Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

**МЕХАНІКО-ТЕХНОЛОГІЧНИЙ ФАКУЛЬТЕТ**

Кафедра програмування та захисту інформації

**Звіт**

з виконаної лабораторної роботи № 5

дисципліни Програмування вбудованих систем”

на тему

«**Мікроконтролери ARM STM32F4xx, використання**

**цифро-аналогового перетворювача напруги — АЦП**

.»

Виконав :

студент академічної групи КІ-15

Аннаєв А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Перевірив :

Викладач

Савеленко О.К.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кропивницкий- 2017

ЛАБОЛАТОРНА РОБОТА №5

Тема:

**Мікроконтролери ARM STM32F4xx, використання цифро-аналогового перетворювача напруги — АЦП**

Мета:

1) Створити програмне забезпечення для мікроконтролеру, яке керуватиме світлодіодами на основі виміряної напруги.

2) Час світіння та вимкнення світлодіоду повинне регулюватися з різних каналів АЦП.

**Хід виконання:**

1. Створити новий проект для процесору STM32F4xx. Додати для проекту модулі RCC, GPIO, TIM, ADC.

2. Додати налаштування входів, які будуть використовуватися для вимірювання напруги:

void adc\_pin\_init()

{

GPIO\_InitTypeDef gpio;

RCC\_AHB1PeriphClockCmd(RCC\_AHB1Periph\_GPIOA, ENABLE);

GPIO\_StructInit(&gpio);

gpio.GPIO\_Mode = GPIO\_Mode\_AN;

gpio.GPIO\_Pin = GPIO\_Pin\_1 | GPIO\_Pin\_2;

GPIO\_Init(GPIOA, &gpio);

}

3. Налаштувати АЦП для вимірювання напруги по запиту до конкретного каналу:

void adc\_init()

{

ADC\_InitTypeDef ADC\_InitStructure;

ADC\_CommonInitTypeDef adc\_init;

RCC\_APB2PeriphClockCmd(RCC\_APB2Periph\_ADC1, ENABLE);

ADC\_DeInit();

ADC\_StructInit(&ADC\_InitStructure);

adc\_init.ADC\_Mode = ADC\_Mode\_Independent;

adc\_init.ADC\_Prescaler = ADC\_Prescaler\_Div2;

ADC\_InitStructure.ADC\_ScanConvMode = DISABLE;

ADC\_InitStructure.ADC\_ContinuousConvMode = DISABLE;

ADC\_InitStructure.ADC\_ExternalTrigConv = ADC\_ExternalTrigConvEdge\_None;

ADC\_InitStructure.ADC\_DataAlign = ADC\_DataAlign\_Right;

ADC\_InitStructure.ADC\_Resolution = ADC\_Resolution\_12b;

ADC\_CommonInit(&adc\_init);

ADC\_Init(ADC1, &ADC\_InitStructure);

ADC\_Cmd(ADC1, ENABLE);

}

4. Додати функцію безпосередньо для вимірювання напруги на вказаному каналі:

u16 readADC1(u8 channel)

{

ADC\_RegularChannelConfig(ADC1, channel, 1, ADC\_SampleTime\_3Cycles);

ADC\_SoftwareStartConv(ADC1);

while (ADC\_GetFlagStatus(ADC1, ADC\_FLAG\_EOC) == RESET);

return ADC\_GetConversionValue(ADC1);

}

5. Додати функцію для налаштування виходів для керування світлодіодами (див. попередні роботи).

6. Реалізуйте основний цикл програми:

While(1){

unsigned int bin\_code = readADC1(ADC\_Channel\_1);

GPIO\_SetBits(GPIOD,GPIO\_Pin\_12);

Delay(bin\_code\*1000); //Час світіння пропорційний виміряній напрузі

bin\_code = readADC1(ADC\_Channel\_2);

GPIO\_ResetBits(GPIOD,GPIO\_Pin\_12);//Час вимкнення пропорційний виміряній напрузі

Delay(bin\_code\*1000);

}