**SLOVENKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE**

**FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

**Vnorene systémy**

Cvičenie 4: AD prevodník

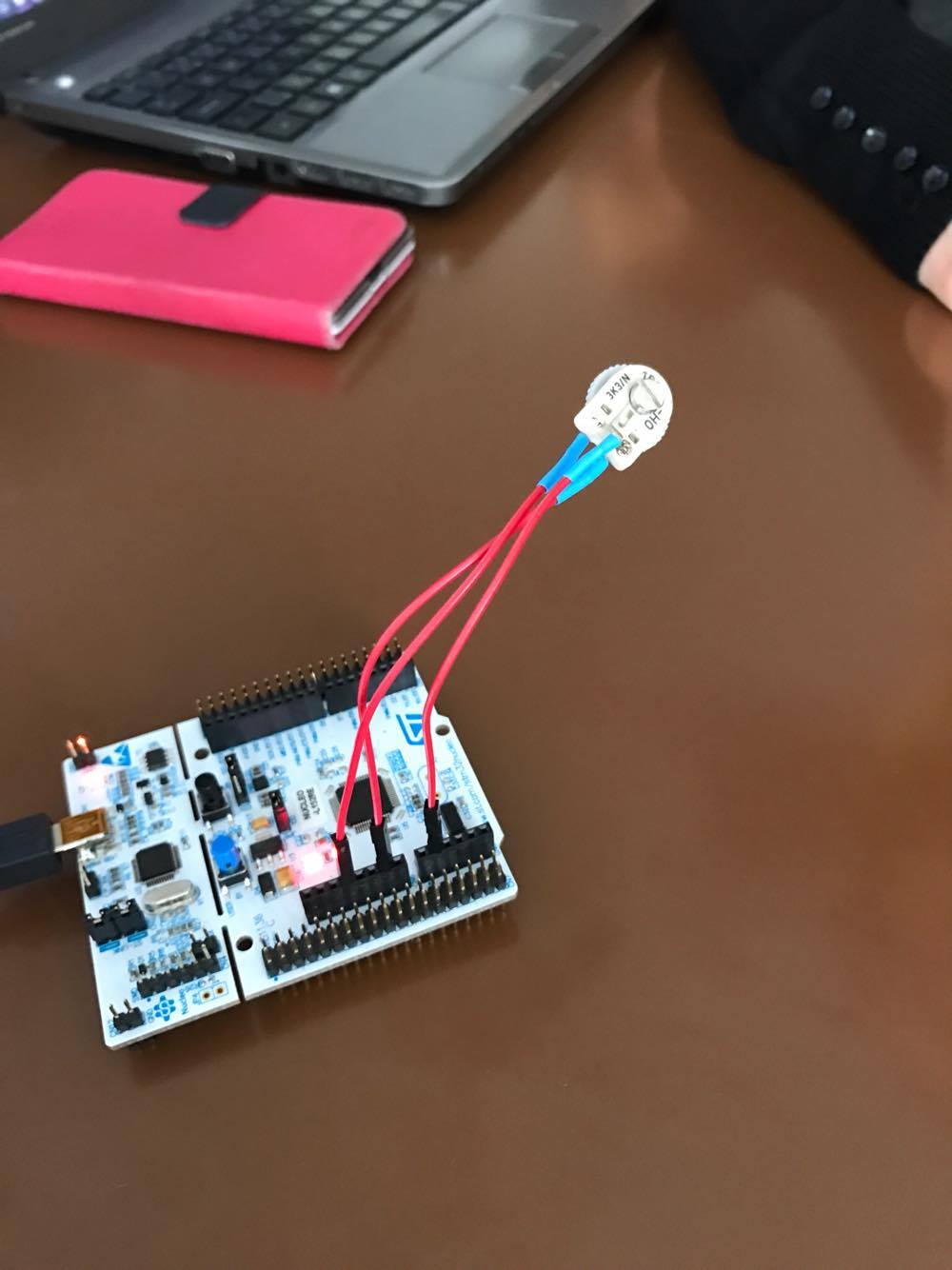
Cvičenie o 13.00

Zimný semester 16/17 Bálint Domonkos  
1.ročník ing., Robotika

# Vypracovanie

## Úloha 1. – Zapájanie

Úlohou bolo pripojiť k vývojovej doske potenciometer.



Úprava vzorového kódu pre nastavenie ADC.

Nastavenie ADC na pin

GPIO\_InitStructure.GPIO\_Pin = GPIO\_Pin\_1 ;

Nastavenie channelu.

ADC\_RegularChannelConfig(ADC1, ADC\_Channel\_1, 1, ADC\_SampleTime\_16Cycles);

Spustenie blikania a sledovanie hodnoty z potenciometra.

**int** **main**(**void**)

{

adcInit();

initLED();

uint16\_t ADvalue=(uint16\_t)0;

**while**(1){

ADC\_SoftwareStartConv(ADC1);

**while**(!ADC\_GetFlagStatus(ADC1, ADC\_FLAG\_EOC)){}

ADvalue = ADC\_GetConversionValue(ADC1);

}

**return** 0;

}

Funkcia inicializovania ledky.

**void** **initLED**(**void**){

RCC\_AHBPeriphClockCmd(RCC\_AHBPeriph\_GPIOA,*ENABLE*);

GPIO\_InitTypeDef gpioInitStruc;

gpioInitStruc.GPIO\_Mode = *GPIO\_Mode\_OUT*;

gpioInitStruc.GPIO\_OType = *GPIO\_OType\_PP*;

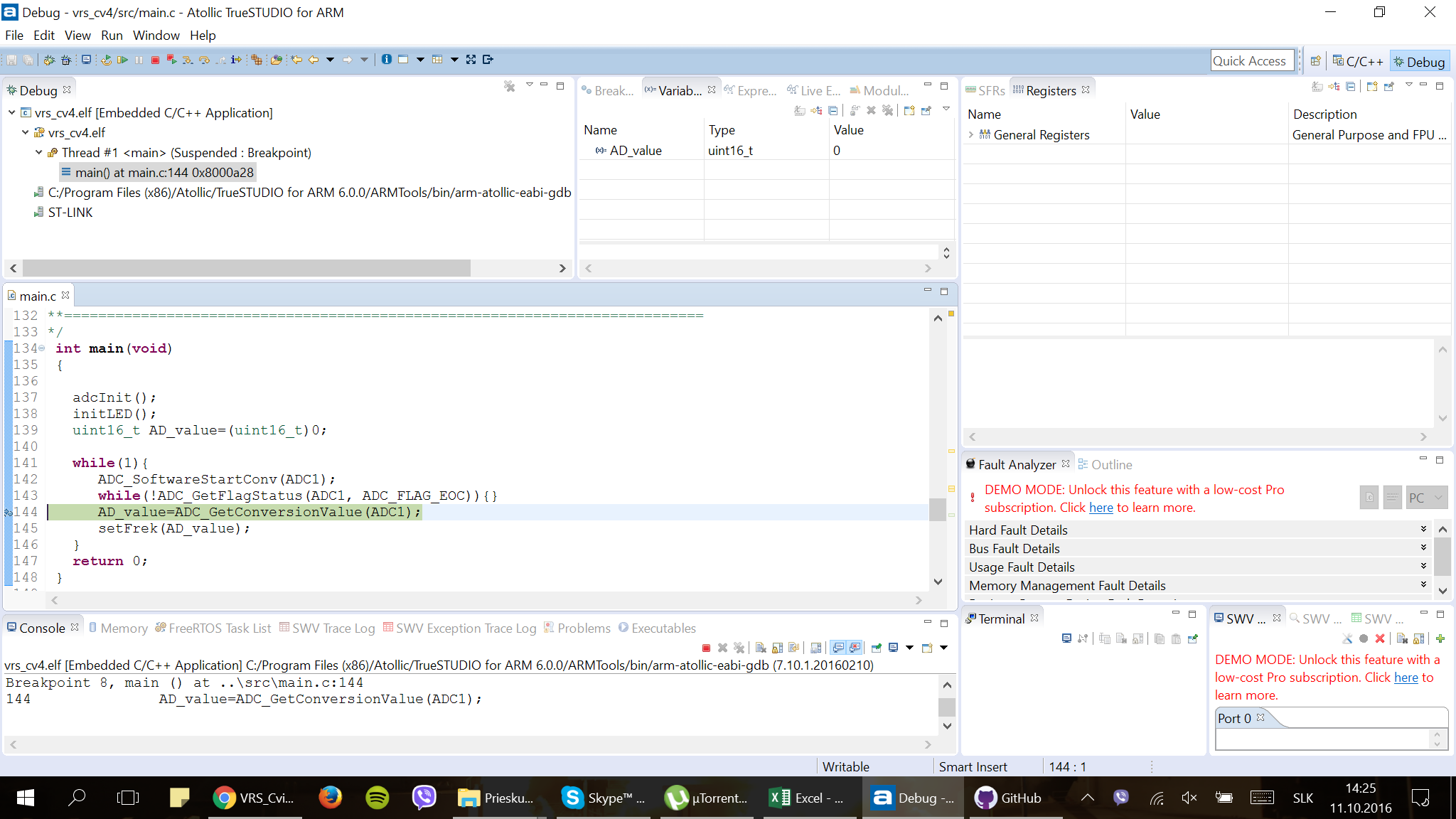
gpioInitStruc.GPIO\_Pin = GPIO\_Pin\_5;

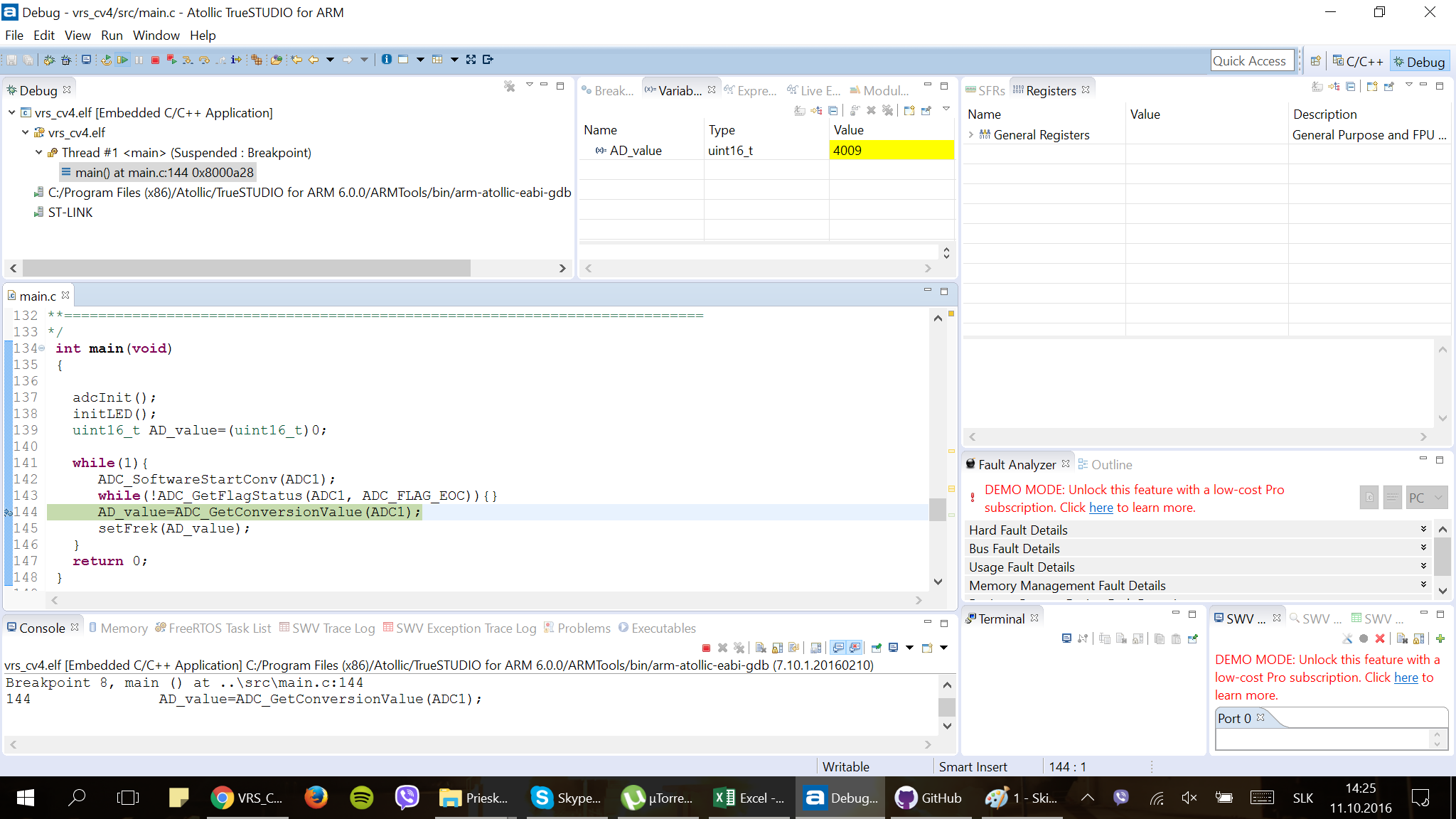
gpioInitStruc.GPIO\_Speed=*GPIO\_Speed\_400KHz*;

GPIO\_Init(GPIOA,&gpioInitStruc);

}

Hodnoty z potenciometra.





## Úloha 2. –

Nastavenie frekvencie blikania podľa hodnoty získaného z potenciometra.

**int** **main**(**void**)

{

adcInit();

initLED();

uint16\_t ADvalue=(uint16\_t)0;

**while**(1){

ADC\_SoftwareStartConv(ADC1);

**while**(!ADC\_GetFlagStatus(ADC1, ADC\_FLAG\_EOC)){}

ADvalue = ADC\_GetConversionValue(ADC1);

setFrek(ADvalue); // volanie funkcie kde sa nastavz frekvencia

}

**return** 0;

}

Funkcia nastavenia frekvencie podla hodnoty zo vstupu.

**void** **setFrek**(uint16\_t ADvalue){

**if**(ADvalue<2200){

GPIO\_SetBits(GPIOA, GPIO\_Pin\_5);

delay(50000);

GPIO\_ResetBits(GPIOA, GPIO\_Pin\_5);

delay(50000);

}

**else** **if**(ADvalue<3200){

GPIO\_SetBits(GPIOA, GPIO\_Pin\_5);

delay(150000);

GPIO\_ResetBits(GPIOA, GPIO\_Pin\_5);

delay(150000);

}

**else** **if**(ADvalue<3600){

GPIO\_SetBits(GPIOA, GPIO\_Pin\_5);

delay(300000);

GPIO\_ResetBits(GPIOA, GPIO\_Pin\_5);

delay(300000);

}

**else** **if**(ADvalue<3800){

GPIO\_SetBits(GPIOA, GPIO\_Pin\_5);

delay(500000);

GPIO\_ResetBits(GPIOA, GPIO\_Pin\_5);

delay(500000);

}

**else**{

GPIO\_SetBits(GPIOA, GPIO\_Pin\_5);

}

}