



La platine d'évaluation Explorer 700



[Mise à jour le 23/11/2019]

Sources

- Page du site **Joy-it** pour le module [EXPLORER 700](#) (description, installation, **image Raspbian** configurée pour le module, exemples, datasheets, [manuel](#), etc.)
- Le **schéma** du module Pioneer 600 (équivalent à l'explorer 700) est téléchargeable [ici](#)

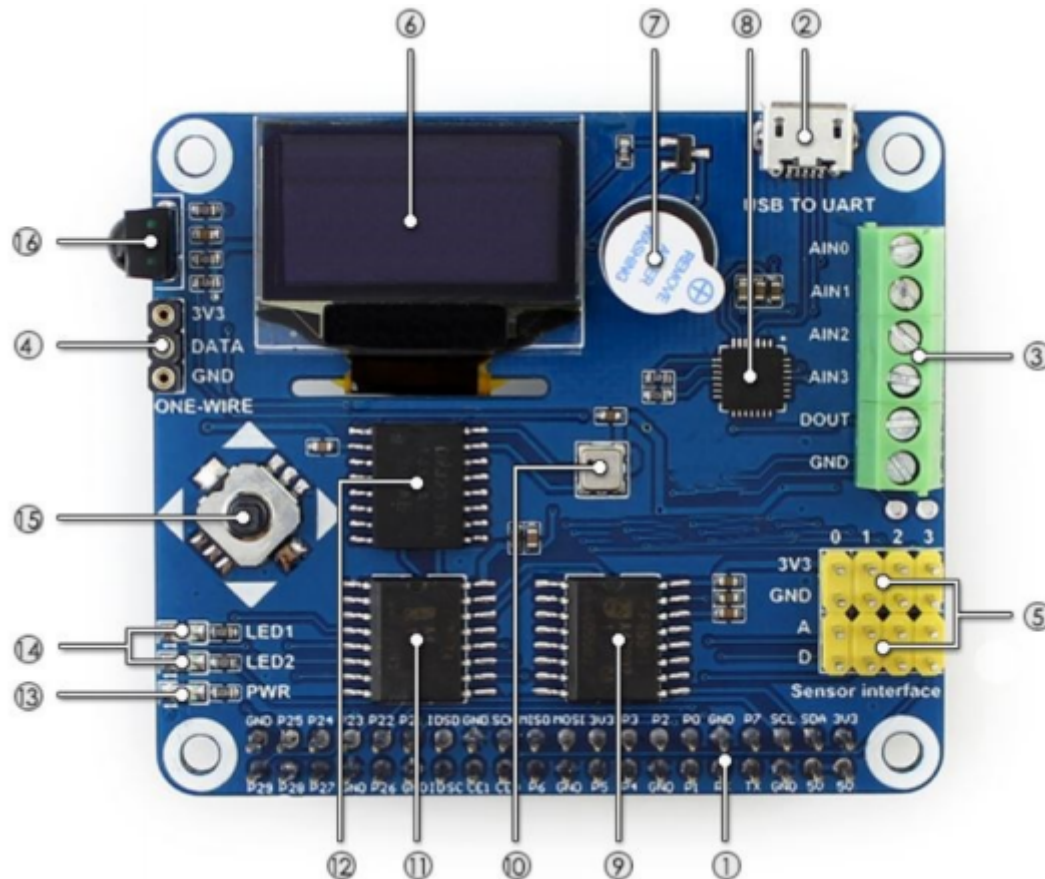
Distributeurs

[Gotronic](#).

1. Introduction

La platine d'évaluation **Explorer 700** regroupe des **capteurs**, un **afficheur graphique**, des **entrées et sorties** digitales et analogiques. Elle est enfichable sur le port GPIO des Raspberry Pi 2, 3 ou 4.

2. Description



1. **Interface GPIO** pour se connecter au Raspberry Pi
2. **USB TO UART** : contrôle le Raspberry Pi via un terminal série
3. **Interface "AD/DA"** via un bornier à vis
4. **Interface 1-WIRE** pour connecter des périphériques 1-WIRE tels que le capteur de température DS18B20
5. **Connecteurs** pour le raccordement de capteurs externes
6. **Écran OLED 0,96"** : contrôlé par un circuit SSD1306 (SPI)
7. **Buzzer**
8. **CP2102** (convertisseur USB <> UART)
9. **PCF8591** : convertisseur 8 bits "AD/DA" (I2C™)
10. **BMP280** : capteur de pression (I2C™)
11. **PCF8574** : Port I/O (I2C™)
12. **DS3231** : horloge temps réel (I2C™) - Support de pile au dos (pile non livrée)
13. **LED** de présence de l'alimentation
14. **LED** utilisateur (LED1, LED2)
15. **Mini Joystick**
16. **LFN0038K** : récepteur IR

3. Installation et tests

- **Télécharger** l'image Raspbian configurée pour l'explorer 700, les exemples de code et le manuel [ici](#).
- **Installer** l'image Raspbian pour l'explorer 700.
- Suivre les indications du manuel pour **tester les exemples** (Python).

4. Blink : un premier exemple en Python



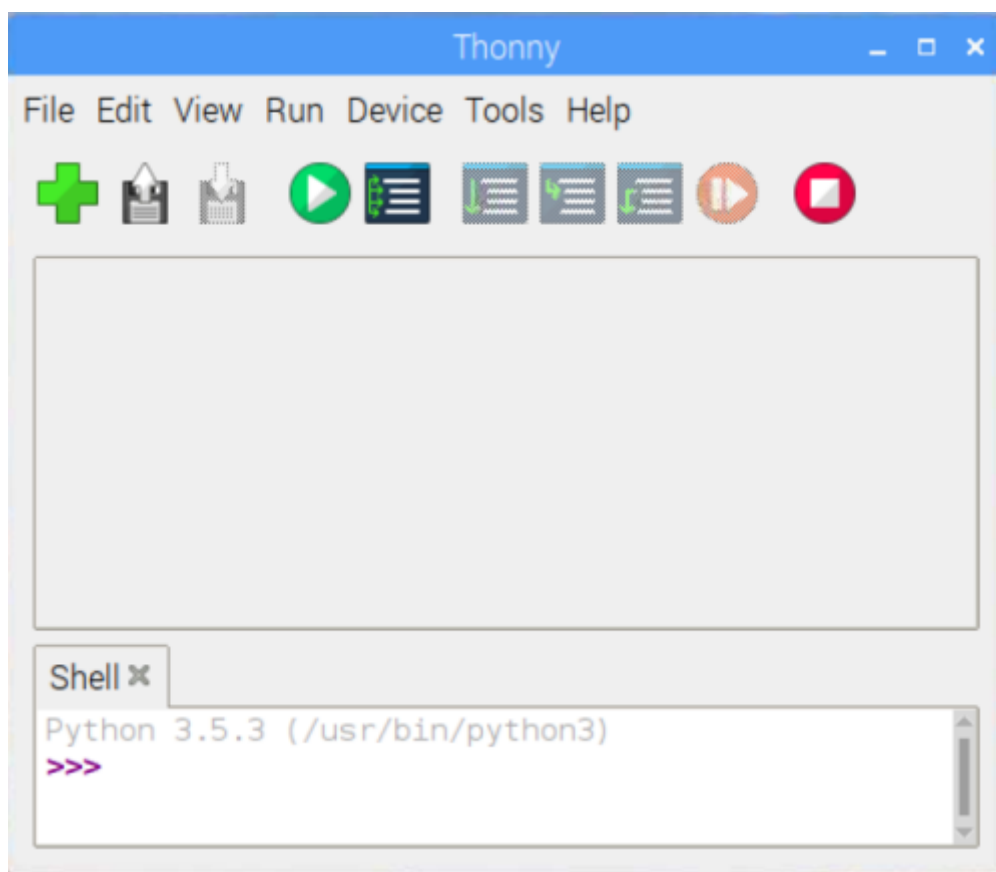
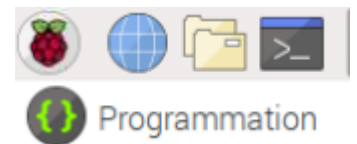
Objectif : faire clignoter la diode électroluminescente **LED1** de la carte Explorer 700 (repère 14 sur la photo ci-dessus).

- **Aspect matériel**

Pour commander cette LED, il est nécessaire d'identifier sa connexion au port GPIO du Raspberry Pi. D'après le [schéma](#) du fabricant, elle est connectée à la **broche 37 (GPIO26)** de ce connecteur. Voir la page "[Le port GPIO](#)". La LED s'éclaire avec un niveau logique "1".

- **Aspect logiciel**

Ouvrir l'éditeur **Thonny Python IDE** situé dans *Programmation* sur le Raspberry Pi. Une fenêtre doit s'ouvrir comme sur la copie d'écran ci-contre.



Créer un nouveau fichier ("+" ou **File** → **New**), copier le code ci-dessous et l'enregistrer avec le nom **led.py** dans **/home/pi**.


[blink.py](#)

```
import RPi.GPIO as GPIO # module de gestion du port GPIO
import time #module de gestion du temps
```

```
# Identification de la broche utilisée
LED = 37 # Une LED est connectée sur la broche 37
GPIO.setmode(GPIO.BOARD)
# Autre solution
#LED = 26
#GPIO.setmode(GPIO.BCM)

#Configuration de la broche en sortie
GPIO.setup(LED,GPIO.OUT)

while True:
    GPIO.output(LED,GPIO.HIGH) # Activation de la LED
    time.sleep(1) # Attente 1s
    GPIO.output(LED,GPIO.LOW) # Désactivation de la LED
    time.sleep(1) # Attente 1s
```

Exécuter le code en cliquant sur le bouton suivant 

Comportement attendu : la LED1 de la carte Pioneer 600 doit clignoter.

Autre solution pour exécuter le code : **ouvrir** une console (**LXTerminal**) et se placer dans /home/pi. Exécuter la commande suivante :



`cmd1.bash`

```
python led.py
```

From:

<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<http://webge.fr/dokuwiki/doku.php?id=raspberrypi:matériels:pioneer600>

Last update: **2021/08/11 09:19**

