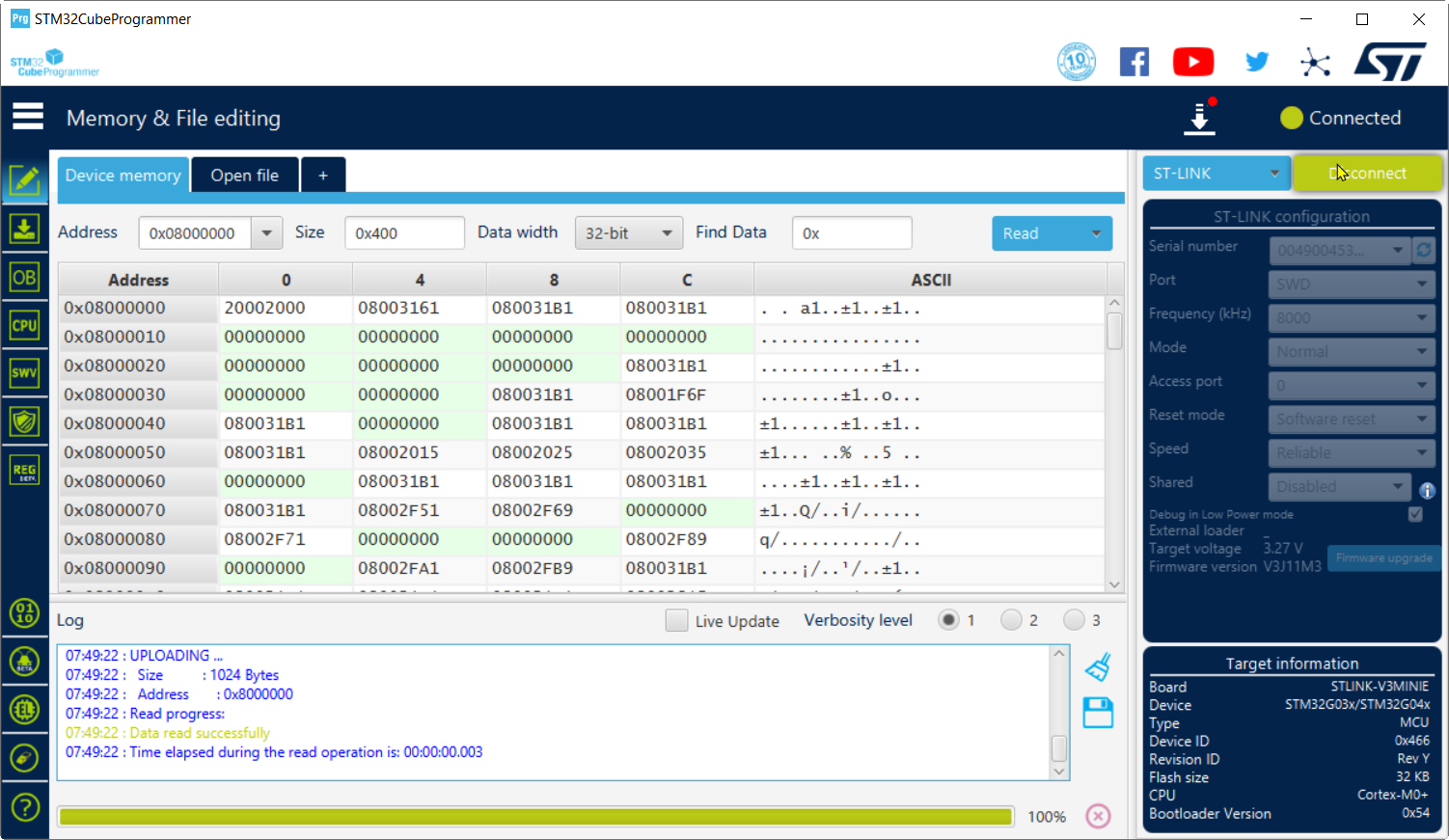
Premier test

Une image contenant texte

Description générée automatiquement

On alimente la cible en 5V et on lance STM32CubeProg

Ca connecte : on est bon.



VScode

Une image contenant texte

Description générée automatiquement

Une image contenant table

Description générée automatiquement

On a 2 types de cartes en STM32G030 (32k de flash et 8k de RAM)

#include <Arduino.h>

#include "rgb\_lcd.h"

**HardwareSerial** **DebugSerial**(**PB7**, **PB6**); *// compiles, can also use (USART1)*

const int LED1 = **PA11**;

const int LED2 = **PC6**;

const int BP = 2;

const int BP1 = **PA0**;

const int BP2 = **PA1**;

const int POT = **PA5**;

int valeur\_potentiometre;

**rgb\_lcd** lcd;

void **setup**()

{

*// put your setup code here, to run once:*

**pinMode**(LED1, **OUTPUT**);

**pinMode**(LED2, **OUTPUT**);

**Serial**.**begin**(9600);

**pinMode**(BP1, **INPUT\_PULLUP**);

**pinMode**(BP2, **INPUT\_PULLUP**);

  DebugSerial.**begin**(9600);

    lcd.**begin**(16, 2);

*// Print a message to the LCD.*

    lcd.**print**("hello, world!");

**delay**(1000);

}

void **loop**()

{

   static int cpt = 0;

*// put your main code here, to run repeatedly:*

**digitalToggle**(LED1);

**digitalToggle**(LED2);

**delay**(200);

**Serial**.**print**("serial:");

**Serial**.**println**(cpt);

    cpt++;

    valeur\_potentiometre = **analogRead**(POT);

    int tension=valeur\_potentiometre\*32;

  lcd.**clear**();

  lcd.**blinkLED**();

  lcd.**setCursor**(0,0);

  lcd.**print**("pot=");

  lcd.**print**(tension/10000);

  lcd.**print**(".");

  lcd.**print**(tension%10000);

  lcd.**print**(" V");

  lcd.**setCursor**(0,1);

      lcd.**print**("BP1=");

      lcd.**print**(**digitalRead**(BP1));

      lcd.**print**("  BP2=");

      lcd.**print**(**digitalRead**(BP2));

**Serial**.**print**("cpt=");

**Serial**.**println**(cpt);

      DebugSerial.**print**("Debug:");

      DebugSerial.**print**(**digitalRead**(BP1));

      DebugSerial.**print**(" ");

      DebugSerial.**print**(**digitalRead**(BP2));

      DebugSerial.**print**(" ");

    DebugSerial.**println**(valeur\_potentiometre);

}

*/\**

*void poubelle(){*

*lcd.clear();*

*lcd.blinkLED();*

*lcd.print("pot=");*

*lcd.print(valeur\_potentiometre);*

*// valeur\_potentiometre = analogRead(POT);*

*DebugSerial.println(valeur\_potentiometre);*

*int tension = valeur\_potentiometre \* 3300 / 4096;*

*lcd.clear();*

*lcd.blinkLED();*

*lcd.print("pot=");*

*lcd.print(tension/100);*

*lcd.print(".");*

*lcd.print(tension%100);*

*Serial.println(tension);*

*lcd.print(" V");*

*digitalWrite(LED1, !digitalRead(BP1));*

*digitalWrite(LED2, !digitalRead(BP1));*

*delay(100);*

*}\*/*

*/\*#include "Wire.h"*

*void setup(){*

*Wire.begin();*

*Serial.begin(9600);*

*}*

*void loop(){*

*byte error, address;*

*int nDevices;*

*Serial.println("Scanning...");*

*nDevices = 0;*

*for(address = 8; address < 127; address++ ){*

*Wire.beginTransmission(address);*

*error = Wire.endTransmission();*

*if (error == 0){*

*Serial.print("I2C device found at address 0x");*

*if (address < 16)*

*Serial.print("0");*

*Serial.print(address,HEX);*

*Serial.println(" !");*

*nDevices++;*

*}*

*else if (error == 4) {*

*Serial.print("Unknow error at address 0x");*

*if (address < 16)*

*Serial.print("0");*

*Serial.println(address,HEX);*

*}*

*}*

*if (nDevices == 0)*

*Serial.println("No I2C devices found\n");*

*else*

*Serial.println("done\n");*

*delay(5000);*

*}\*/*