

STM32Duino__prog_2.txt

Jednoduché programy pro základní seznámení s STM32Duino pro klub ETC22 na ČVUT - FEL J.Fischer 2.12.2024,
Jazyk Wiring pro STM32G030J6 s 8-mi vývody v pouzdře S08
Zde se zatím využívá základní varianta programování, takže lze využít literaturu pro programování Arduino
všechny piny (c. 1, 4, 7) lze využít jako digitální vstupy/vystupy nebo i analogové vstupy
pin c.8 PB5 s funkcí BOOT můžeme využít jen jako digitální výstup pro LED, protože zde je navíc připojen rezistor 470 R na VDD (+3,3 V), který zajistí po zapnutí napájení přechod do BOOT
Programy nevyužívají rozšířené možnosti programování STM32v STM32Duino, jako je napr. využití dalších periférií, programování přes registry procesoru,....

Pokud by se využily programy určené přímo pro Arduino, pak platí vztah přiřazení
pin c.1 PB_7, D13 /A9 (konverze STM32 na Arduino);
pin c.4 PA_0, // D0 /A0 ;
pin c.7 PA_13, // D6/A5
pin c.8 PB_5, // D11 (v naší desce můžeme využít pouze jako výstup)

na pouzdře STM32G030 je tzv Multibond, tedy na jeden pin je připojeno více kontaktů z čipu
(z nichž jako výstup můžeme využít pouze jediný)

pin c.1 PB_7, PB8, PB9,PC14
pin c.4 PA_0, PA1, PA2, PF2-NRST
pin c.7 PA_13
pin c.8 PB_5, PB6, PA14-BOOT0, PA15

Pozn. Pin. c.4 má také funkci reset, která je aktivována pouze při zapnutí napájení,
pokud pro zapnutí napájení zde bude nízká úroveň napětí " LOW, procesor zůstane "viset" (zastaví se)
ve stavu reset, dokud se úroveň LOW neodpojí

Na PIN c.4 proto připojujeme modrou LED1, na níž bude (působením vnitřního pull-up rezistoru) v okamžiku reset dostatečně velké napětí, které nezpůsobí pozdržení ve stavu RESET

Celkový reset zajišťujeme vypnutím a zapnutím napájení.
Pro spolehlivé vybití el.kondenzátoru zapojíme mezi GND a VDD (+3,3 V) vybíjecí rezistor (napr. 470 Ohmů až 2k2)
Pro uvedení do stavu BOOT na pin c. 8 připojíme pull-up rezistor 470 Ohmů (stejná hodnota, jako má rezistor u LED připojené na pin c.)

UART Tx na pinu c. 4
UART Rx na pinu c.5

základní blikací (červená) LED na pinu c.8 PB5
doplňková MODRA LED1 na pinu č.4 PA0

BUZZER přes rezistor 470 R a 2k2 pin c. 1 PA13

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXX
```

```
// PROGRAM_1  
/*
```

```
STM32G030J6 Blikání LED,  
připojena na pin. c. 8, PB5, jako Arduino D11  
Stránka 1
```

```

STM32Duino__prog_2.txt
program v jazyce wiring - stejné jako pro Arduino
*/

// inicializace, probehne jednou

void setup() { // inicializace digitaliho pinu LED jako vystupu
  pinMode(D11, OUTPUT);
}

// nekonecna smycka

void loop() {
  digitalWrite(D11, HIGH); // LED ON (privede vysokou uroven HIGH napeti High -u
STM32 to je cca 3,3 V
  delay(500); // cekej 500 milisekund
  digitalWrite(D11, LOW); // LED OFF privede nizkou uroven napeti LOW
  delay(500); // cekej 500 milisekund
}

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_2

/*
STM32G030J6 Blikani dvema LED v protifazi
pripojene na pin. c. 8, PB5 (jako Arduino D11) a pin c.4 PA0 jako (Arduino D0)
program v jazyce wiring - stejné jako pro Arduino
*/

# define LED PB5 // PB5 si pojmenujeme jako LED
# define LED1 PA0 // PA0 si pojmenujeme jako LED1

// inicializace, probehne jednou
void setup() {
  pinMode(LED, OUTPUT); // pro funkci "tone" není třeba inicializovat, ale
  digitalWrite(LED_, LOW)
  pinMode(LED1, OUTPUT); // pro funkci "tone" není třeba inicializovat, ale
  digitalWrite(LED_, LOW)
}

// nekonecna smycka
void loop() {
  digitalWrite(LED, HIGH); // LED ON (privede vysokou uroven napeti HIGH (cca
3,3 V)
  digitalWrite(LED1, LOW); // LED1 OFF (privede nizkou uroven napeti LOW (cca 0
V)
  delay(500); // cekej 500 milisekund
  digitalWrite(LED, LOW);
  digitalWrite(LED1, HIGH);
  delay(500); // cekej 500 milisekund
}

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_3

/* Blikani LED na PB5 ve stupnich jasu pomoci "analogwrite" meni jas LED na PB5
na č. 8
v 5 stupnich, LED1 na PA0 blikne na zacatku cyklu */
# define LED PB5 // pin č. 8, PB5, take jako 11 Ardu
# define LED1 PA0 // pin c. 4 PA0 si pojmenujeme jako LED1

// inicializace, probehne jednou
void setup() {
  pinMode(LED, OUTPUT); // pro funkci analogwrite není třeba inicializovat, ale

```

STM32Duino__prog_2.txt

```
digitalWrite(LED_, LOW)
pinMode(LED1, OUTPUT); // pro funkci "tone" neni třeba inicializovat, ale
digitalWrite(LED_, LOW)
}
```

```
void loop() {
digitalWrite(LED1, HIGH); // bliknutí LED1 na PA0 na začátku cyklu
delay(1000); // prodleva v cyklu pouzita 1sekunda
digitalWrite(LED1, LOW);
analogWrite(LED, 1);
delay(1000);
analogWrite(LED, 5);
delay(1000);
analogWrite(LED, 20);
delay(1000);
analogWrite(LED, 70);
delay(1000);
analogWrite(LED, 250);
delay(1000);
analogWrite(LED, 0);
}
```

```
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
```

```
// PROGRAM_4
```

```
/* Blikání LED na PB5 pomocí funkce tone
LED1 na PA0 blikne na konci cyklu */
```

```
# define LED PB5 // pin č. 8, PB5, také jako 11 Ardu
# define LED1 PA0 // pin č. 4 PA0 si pojmenujeme jako LED1
```

```
// inicializace, proběhne jednou
```

```
void setup() {
pinMode(LED1, OUTPUT);
// pinMode(LED, OUTPUT); // pro funkci "tone" neni třeba inicializovat,
}
```

```
void loop() {
tone(LED, 8, 1500); // LED bliká frekvencí 8 Hz po dobu 1500 ms
digitalWrite(LED1, LOW); // LED1 ON
delay(1500); // čeká 1500 milisekund
digitalWrite(LED1, HIGH); // LED1 OFF
delay(1500); // čeká 1500 milisekund
}
```

```
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
```

```
// PROGRAM_5
```

```
/* Blikání LED na PB5 pomocí funkce tone, LED1 na PA0 blikne na konci cyklu,
po ukončení blikání vynutíme zhasnutí LED na PB0, i kdyby na pinu LED zůstalo
HIGH z funkce tone */
```

```
# define LED PB5 // pin č. 8, PB5, musíme inicializovat, abychom mohli vynutit
zhasnutí po noTone
# define LED1 PA0 // pin č. 4, PA0
```

```
void setup() {
pinMode(LED1, OUTPUT);
pinMode(LED, OUTPUT);
}
```

STM32Duino__prog_2.txt

```
void loop() {
  tone(LED, 8); // LED bliká frekvencí 8 Hz dokud se nevypne blikání
  digitalWrite(LED1, LOW); // LED1 ON
  delay(1500); // čekej 1500 milisekund
  noTone(LED); // konec funkce tone - vypnutí blikání
  digitalWrite(LED, LOW); // vynutíme zhasnutí LED po ukončení funkce tone,
  // pokud by tam zůstala úroveň HIGH v okamžiku po
  provedení noTone
  digitalWrite(LED1, HIGH); // LED1 OFF
  delay(1500); // čekej 1500 milisekund
}
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_6

/* bliká LED a vypisuje počet cyklu na UART a terminal
# define LED PB5 // pin č. 8, PB5 S08 STM32G030J6
# define LED1 PA0 // pin č. 4, PA0 S08 STM32G030J6

HardwareSerial Serial1 (PA10_R, PA9_R); // UART na remap. pinech č. 5 a 6 na
STM32G030J6 pouzdro S08
// kanál UART1 je na branach PA10 a PA9, ale pro pouzdro S08 musime oznacit,
ze se ma pouzit remapovani

void setup() {
  Serial1.begin(9600);
  pinMode(LED, OUTPUT); //
  pinMode(LED1, OUTPUT);
  delay(3000); // prodleva 3 sekundy , abychom stacili zapnout serial
terminal // u verze Arduino 2.3.3 muze byt serial interni terminal zapnut
stale
  Serial1.println("AHOJ");
}
int pocet= 0;
void loop() {
  pocet ++;
  digitalWrite(LED, HIGH); // blikni LED1
  delay(500);
  digitalWrite(LED, LOW); // zhasni LED1
  delay(500);

  Serial1.print("pocet cyklu= ");
  Serial1.println(pocet);
}
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_7

/*
Nastavení LED1 podle UARTU, blikání LED podle UARTU přijímá znaky z UART
-psané na klávesnici , při "1" rozsvítí LED1 na PA0, ostatní zhasne LED1 na PA0,
LED na PB5 blikne podle čísla, 1 blikne 1x, 2 blikne 2x, znaky menší než "0"
bliknou 1x,
znaky vyšší než 9 bliknou 9x, písmena bliknou 9x
vypisuje znak, který přišel */

# define LED PB5 // pin č. 8, PB5 S08 STM32G030J6
# define LED1 PA0 // pin č. 1, PA0 S08 STM32G030J6

HardwareSerial Serial1 (PA10_R, PA9_R); // UART na remap. pinech č. 5 a 6 na
STM32G030J6 pouzdro S08
// kanál UART1 je na branach PA10 a PA9, ale pro pouzdro S08 musime oznacit,
ze se ma pouzit remapovani
```

STM32Duino__prog_2.txt

```

void setup() {
  Serial1.begin(9600);
  pinMode(LED, OUTPUT); // PB5 pin. c.8
  pinMode(LED1, OUTPUT); // PA0 pin c.4
}

int poc= 0;

char znak =0;
void loop()
{
  if (Serial1.available() >0)
  {
    Serial1.print ("prislo=");
    znak = Serial1.read ();
    Serial1.println (znak);
    if (znak == '1') // nebo (znak == 0x31)
    {
      digitalWrite (LED1, HIGH);
    }
    else
    {
      digitalWrite (LED1, LOW);
    }
  }
  if (znak <= '1')
  {
    poc= 1;
  }
  else if
  (znak >= '9')
  {
    poc= 9;
  }
  else
  {
    poc= znak - '0';
  }
  blikej (poc);
}
} // nize je nase funkce blikej
void blikej (int opak) {
  while ((opak)>0) {

    digitalWrite (LED, HIGH); // blikni LED
    delay(500);
    digitalWrite (LED, LOW); // zhasni LED
    delay(500);
    opak--;
  }
}

XXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXX

// PROGRAM_8

/*
Na zacatku blikne LED a nypise AHOJ,
Nastaveni LED1 podle UARTU, blikani LED podler UARTU prijima znaky z UART
-psane na klavesnici , pri "1" rozsvoti LED1 na PA0, ostani zhasne LED1na PA0,
LED na PB5 blikne podle cisla, 1 blikne 1x, 2 blikne 2x, znaky mensi nez "0"
bliknou 1x,
znaky vyssi nez 9 bliknou 9x, pismena bliknou 9x
vypisuje znak, ktery prisel */

# define LED PB5 // pin č. 8, PB5 S08 STM32G030J6
# define LED1 PA0 // pin č. 4, PA0 S08 STM32G030J6

HardwareSerial Serial1 (PA10_R, PA9_R); // UART na remap. pinech č. 5 a 6 na
STM32G030J6 pouzdro S08
// kanal UART1 je na branach PA10 a PA9, ale pro pouzdro S08 musime oznacit,
ze se ma pouzit remapovani

```

STM32Duino__prog_2.txt

```

void setup() {
  Serial1.begin(9600);
  pinMode(LED, OUTPUT); // PB5 pin. c.8
  pinMode(LED1, OUTPUT); // PA0 pin c.4
  digitalWrite(LED, LOW); // LED1 ON
  delay(1500); // cekej 1500 milisekund
  digitalWrite(LED, HIGH); // LED1 OFF
  delay(1500); // cekej 1500 milisekund
  digitalWrite(LED, LOW); // LED1 ON
  Serial1.print ("AHOJ");
}

int poc= 0; // nase promenna pro pocet opakovani
char znak =0; // nase promenna pro ulozeni prijateho znaku

void loop()
{
  if (Serial1.available() >0)
  {
    Serial1.print ("prislo=");
    znak = Serial1.read ();
    Serial1.println (znak);
    if (znak == '1') // nebo (znak == 0x31)
    {
      digitalWrite (LED1, HIGH);
    }
    else
    { digitalWrite (LED1, LOW);
    }
    if (znak <= '1')
    { poc= 1; }
    else if
    (znak >= '9')
    { poc= 9; }
    else
    { poc= znak - '0'; }
    blikej (poc);
  }
} // tady je konec hlavni smycky, nize je nase funkce "blikej", vstupni parametr
ma pocet bliknuti
void blikej (int opak)
{
  while ((opak)>0) // opakuje, dokud je promenna opka vetsi nez nula
  {
    digitalWrite (LED, HIGH); // blikni LED
    delay(250);
    digitalWrite (LED, LOW); // zhasni LED
    delay(250);
    opak--; // pocitadlo bliknuti
  }
}

XXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_9

/* vPipani funkci pip, ktrera vyuziva funkci "tone", pri kazdem pipnuti blikne
take LED- na PB5
LED1 na PA0 na funkci blik,
*/
# define LED PB5 // pin č. 8, PB5 S08 STM32G030J6
# define LED1 PA0 // pin č. 4, PA0 S08 STM32G030J6
# define BUZ PA13 // pin č. 7, PA13 S08 STM32G030J6
void setup() {

```

STM32Duino__prog_2.txt

```

pinMode(LED, OUTPUT);
pinMode(LED1, OUTPUT);
}
void pip (int vyska, int delka) /*nase funkce pro pisknuti na pinu BUZ ,
                                to je PA13, se vstupnimi parametry */
{
tone(BUZ, vyska, delka);
digitalWrite(LED, HIGH);
delay(delka/2);
digitalWrite(LED, LOW);
delay(delka/2);
}

void blik (void) //nase funkce ( bez vstupnich parametru) pro bliknuti LED na
PB5, pin. c. 8
{
digitalWrite(LED1, HIGH); //
delay(100);
digitalWrite(LED1, LOW); // blikni LED1
delay(300);
}

void loop() {

blik (); // volame funkci blik
pip (500, 500);

blik();
pip (1000, 500); /* volame funkci pip a predavame parametry 1000 Hz,500 ms*/
blik();
pip (1500, 500);
blik();
pip (2000, 400);
blik();
pip (2500, 400);
blik();
pip (3000, 400);
blik();
}

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

// PROGRAM_10

/* STM32G030J6 v SO8, Pipani s funkci , na zacatku blikne LED1 na PA0 pin. c 4.
pak pisti buzzer nizsi frekvenci a sviti LED PB5 na c.8, pak pisti vyssi
frekvenci a LED zhasne
*/

# define LED PB5 // pin č. 8, PB5 SO8 STM32G030J6
# define LED1 PA0 // pin č. 4, PB5 SO8 STM32G030J6
# define BUZ PA13 // pin č. 7, PA13 SO8 STM32G030J6
void setup() {

pinMode(LED, OUTPUT);
pinMode(LED1, OUTPUT);
}

void loop() {
digitalWrite (LED1, HIGH);
delay(200);
digitalWrite (LED1, LOW);
delay(200);
}

```

STM32Duino__prog_2.txt

```
tone (BUZ, 1000);  
digitalwrite (LED, HIGH);  
delay(500);  
noTone (BUZ);  
delay(500);  
tone (BUZ, 1500);  
digitalwrite(LED, LOW);  
delay(500);  
noTone (BUZ);  
delay(500);
```

```
}
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```