX	output													
OSC	CLKIN	input													
	CLKOUT	output													
PWM1	PWM1OUT	output													
PWM2	PWM2OUT	output													
Pin Module	GPIO	input													
	GPIO	output													
RESET	MCLR	input													
TMR0	T0CKI	input													
	T1CKI	input													
TMR1	T1G	input													

Pour chaque fonction activée dans MCC deux fichiers sont générés : .c (visible dans l'onglet "projects"+"source Files") et .h (visible dans l'onglet "projects"+"header files").

3. PROGRAMME EN C

Écrire le programme en C dans le fichier main.c

Le code de la boucle principale est :

```
ADC1_StartConversion(channel_AN1);

while (1)
{
    // Add your application code
    LEDRC1_PORT = 1 ;
    __delay_ms(1000);
    LEDRC1_PORT= 0;
    __delay_ms(1000);
    if (BPRC0_PORT==0)
    {
        LEDRC2_PORT = 1 ;
    }
    else LEDRC2_PORT=0;
    valLue = ADC1_GetConversionResult();
}
```

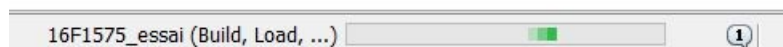
Une fois le code écrit, vous pouvez construire le projet (Run+Built¹) ou le débbugger (Debug+Debug project)

4. SIMULER LE PROJET

4.1. Lancement

Passer ne mode : Debug project

Une barre de progression en bas à droite indique que le projet est en mode RUN :



Il est maintenant possible d'agir sur les STIMULUS (entrées simulées) et de voir l'effet sur les sorties.

4.2. Stimulus

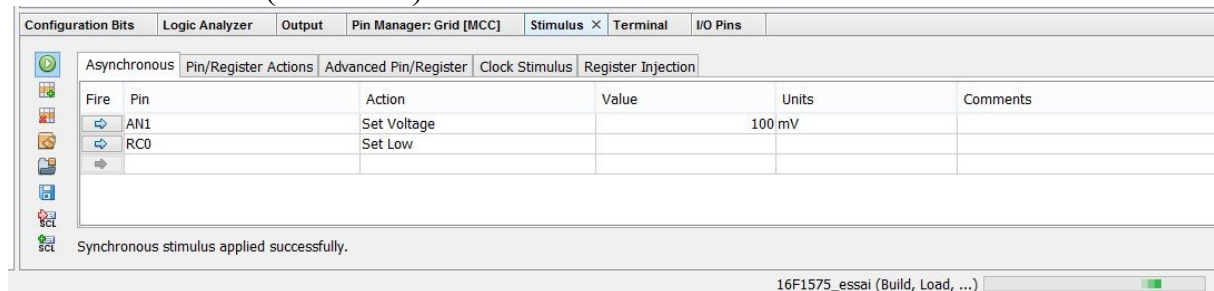
Un stimulus permet de simuler une action en entrée : appui sur Bouton poussoir, valeur d'une entrée analogique (CAN0...)

¹ Lorsque le projet est construit il crée un fichier .hex à téléverser dans le PIC par le PICKIT3.

Ici nous simulons :

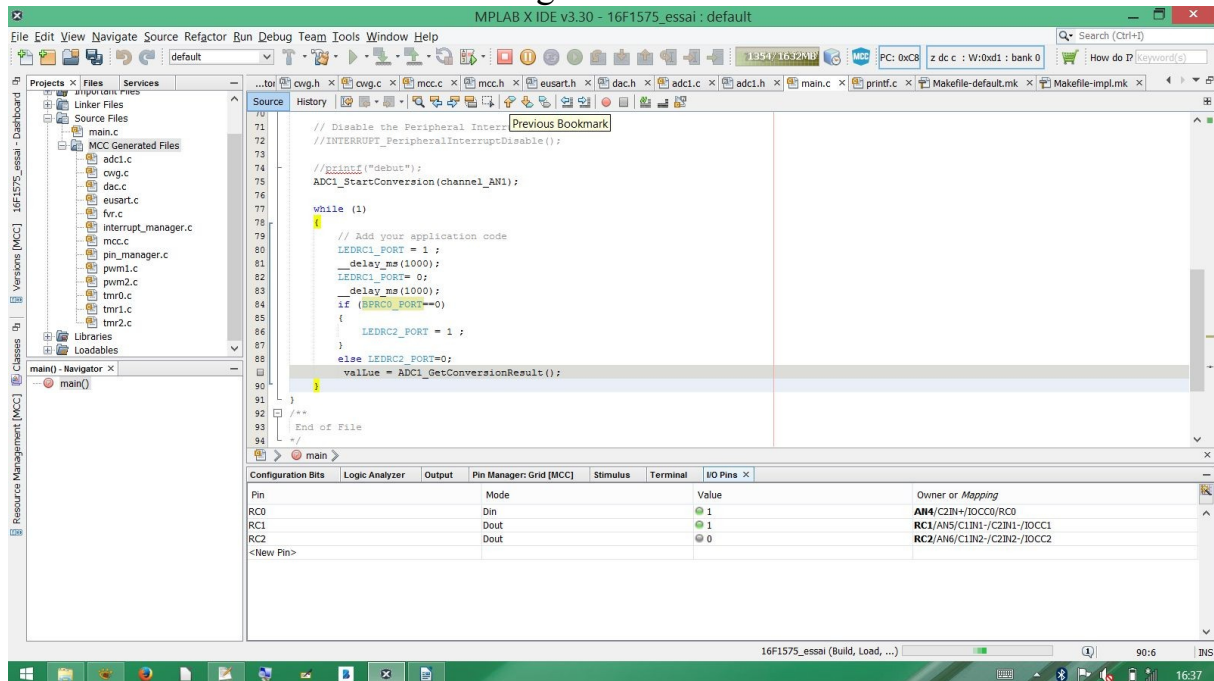
AN1 : 100mV sur l'entrée analogique

RC0 : valeur à 0 (set LOW)



4.3. Résultats de simulation

Les résultats sont visibles dans l'onglet : "I/O Pins"



Configuration Bits	Logic Analyzer	Output	Pin Manager: Grid [MCC]	Stimulus	Terminal	I/O Pins
Pin		Mode			Value	Owner or Mapping
RC0		Din			1	AN4/C2IN+/IOCC0/RC0
RC1		Dout			0	RC1/AN5/C1IN1-/C2IN1-/IOCC1
RC2		Dout			0	RC2/AN6/C1IN2-/C2IN2-/IOCC2
<New Pin>						

La sortie RC1 évolue au rythme de 1s à 0 et 1s à 1.

En modifiant la valeur de RC0 la sortie RC2 change : tester le !.

Pin	Mode	Value	Owner or Mapping
RC0	Din	 0	AN4 /C2IN+ /IOCC0/RC0
RC1	Dout	 1	RC1 /AN5/C1IN1- /C2IN1- /IOCC1
RC2	Dout	 1	RC2 /AN6/C1IN2- /C2IN2- /IOCC2
<New Pin>			

5. CONCLUSION

L'utilisation de la simulation est pratique afin de tester un programme en l'absence de partie opérative. Cela nécessite de créer des stimulus bien choisis ce qui facilite la compréhension du process complet.