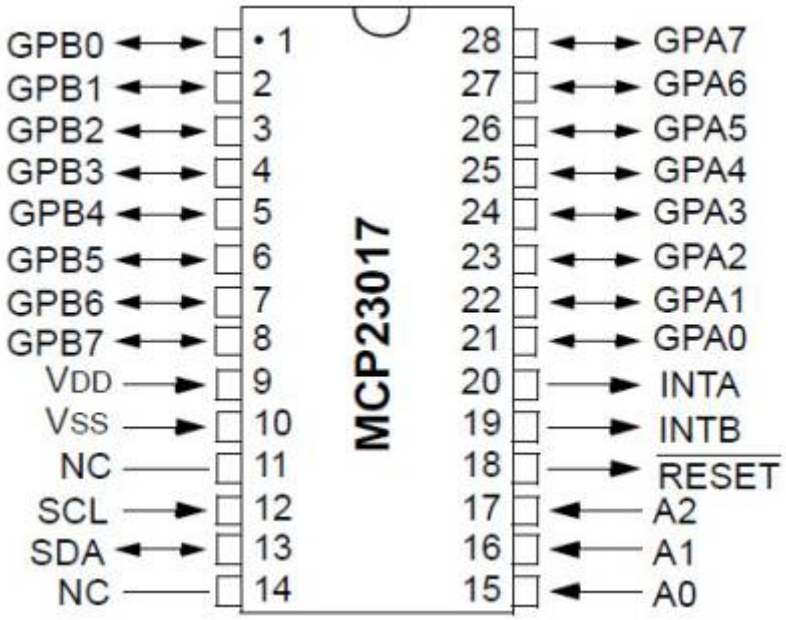
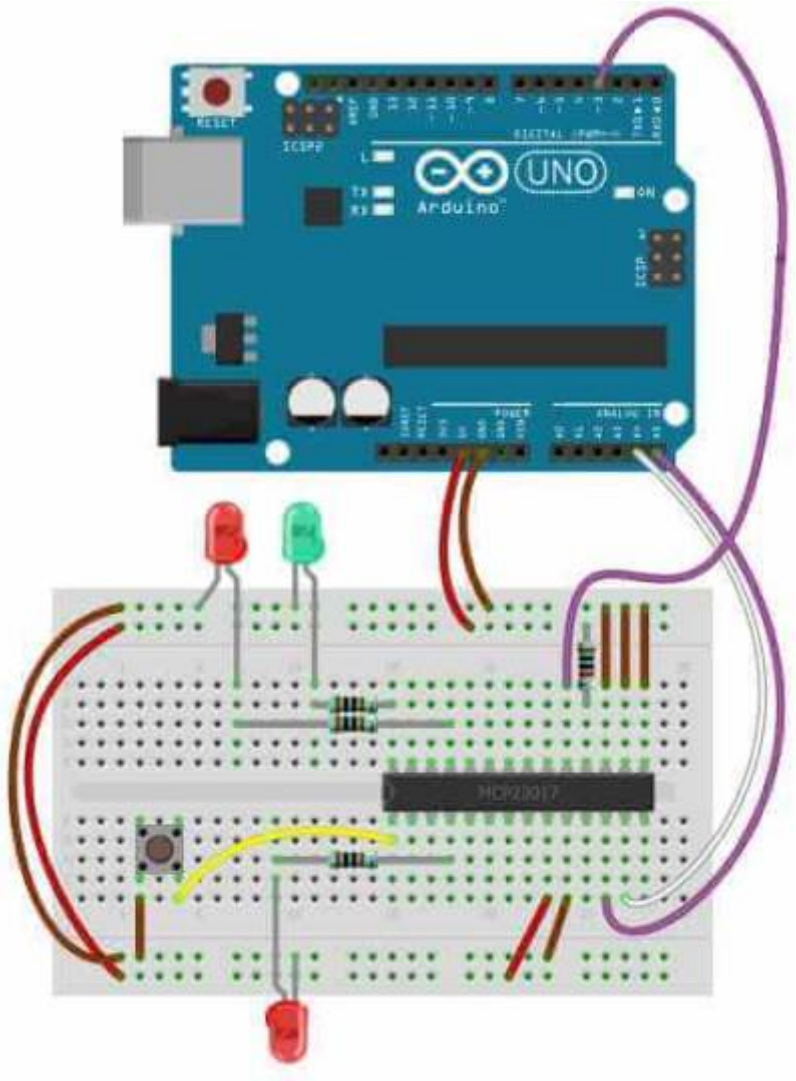
MCP23017



MCP23017 16 entrées/sorties sur un arduino

MCP23017 En Entrée

MCP23017 EN



mcp23017.ino

```
// MCP23017 Example: Slow key press reaction.
//
// Toggle LEDs and detect keypress.
//
// Example code showing slow reaction of 'button'
// LED to keypress. Leading into why interrupts
// are useful (See next example).
//
// Copyright : John Main
// Free for non commercial use.

#include <Wire.h>
#include <Adafruit_MCP23017.h>

#define MCP_LED1 7
#define MCP_INPUTPIN 8
#define MCP_LEDTOG1 11
#define MCP_LEDTOG2 4
```

```
Adafruit_MCP23017 mcp;

void setup() {
  mcp.begin();      // Default device address 0

  mcp.pinMode(MCP_LEDTOG1, OUTPUT);  // Toggle LED 1
  mcp.pinMode(MCP_LEDTOG2, OUTPUT);  // Toggle LED 2

  mcp.pinMode(MCP_LED1, OUTPUT);     // LED output
  mcp.digitalWrite(MCP_LED1, HIGH);

  mcp.pinMode(MCP_INPUTPIN,INPUT);   // Button i/p to GND
  mcp.pullUp(MCP_INPUTPIN,HIGH);     // Puled high to ~100k
}

// Alternate LEDTOG1 and LEDTOG2.
// Transfer pin input to LED1.
void loop() {

  delay(300);

  mcp.digitalWrite(MCP_LEDTOG1, HIGH);
  mcp.digitalWrite(MCP_LEDTOG2, LOW);

  delay(300);

  mcp.digitalWrite(MCP_LEDTOG1, LOW);
  mcp.digitalWrite(MCP_LEDTOG2, HIGH);

  // Transfer input pin state to LED1
  if (mcp.digitalRead(MCP_INPUTPIN)) {
     mcp.digitalWrite(MCP_LED1,HIGH);
  } else {
     mcp.digitalWrite(MCP_LED1,LOW);
  }

}
```

Bibliothèque de logiciels et versions

Version IDE Arduino La dernière version: 1.8.13 (12/2020)

Bibliothèque Adafruit Bibliothèque MCP23017 pour arduino

```
[[http://downloads.arduino.cc/libraries/github.com/blemasle/MCP23017-2.0.0.zip|• Bibliothèque Adafruit MCP23017 2.0.0]]
```

Ceci est facilement installé à partir de l'IDE Arduino. Si vous ne voyez pas la bibliothèque comme une entrée lorsque vous cliquez sur les menus, installez la bibliothèque comme suit: Esquisse → Inclure la bibliothèque Sélectionnez ensuite gérer les bibliothèques: Esquisse → Inclure la bibliothèque → Gérer les bibliothèques ... Recherchez et installez <nom de la bibliothèque> en utilisant le formulaire “Filtrer votre recherche”. Fonctionnement de la bibliothèque Définition des broches pour la bibliothèque MCP23017 Remarque: dans la bibliothèque, les broches sont étiquetées de 0 à 15 où: la broche 0 est le bit 0 du port A la broche 7 est le bit 7 du port A la broche 8 est le bit 0 du port B la broche 15 est le bit 7 du port B Fonctions de contrôle E / S du MCP23017 E / S à bit unique Des fonctions membres similaires aux commandes de broches sur l'Arduino sont utilisées pour contrôler les broches MCP23017: mcp.pinMode (0, SORTIE); mcp.digitalWrite (0, HAUT); mcp.digitalRead (0); Connexions Connexions Netlist simples La netlist et le diagramme suivants vous montrent comment la connexion du MCP23017 à l'Arduino est très simple. Connectez la broche n ° 12 de l'extenseur à Arduino Analog 5 (horloge i2c) Connectez la broche n ° 13 de l'extenseur à Arduino Analog 4 (données i2c) Connectez la broche n ° 19 de l'extension à la broche 3 d'Arduino (entrée d'interruption). Connectez les broches n ° 15, 16 et 17 de l'extension à la masse Arduino (sélection d'adresse) Connectez la broche n ° 9 de l'extenseur à Arduino 5V (alimentation) Connectez la broche n ° 10 de l'extenseur à la terre Arduino (masse commune) Connectez la broche n ° 18 de l'expanseur à travers une résistance ~ 10 kohm à 5 V (broche de réinitialisation, actif bas). Connectez la broche # 28 de l'extenseur à l'extrémité + ve d'une LED puis à une résistance ~ 1kohm à GND (MCP_LED1). Connectez la broche # 26 de l'extenseur à l'extrémité + ve d'une LED puis à une résistance ~ 1kohm à GND (MCP_LEDTOG1). Connectez la broche n ° 4 de l'extenseur à l'extrémité + ve d'une LED puis à une résistance ~ 1kohm à GND (MCP_LEDTOG2). Connectez la broche n ° 1 du module d'extension à un bouton-poussoir normalement ouvert qui se connecte ensuite à GND (MCP_INPUTPIN). Remarque: les pullups sont activés pour les broches I2C dans la bibliothèque “Wire” et ne sont donc pas affichés dans la connexion ou la disposition du circuit ci-dessus (ci-dessous). Ils ont une valeur élevée (probablement 50k ~ 100k), donc pour un front montant plus rapide sur les signaux I2C, utilisez des résistances d'extraction physiques de valeur inférieure qui remplaceront la valeur élevée.

From:
https://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
https://chanterie37.fr/fablab37110/doku.php?id=start:arduino:mcp23017&rev=1606995112

Last update: **2023/01/27 16:08**