SLOVENKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Vnorene systémy

Cvičenie 4: AD prevodník

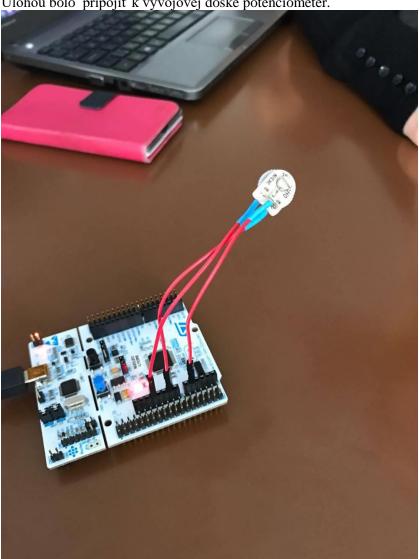
Cvičenie o 13.00 Zimný semester 16/17 1.ročník ing., Robotika

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.

 $ADC_Regular Channel Config (ADC1, ADC_Channel_1, 1, ADC_Sample Time_16 Cycles);$

Spustenie blikania a sledovanie hodnoty z potenciometra.

```
int\ main(void)
 adcInit();
 initLED();
 uint16_t ADvalue=(uint16_t)0;
```

```
\mathbf{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.
 3 Debug - vrs_cv4/src/main.c - Atollic TrueSTUDIO for ARM File Edit View Run Window Help
                                                                                                                                                                                                          Ø
                                                                                                                                                                                  Quick Access ■ C/C++ Debug
  © Debug II

✓ III vs.cvd.elf [Embedded C/C++ Application]

✓ IP vs.cvd.elf

✓ IP vs.cvd.elf

— Imaing at mainc144 0x8000a28

III C/Program Files (b86)/Atollic/InueSTUDIO for ARM 6.0.0/A

III ST-UNK
         int main (void)
            adcInit();
initLED();
uint16_t AD_value=(uint16_t)0;
                                                                                                                                               ■ Fault Analyzer 🕮 🕃 Outline
                                                                                                                                               DEMO MODE: Unlock this feati
subscription. Click <u>here</u> to learn
                                                                                                                                                                                                          D PC 4
  vn_cvd.elf [Embedded C/C++ Application] C:/Program Files (x86)/Atollic/InueSTUDIO for ARM 6.0.0/ARMTools/bi
Breakpoint 8, main () at ..\src\main.c:144
144 AD_value=ADC_GetConversionValue(ADCl);
                                                                                                                                                                                    Port 0 8
                                                                                                                                                                                                           ø
Debug - vrs_cv4/src/main.c - Atollic TrueSTUDIO for ARM

    Debug □
    □ vs_cv4.elf [Embedded C/C++ Application]
    □ vs_cv4.elf [Embedded C/C++ Application]
    □ vs_ts_cv4.elf
    □ Thread #1 < main > (Suspended : Breakpoint)
    □ main() at mainc:144 0:80000x28
    □ C/Program Files (x86)/Atollic/frueSTUDIO for ARM 6.0.0/ARMTools/bin/arm-atollic-eabl-gdb
                                                                                                                                                                                                   648 66
        int main (void)
           adcInit();
initLED();
uint16_t AD_value=(uint16_t)0;
   idd
while(1){
    while(1){
    ADC_SOftwareStartConv(ADCl);
    while(iADC_GetFlagStatus(ADCl, ADC_FLAG_EOC))()
    AD_value=ADC_GetConversionValue(ADCl);
    setFrek(AD_value);

    Fault Analyzer 
    □ Gutlin

                                                                                                                                               DEMO MODE: Unlock this feature with a low-cost Pro
                                                                                                                                               Hard Fault Details
                                                                                                                                                Usage Fault Details
Memory Management Fault Details
 vs_cvd.eff [Embedded C/C++ Application] C/Program Files (w86)/Atollic/TrueSTUDIO for ARM 6.0.0/ARM/Tools/bin/arm-atollis
Breakpoint 8, main () at .\src.\main.c:144

AD value=ADC_detConversionValue (ADC1);
                                                                                                                                                                                     Port 0 8
```

Úloha 2. –

Nastavenie frekvencie blikania podľa hodnoty získaného z potenciometra.

```
int main(void)
 adcInit();
 initLED();
 uint16_t ADvalue=(uint16_t)0;
 \mathbf{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);
        }
}
```