**Додаток Б**

ТЕКСТ ФАЙЛУ MAIN.С

#include "stm32f10x\_gpio.h"

#include "wh0802.h"

#include "menu.h"

#define RCC\_APB1ENR\_INIT (RCC\_APB1ENR\_TIM2EN | RCC\_APB1ENR\_TIM3EN | RCC\_APB1ENR\_TIM6EN)

#define RCC\_APB2ENR\_INIT (RCC\_APB2ENR\_IOPAEN | RCC\_APB2ENR\_IOPBEN | RCC\_APB2ENR\_IOPCEN | RCC\_APB2ENR\_AFIOEN | RCC\_APB2ENR\_USART1EN)

#define GPIOA\_CRH\_INIT 0x44442222

#define GPIOB\_CRL\_INIT 0x4a444444

#define GPIOB\_CRH\_INIT 0x4444a244

#define GPIOC\_CRL\_INIT 0x44424444

#define GPIOC\_CRH\_INIT 0x444222a2

#define AFIO\_MAPR\_INIT (AFIO\_MAPR\_TIM2\_REMAP | AFIO\_MAPR\_TIM3\_REMAP | AFIO\_MAPR\_USART1\_REMAP)

#define USART1\_BRR\_INIT (0x9c4)

#define USART1\_CR1\_INIT (USART\_CR1\_UE | USART\_CR1\_TE | USART\_CR1\_RE | USART\_CR1\_TCIE | USART\_CR1\_RXNEIE)

#define EXTI\_IMR\_INIT 0x0000000f

#define EXTI\_RTSR\_INIT 0x0000000f

#define NVIC\_ISER0\_INIT (NVIC\_ISER\_SETENA\_6 | NVIC\_ISER\_SETENA\_7 | NVIC\_ISER\_SETENA\_8 | NVIC\_ISER\_SETENA\_9)

#define NVIC\_ISER1\_INIT (NVIC\_ISER\_SETENA\_5)

uint32\_t PrButton = 0;

void MOTOR\_Start() {

unsigned int MHSpeed;

unsigned int MLSpeed = 240;

if (!CONTROL.Direction) {

MHSpeed = CONTROL.Speed \* 24000 / 100;

TIM3->CCR4 = 23760;

}

else {

MHSpeed = CONTROL.Speed \* 24000 / 100;

TIM2->CCR4 = 240;

}

GPIOC->ODR |= (1<<4);

TIM2->CR1 |= TIM\_CR1\_CEN;

TIM3->CR1 |= TIM\_CR1\_CEN;

while (MHSpeed >= MLSpeed) {

if (!CONTROL.Direction) TIM2->CCR4 = MLSpeed;

else TIM3->CCR4 = 24000 - MLSpeed;

WH0802\_DelayMS(25);

MLSpeed += 240;

}

}

void MOTOR\_UpdateSpeed() {

if (!CONTROL.Direction) {

TIM2->CCR4 = CONTROL.Speed \* 24000 / 100;

TIM3->CCR4 = 23760;

}

else {

TIM3->CCR4 = (100 - CONTROL.Speed) \* 24000 / 100;

TIM2->CCR4 = 240;

}

}

void MOTOR\_Stop() {

unsigned int MHSpeed;

unsigned int MLSpeed = 240;

if (!CONTROL.Direction) {

MHSpeed = CONTROL.Speed \* 24000 / 100;

TIM3->CCR4 = 23760;

}

else {

MHSpeed = CONTROL.Speed \* 24000 / 100;

TIM2->CCR4 = 240;

}

while (MHSpeed >= MLSpeed) {

if (!CONTROL.Direction) TIM2->CCR4 = MHSpeed;

else TIM3->CCR4 = 24000 - MHSpeed;

WH0802\_DelayMS(25);

MHSpeed -= 240;

}

GPIOC->ODR &= ~(1<<4);

TIM2->CR1 &= ~TIM\_CR1\_CEN;

TIM3->CR1 &= ~TIM\_CR1\_CEN;

}

void MENU\_UpdateSpeed(unsigned int Speed) {

CONTROL.VALUE[0][0] = '>';

CONTROL.VALUE[0][4] = (Speed / 10) | 0x30;

CONTROL.VALUE[0][5] = (Speed % 10) | 0x30;

MOTOR\_UpdateSpeed();

}

void MENU\_UpdateDirection(unsigned int Direction) {

if (!Direction) {

CONTROL.VALUE[1][0] = '>'; CONTROL.VALUE[1][1] = 'f';

CONTROL.VALUE[1][2] = 'o'; CONTROL.VALUE[1][3] = 'r';

CONTROL.VALUE[1][4] = 'w'; CONTROL.VALUE[1][5] = 'a';

CONTROL.VALUE[1][6] = 'r'; CONTROL.VALUE[1][7] = 'd';

}

else {

CONTROL.VALUE[1][0] = '>'; CONTROL.VALUE[1][1] = 'r';

CONTROL.VALUE[1][2] = 'e'; CONTROL.VALUE[1][3] = 'v';

CONTROL.VALUE[1][4] = 'e'; CONTROL.VALUE[1][5] = 'r';

CONTROL.VALUE[1][6] = 's'; CONTROL.VALUE[1][7] = 'e';

}

}

void MENU\_Update(unsigned int Button) {

if (Button == 1) {

if (!CONTROL.ItemValue) {

if (CONTROL.CurItem) CONTROL.CurItem = 0;

else CONTROL.CurItem = 1;

}

else {

if (CONTROL.CurItem == 0) {

if (CONTROL.Speed > 5) CONTROL.Speed -= 5;

else CONTROL.Speed = 5;

MENU\_UpdateSpeed(CONTROL.Speed);

}

else {

if (CONTROL.Direction) CONTROL.Direction = 0;

else CONTROL.Direction = 1;

MENU\_UpdateDirection(CONTROL.Direction);

}

}

}

if (Button == 2) {

if (!CONTROL.ItemValue) {

CONTROL.ItemValue = 1;

CONTROL.ITEM[CONTROL.CurItem][0] = ' ';

CONTROL.VALUE[CONTROL.CurItem][0] = '>';

}

else {

CONTROL.ItemValue = 0;

CONTROL.ITEM[CONTROL.CurItem][0] = '>';

CONTROL.VALUE[CONTROL.CurItem][0] = ' ';

}

}

if (Button == 3) {

if (!CONTROL.ItemValue) {

if (!CONTROL.CurItem) CONTROL.CurItem = 1;

else CONTROL.CurItem = 0;

}

else {

if (CONTROL.CurItem == 0) {

if (CONTROL.Speed < 95) CONTROL.Speed += 5;

else CONTROL.Speed = 95;

MENU\_UpdateSpeed(CONTROL.Speed);

}

else {

if (CONTROL.TurnOn) MOTOR\_Stop();

if (CONTROL.Direction) CONTROL.Direction = 0;

else CONTROL.Direction = 1;

MENU\_UpdateDirection(CONTROL.Direction);

if (CONTROL.TurnOn) MOTOR\_Start();

}

}

}

if (Button == 4) {

if (CONTROL.TurnOn) {

CONTROL.TurnOn = 0;

MOTOR\_Stop();

}

else {

CONTROL.TurnOn = 1;

MOTOR\_Start();

}

}

if (Button == 5) {

EXTI->IMR &= ~EXTI\_IMR\_INIT;

if (!CONTROL.TurnOn) {

CONTROL.Direction = 0;

MENU\_UpdateDirection(CONTROL.Direction);

CONTROL.Speed = 5;

MENU\_UpdateSpeed(CONTROL.Speed);

CONTROL.TurnOn = 1;

MOTOR\_Start();

}

}

if (Button == 6) {

EXTI->IMR |= EXTI\_IMR\_INIT;

if (CONTROL.TurnOn) {

CONTROL.TurnOn = 0;

MOTOR\_Stop();

}

}

if (Button == 7) {

if (CONTROL.TurnOn) MOTOR\_Stop();

if (CONTROL.Direction) CONTROL.Direction = 0;

else CONTROL.Direction = 1;

MENU\_UpdateDirection(CONTROL.Direction);

if (CONTROL.TurnOn) MOTOR\_Start();

}

if (Button == 8) {

if ((CONTROL.Speed >= 5) && (CONTROL.Speed <= 95))

MENU\_UpdateSpeed(CONTROL.Speed);

}

}

void main(void) {

RCC->APB1ENR |= RCC\_APB1ENR\_INIT;

RCC->APB2ENR |= RCC\_APB2ENR\_INIT;

GPIOA->CRH = GPIOA\_CRH\_INIT;

GPIOB->CRL = GPIOB\_CRL\_INIT;

GPIOB->CRH = GPIOB\_CRH\_INIT;

GPIOC->CRL = GPIOC\_CRL\_INIT;

GPIOC->CRH = GPIOC\_CRH\_INIT;

AFIO->MAPR = AFIO\_MAPR\_INIT;

USART1->BRR = USART1\_BRR\_INIT;

USART1->CR1 = USART1\_CR1\_INIT;

EXTI->IMR = EXTI\_IMR\_INIT;

EXTI->RTSR = EXTI\_RTSR\_INIT;

NVIC->ISER[0] = NVIC\_ISER0\_INIT;

NVIC->ISER[1] = NVIC\_ISER1\_INIT;

GPIOC->ODR &= ~(1<<4);

GPIOC->ODR |= (1<<8);

GPIOB->ODR |= (1<<10);

TIM2->CR1 |= TIM\_CR1\_ARPE;

TIM2->CCMR2 |= (TIM\_CCMR2\_OC4PE | TIM\_CCMR2\_OC4M\_2 | TIM\_CCMR2\_OC4M\_1);

TIM2->ARR = 24000;

TIM2->CCR4 = 1200;

TIM2->CCER |= (TIM\_CCER\_CC4E);

TIM3->CR1 |= TIM\_CR1\_ARPE;

TIM3->CCMR2 |= (TIM\_CCMR2\_OC4PE | TIM\_CCMR2\_OC4M\_2 | TIM\_CCMR2\_OC4M\_1);

TIM3->ARR = 24000;

TIM3->CCR4 = 22800;

TIM3->CCER |= (TIM\_CCER\_CC4E | TIM\_CCER\_CC4P);

WH0802\_Init();

WH0802\_WriteString(CONTROL.ITEM[CONTROL.CurItem], 8, 1);

WH0802\_WriteString(CONTROL.VALUE[CONTROL.CurItem], 8, 2);

while (1) {

if (PrButton) {

MENU\_Update(PrButton);

WH0802\_WriteString(CONTROL.ITEM[CONTROL.CurItem], 8, 1);

WH0802\_WriteString(CONTROL.VALUE[CONTROL.CurItem], 8, 2);

PrButton = 0;

}

}

}

void EXTI0\_IRQHandler(void) {

EXTI->IMR &= ~EXTI\_IMR\_MR0;

WH0802\_DelayMS(20);

if ((GPIOA->IDR)&(GPIO\_Pin\_0)) PrButton = 1;

EXTI->PR |= EXTI\_PR\_PR1;

EXTI->IMR |=(EXTI\_IMR\_MR0);

}

void EXTI1\_IRQHandler(void) {

EXTI->IMR &= ~EXTI\_IMR\_MR1;

WH0802\_DelayMS(20);

if ((GPIOA->IDR)&(GPIO\_Pin\_1)) PrButton = 2;

EXTI->PR |= EXTI\_PR\_PR1;

EXTI->IMR |= (EXTI\_IMR\_MR1);

}

void EXTI2\_IRQHandler(void) {

EXTI->IMR&= ~EXTI\_IMR\_MR2;

WH0802\_DelayMS(20);

if ((GPIOA->IDR)&(GPIO\_Pin\_2)) PrButton = 3;

EXTI->PR|= EXTI\_PR\_PR2;

EXTI->IMR|=(EXTI\_IMR\_MR2);

}

void EXTI3\_IRQHandler(void) {

EXTI->IMR&= ~EXTI\_IMR\_MR3;

WH0802\_DelayMS(20);

if ((GPIOA->IDR)&(GPIO\_Pin\_3)) PrButton = 4;

EXTI->PR|= EXTI\_PR\_PR3;

EXTI->IMR|=(EXTI\_IMR\_MR3);

}

void USART1\_IRQHandler(void) {

unsigned int USARTData;

if (USART1->SR & USART\_SR\_RXNE) {

USART1->SR &= ~USART\_SR\_RXNE;

USARTData = USART1->DR;

if (USARTData == 0x20) PrButton = 5;

if (USARTData == 0x40) PrButton = 7;

if (USARTData == 0x60) PrButton = 6;

if ((USARTData & 0xe0) == 0x80) {

CONTROL.Speed = (USARTData & 0x1f) \* 5;

PrButton = 8;

}

}

if (USART1->SR & USART\_SR\_TC) {

USART1->SR &= ~USART\_SR\_TC;

}

}

**Додаток В**

ТЕКСТ ФАЙЛУ MENU.H

#ifndef \_MENU\_

#define \_MENU\_

struct \_control{

unsigned char ITEM [2][8];

unsigned char VALUE [2][8];

unsigned int CurItem;

unsigned int TurnOn;

unsigned int Speed;

unsigned int Direction;

unsigned int ItemValue;

};

struct \_control CONTROL = {">SPEED: ", ">MT\_DIR:", " dc=05% ", " forward", 0, 0, 5, 0, 0};

#endif //!\_MENU\_

**Додаток Г**

ТЕКСТ ФАЙЛУ WH0802.H

#ifndef \_WH0802\_

#define \_WH0802\_

typedef enum {false = 0, true = 1} bool;

void WH0802\_DelayMS(unsigned int Delay);

void WH0802\_DelayUS(unsigned int Delay);

void WH0802\_Init();

void WH0802\_SendDC(unsigned char Data, bool DC);

void WH0802\_WriteString(unsigned char\* String, unsigned int Length, unsigned int Row);

#endif //!\_WH0802\_

**Додаток Д**

ТЕКСТ ФАЙЛУ WH0802.C

#include "stm32f10x\_gpio.h"

#include "wh0802.h"

#define DATA\_PORT GPIOA->ODR

#define COM\_PORT GPIOC->ODR

#define DB4 (1<<8)

#define DB5 (1<<9)

#define DB6 (1<<10)

#define DB7 (1<<11)

#define RS (1<<10)

#define RW (1<<11)

#define E (1<<12)

#define F\_APB1 24000000

#define WH0802\_TIM TIM6

void WH0802\_DelayMS(unsigned int Delay) {

WH0802\_TIM->PSC = F\_APB1/1000+1;

WH0802\_TIM->ARR = Delay;

WH0802\_TIM->EGR |= TIM\_EGR\_UG;

WH0802\_TIM->CR1 |= TIM\_CR1\_CEN|TIM\_CR1\_OPM;

while (WH0802\_TIM->CR1&TIM\_CR1\_CEN!=0);

}

void WH0802\_DelayUS(unsigned int Delay) {

WH0802\_TIM->PSC = F\_APB1/1000000+1;

WH0802\_TIM->ARR = Delay;

WH0802\_TIM->EGR |= TIM\_EGR\_UG;

WH0802\_TIM->CR1 |= TIM\_CR1\_CEN|TIM\_CR1\_OPM;

while (WH0802\_TIM->CR1&TIM\_CR1\_CEN!=0);

}

void WH0802\_SendDC(unsigned char Data, bool DC) {

if (DC) COM\_PORT |= RS;

else COM\_PORT &= ~ RS;

DATA\_PORT &= ~ (DB4 | DB5 | DB6 | DB7);

if (Data & (1<<7)) DATA\_PORT |= DB7;

if (Data & (1<<6)) DATA\_PORT |= DB6;

if (Data & (1<<5)) DATA\_PORT |= DB5;

if (Data & (1<<4)) DATA\_PORT |= DB4;

COM\_PORT |= E;

WH0802\_DelayUS(2);

COM\_PORT &= ~E;

WH0802\_DelayUS(10);

DATA\_PORT &= ~ (DB4 | DB5 | DB6 | DB7);

if (Data & (1<<3)) DATA\_PORT |= DB7;

if (Data & (1<<2)) DATA\_PORT |= DB6;

if (Data & (1<<1)) DATA\_PORT |= DB5;

if (Data & (1<<0)) DATA\_PORT |= DB4;

COM\_PORT |= E;

WH0802\_DelayUS(2);

COM\_PORT &= ~E;

WH0802\_DelayUS(40);

}

void WH0802\_WriteString(unsigned char\* String, unsigned int Length, unsigned int Row) {

unsigned int i;

if ((Row == 1) || (Row == 2)) {

if (Row == 1) WH0802\_SendDC(0x80, 0);

if (Row == 2) WH0802\_SendDC(0xC0, 0);

for (i = 0; i < Length; ++i) WH0802\_SendDC(String[i], 1);

}

}

void WH0802\_Init() {

COM\_PORT &= ~RW;

WH0802\_DelayMS(50);

WH0802\_SendDC(0x30, 0);//0b001/DL/0000, DL = 1

WH0802\_DelayUS(50);

WH0802\_SendDC(0x28, 0);//0b0010/N/F/00, N = 1, F = 0

WH0802\_DelayUS(50);

WH0802\_SendDC(0x28, 0);//0b0010/N/F/00, N = 1, F = 0

WH0802\_DelayUS(50);

WH0802\_SendDC(0x0c, 0);//0b0000/1/D/C/B, D = 1, C = 1, B = 0

WH0802\_DelayUS(50);

WH0802\_SendDC(0x01, 0);//clear display

WH0802\_DelayMS(2);//delay 2 ms

WH0802\_SendDC(0x06, 0);//0x0000 01 I/D SH, I/D = 1, SH = 0

}