

Bien commencer avec un LaunchPad MSP430G et un Breadboard

Pierre-Yves Rochat

Le LaunchPad et Energia

Texas Instrument propose un environnement de développement très complet pour ses MSP430, appelé *Code Composer Studio* (CCS). Mais le débutant préférera utiliser l'environnement **Energia**, une copie du célèbre *Arduino* (qui utilise un microcontrôleur AVR). Toutes les informations pour l'installer sous Windows, Linux ou MacOs se trouvent sur <http://energia.nu>.

Sous Windows, n'oubliez pas d'installer le pilote USB (driver) : http://energia.nu/Guide_Windows

L'utilisation d'Energia est extrêmement simple. Branchez votre LaunchPad. Après avoir lancé le programme, choisissez dans le menu : *File – Examples – 1. Basics – Blink*.



L'icône **Verify** lance la compilation, qui ne prend du temps que la première fois. S'il y a des erreurs, elles s'affichent. L'icône **Upload** permet ensuite l'envoi du programme binaire sur le microcontrôleur, à travers les signaux Test et Reset. Après quelques secondes, vous allez voir clignoter la Led rouge.

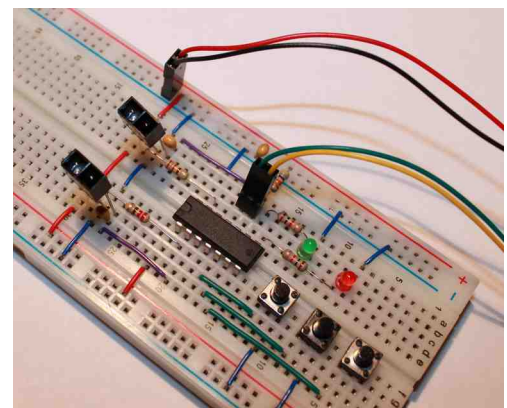
Vous pourrez alors passer des heures à modifier votre programme pour le faire clignoter plus vite, plus lentement, seulement lorsque le bouton-poussoir est pressé, tellement vite que la Led donnera l'impression d'être faiblement allumée, etc.

Passer sur BreadBoard

Mais vous finirez vite par vous lasser de faire des programmes qui se contentent de deux LED et d'un bouton-poussoir ! Ajouter des composants sur les connecteurs du LaunchPad n'est pas vraiment facile.

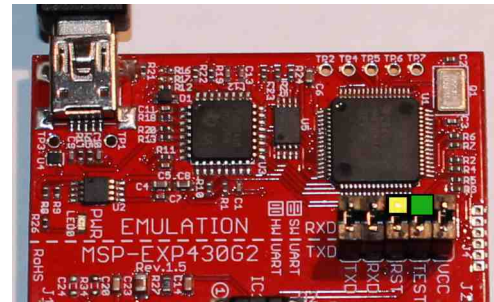
Le LaunchPad MSP430 est un formidable outil de programmation et de debug. Son socle est bien pratique pour commencer les tests sans autre matériel, mais il est souvent plus pratique de placer le microcontrôleur sur un BreadBoard pour réaliser un montage un peu plus complexe.

Or souvenez-vous que la programmation d'un MSP430 ne nécessite que deux signaux : Reset et Test. Il faut encore deux fils pour l'alimentation. Il est plus facile de tirer 4 fils entre le LaunchPad et le BreadBoard que de devoir mettre autant de fils que l'application en utilise !



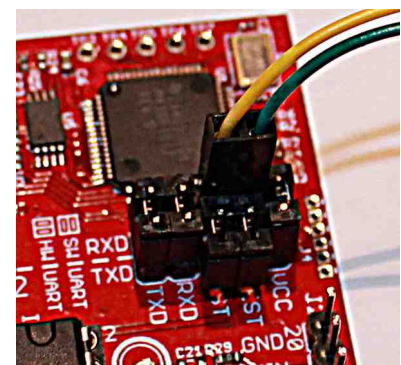
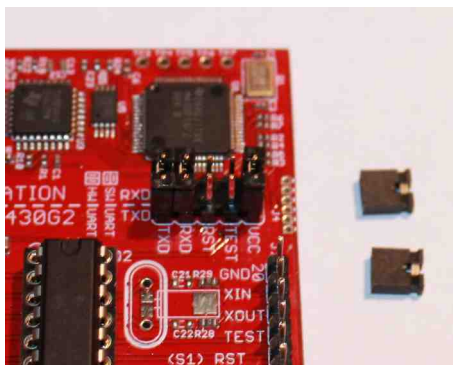
Le câblage est le suivant :

Signal	Patte (DIL 20-14)	Couleur proposées
Gnd	20 (ou 14)	Noir
Vcc	1	Rouge
Test	17 (ou 11)	Vert
Reset	16 (ou 10)	Jaune



Les versions Dual in Line (boîtiers avec deux rangées de pattes) des MSP430 ont 14 ou 20 pattes. Le MSP430G2553 livré sur le socle du LaunchPad a 20 pattes. Le MSP430G2231 (qui était livré jusqu'à la version 1.4 du LaunchPad) a 14 pattes. C'est la raison de la double numérotation de la colonne *Patte* du tableau.

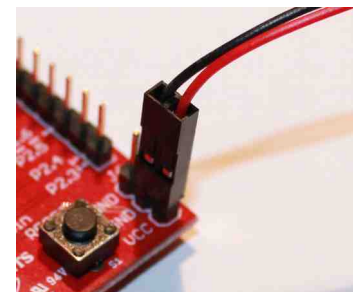
Voici la marche à suivre pour brancher le fil jaune-vert avec Test et Reset :



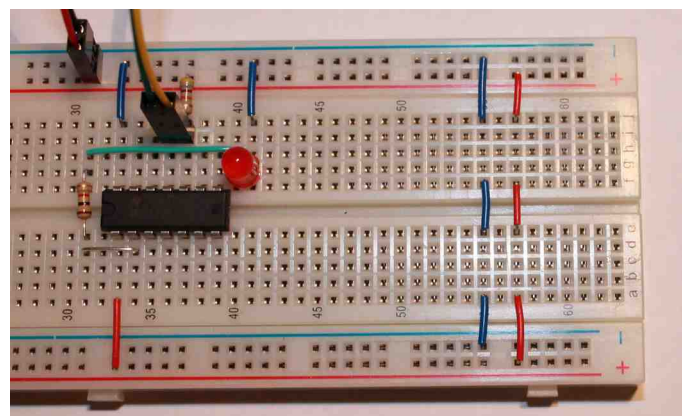
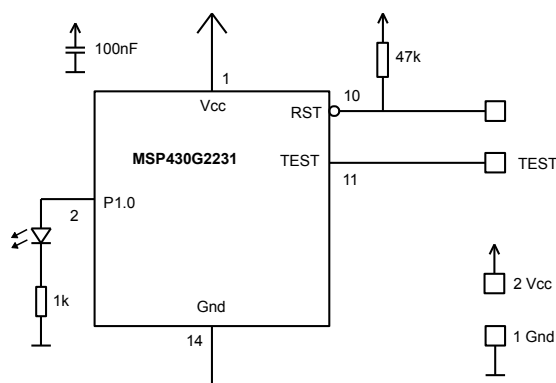
- enlever les jumpers
- les remettre décalés vers le bas pour ne pas les perdre
- brancher le fil

LaunchPad dispose d'une prise de sortie de son alimentation 3.5 Volt. Attention, le courant est limité ! Ne tirez pas plus de 200 mA sur cette prise. Utilisez une alimentation externe pour davantage de courant.

Observez bien le brochage du connecteur à trois pattes, qui donne successivement Gnd, Gnd et Vcc. Respectez les couleurs (Noir pour le Gnd et rouge pour le +Vcc).



Monter les composants sur le BreadBoard



La plaque d'expérimentation, généralement appelée BreadBoard, est couverte de trous. Ils sont connectés ensemble par groupe de 5 trous. Les deux bandes bleues et rouge de chaque côté sont par contre reliées ensemble d'un bout à l'autre de la plaque.

Voici comment vous pouvez réaliser votre premier montage, avec juste ce qu'il faut pour faire clignoter une Led.

Quelques remarques :

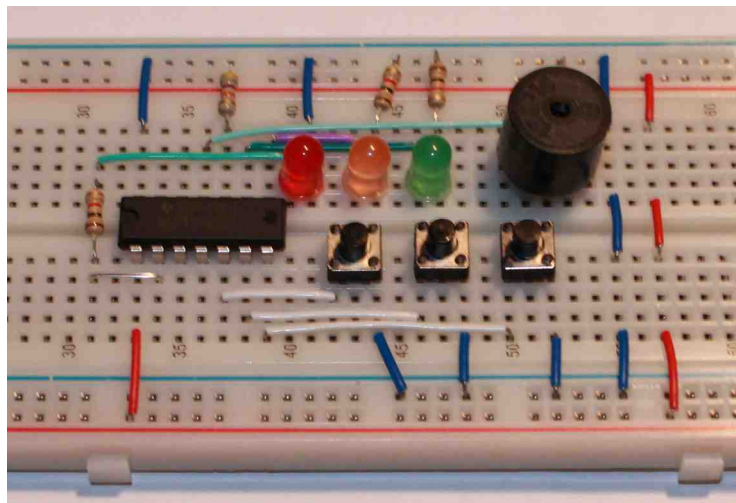
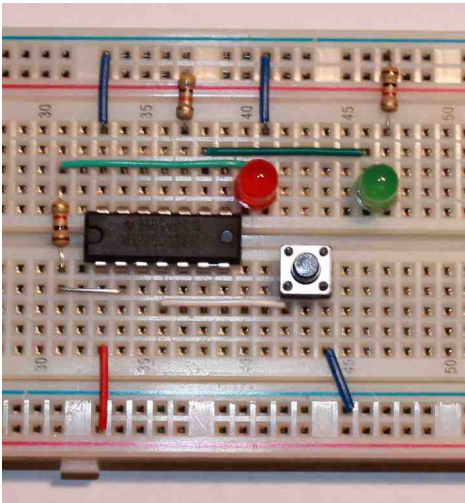
- Le câblage est réalisé avec des fils rigides, d'environ 0.5mm de diamètre. Des chutes de câble téléphonique ou de câble réseau Ethernet conviennent.
- C'est un peu difficile de réaliser des fils ajustés et coudés, mais c'est tellement plus joli et surtout plus pratique. On pourra ensuite facilement ajouter des fils « en cloche » pour compléter le montage, sans être gêné par les fils de base. Une bonne pince à dénuder aide bien. On peut aussi faire du travail propre en utilisant astucieusement un couteau, si possible un couteau suisse ;-)
- Pour les fils courts, ils n'ont pas besoin d'être isolés, comme le fil entre la patte 2 et la résistance, réalisé avec une patte de Led coupée ! Mais on verra plus loin que l'isolation est bien utile dès que le fil est un peu plus long.
- Ne vous étonnez pas que la LED rouge soit si loin de la patte 2. C'est en vue de la suite du montage.
- Les fils souples en provenance du LaunchPad se connectent en face des pattes 10 et 11 pour Reset et Test, et sur les rangées d'alimentation pour le Gnd et le +Vcc.
- Il faut relier les rangées des alimentations de part et d'autre. Ici, on a utilisé des fils courts, pour l'esthétique.



Compléter le montage

Votre montage pourra évoluer, d'abord pour reproduire les mêmes composants que le LaunchPad, en respectant la compatibilité de pattes.

Ensuite, vous pourrez compléter le montage selon votre imagination...



Définitions matérielles

Le choix des pattes utilisées doit être reporté correctement dans les programmes.

En utilisant l'environnement Energia, voici les définitions :

```
#define Led1 P1_0 // Led rouge, active à 1 (compatible Launchpad) pin 2
#define Led2 P1_6 // Led verte, active à 1 (compatible Launchpad) pin 9
#define Pous1 P1_3 // Poussoir, actif à 0 (compatible Launchpad) pin 5
#define Pous2 P1_4 // Poussoir, actif à 0, pin 6
```

On pourra alors définir les accès de bas niveau :

```
#define Led1On digitalWrite (Led1, HIGH);
#define Led1Off digitalWrite (Led1, LOW);
#define Led2On digitalWrite (Led2, HIGH);
#define Led2Off digitalWrite (Led2, LOW);
#define Pous1On !digitalRead (Pous1)
#define Pous2On !digitalRead (Pous2)
```

Pour les initialisation, il est nécessaire d'enclencher les résistances de tirage sur les poussoirs :

```
void setup () {
  pinMode (Led1, OUTPUT);
  pinMode (Led2, OUTPUT);
  pinMode (Pous1, INPUT_PULLUP); // avec résistance de tirage
  pinMode (Pous2, INPUT_PULLUP); // avec résistance de tirage
}
```

Finalement, voici un exemple, qui réalise la bascule Set/Reset :

```
void loop() {
  if (Pous1On) { // on teste d'abord le Set
    Led1On; Led2Off;
  }
  else { // le Reset est actif seulement si le Set ne l'est pas.
    if (Pous2On) {
      Led2On; Led1Off;
    }
  }
}
```

Brochage du MSP430 et Energia

Voici le brochage du MSP430 et des constantes déclarées dans Energia :

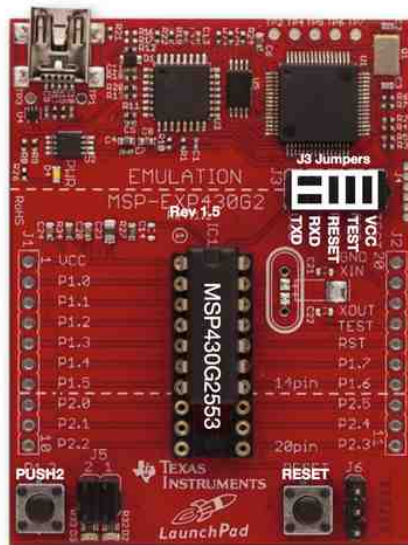


LaunchPad with MSP430G2553

Revision 1.5

Flash 16 KB
Serial Hardware

+3.3V				1
RED_LED		A0	P1_0	2
	RXD	A1	P1_1	3
	TXD	A2	P1_2	4
		A3	P1_3	5
		A4	P1_4	6
PUSH2	SCK (B0)	A5	P1_5	7
	CS (B0)		P2_0	8
			P2_1	9
			P2_2	10



Hardware
Pin number
IO
Serial UART
SPI
analogRead()
digitalRead() and digitalWrite()
digitalRead(), digitalWrite() and analogWrite()

20				GROUND
19	P2_6			XIN
18	P2_7			XOUT
17				TEST
16				RESET
15	P1_7	A7	SDA	MOSI (B0)
14	P1_6	A6	SCL	MISO (B0)
13	P2_5			GREEN_LED
12	P2_4			
11	P2_3			

Programmer des montage définitifs



Les fils pour la programmation peuvent aussi s'utiliser avec des montages définitifs, avec leurs composants soudés sur un VeroBoard ou sur un circuit imprimé.

Si le circuit imprimé a des trous métallisés, la programmation peut être faite même sans connecteur, comme sur l'image ci-contre, sur un dé électronique (un kit facile à monter, même avec des enfants dès 9 ans).