Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

**МЕХАНІКО-ТЕХНОЛОГІЧНИЙ ФАКУЛЬТЕТ**

Кафедра програмування та захисту інформації

**Звіт**

з виконаної лабораторної роботи № 2

дисципліни Програмування вбудованих систем”

на тему

«**Мікроконтролери ARM STM32F4xx, керування напругою**

**на виходах мікроконтролеру, вимірювання часу**»

Виконав :

студент академічної групи КІ-15

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

Перевірив :

Викладач

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

Кропивницкий- 2017

ЛАБОЛАТОРНА РОБОТА №2

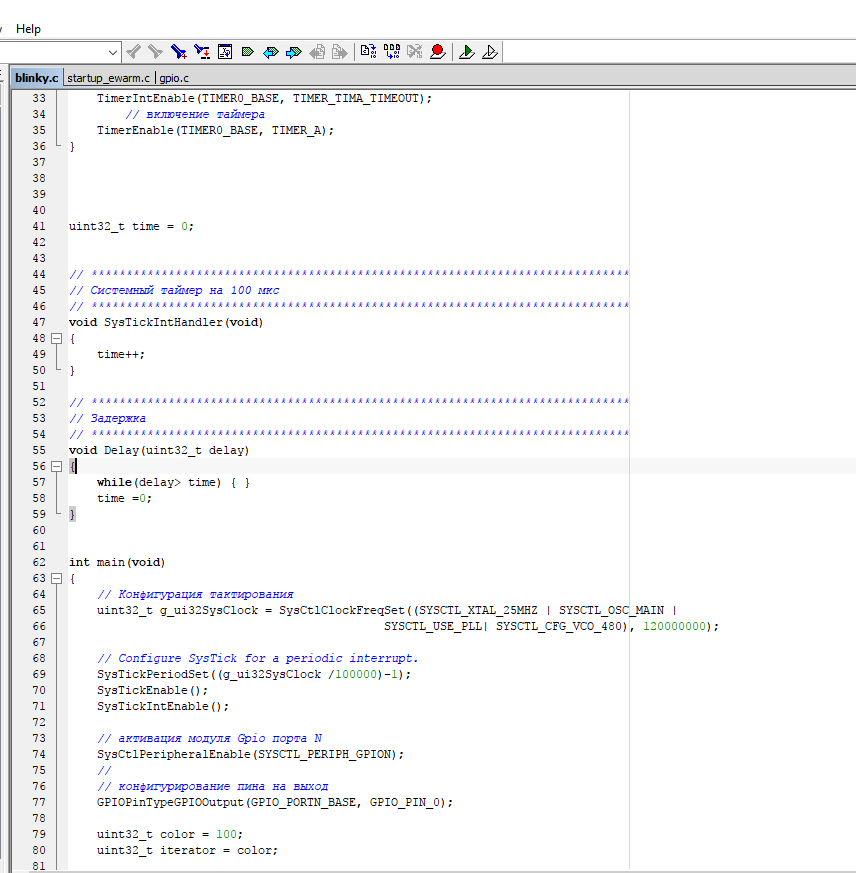
Тема: **Мікроконтролери ARM STM32F4xx, керування напругою на виходах мікроконтролеру, вимірювання часу**

Мета:

1) Створити програмне забезпечення для мікроконтролеру, яке керуватиме світлодіодами.

2) Організувати задані інтервали часу за допомогою таймерів.

3) Досягти плавну зміну яскравості світлодіоду.



uint32\_t time = 0;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Системный таймер на 100 мкс

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void SysTickIntHandler(void)

{

time++;

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Задержка

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void Delay(uint32\_t delay)

{

while(delay> time) { }

time =0;

}

int main(void)

{

// Конфигурация тактирования

uint32\_t g\_ui32SysClock = SysCtlClockFreqSet((SYSCTL\_XTAL\_25MHZ |

SYSCTL\_OSC\_MAIN |

SYSCTL\_USE\_PLL |

SYSCTL\_CFG\_VCO\_480), 120000000);

// Configure SysTick for a periodic interrupt.

SysTickPeriodSet((g\_ui32SysClock /100000)-1);

SysTickEnable();

SysTickIntEnable();

// активация модуля Gpio порта N

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPION);

//

// конфигурирование пина на выход

GPIOPinTypeGPIOOutput(GPIO\_PORTN\_BASE, GPIO\_PIN\_0);

uint32\_t color = 100;

uint32\_t iterator = color;

while(1)

{

GPIOPinWrite(GPIO\_PORTN\_BASE, GPIO\_PIN\_0, GPIO\_PIN\_0);

Delay(iterator);

GPIOPinWrite(GPIO\_PORTN\_BASE, GPIO\_PIN\_0, 0x0);

Delay(color- iterator);

if(iterator== 0)

iterator = color ;

else

iterator--;

}

}