SLOVENKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE
FAKULTA ELEKTROTECHNIKY
A INFORMATIKY

Vnorene systémy

Cvičenie 4: AD prevodník

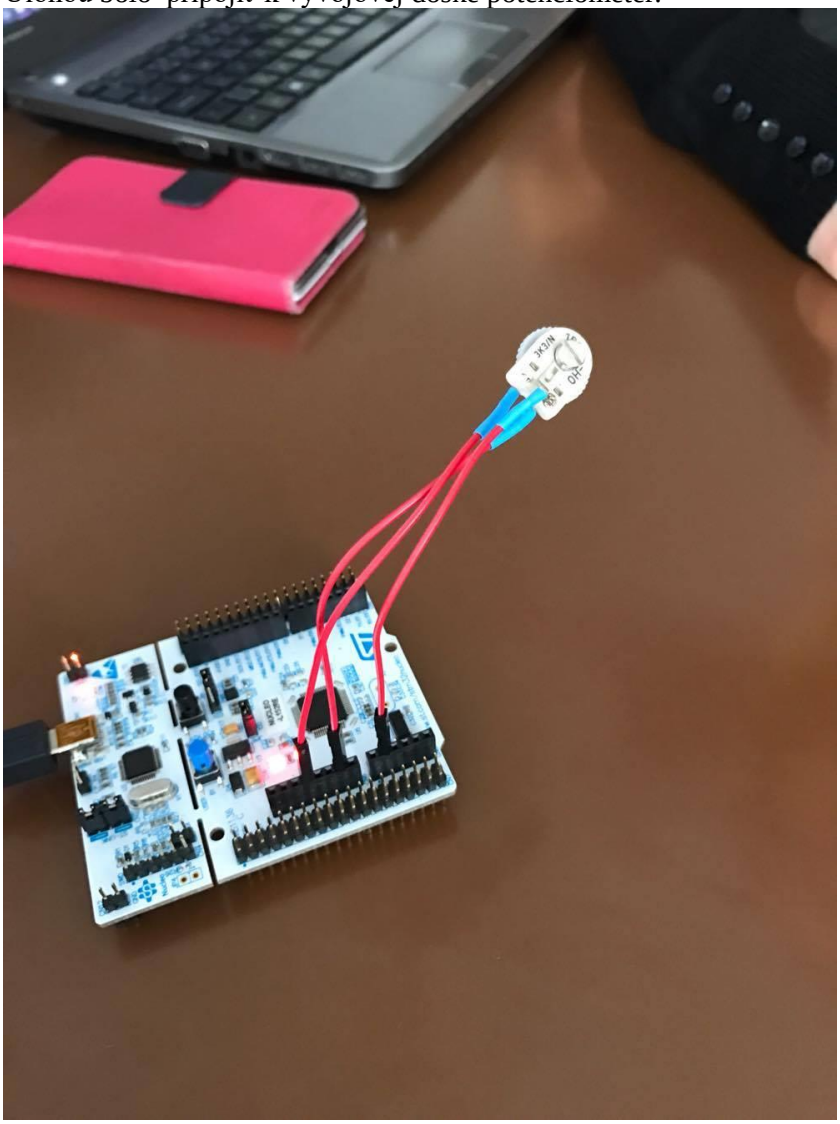
Cvičenie o 13.00
Zimný semester 16/17
1.ročník ing., Robotika

Bálint Domonkos

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.

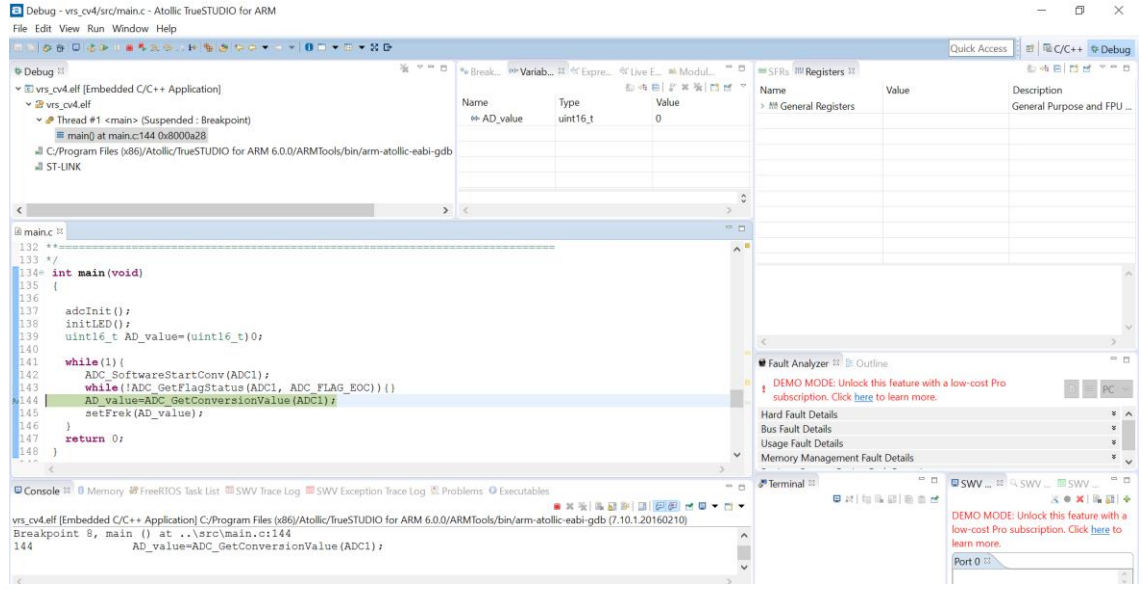
```
int main(void)
 {

   adcInit();
   initLED();
   uint16_t ADvalue=(uint16_t)0;
```

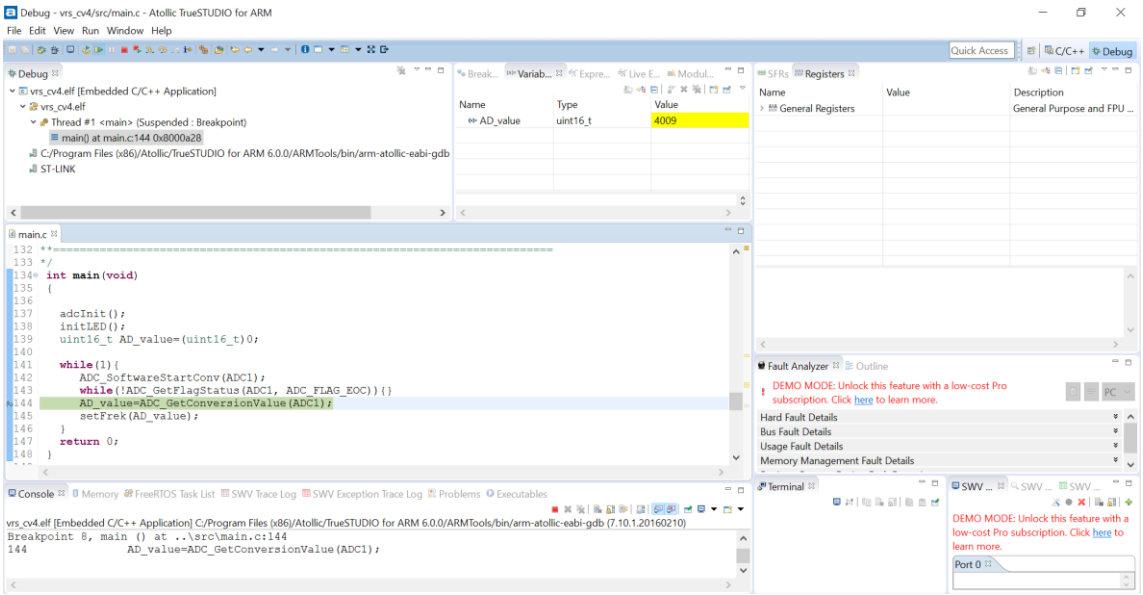
```
  while(1){
        ADC_SoftwareStartConv(ADC1);
        while(!ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC)){}
        ADvalue = ADC_GetConversionValue(ADC1);
  }
  return 0;
 }
```

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);
        }
}
```