MİKROİŞLEMCİLER UYGULAMA VİZE SINAVI

KARANLIK SENSÖRÜ ÇALIŞMA RAPORU

EMİNE VARGÜN-2112903020

24.11.2023

GÖLHİSAR UYGULAMALI BİLİMLER YÜKSEKOKULU
BİLİŞİM SİSTEMLERİ VE UYGULAMALARI

Amaç:

STM32 mikrodenetleyici ve bir karanlık sensörü kullanarak ortam ışığını ölçmek ve bu değeri bir OLED ekran üzerinde görüntülemek.

Malzemeler:

STM32 mikrodenetleyici kartı

Karanlık sensörü (LDR)

OLED ekran

Breadboard ve jumper kabloları

10k ohm direnç

USB kablosu (STM32 kartını bilgisayara bağlamak için)

Devre Bağlantıları:

LDR'yi bir ucu GND'ye, diğer ucu 3.3V'e bağlayın.

LDR'nin ortasındaki bağlantıyı bir ucu A0 pini, diğer ucu 10k ohm direncin diğer ucuna bağlanacak şekilde bağlayın. 10k ohm direncin diğer ucu 3.3V'e bağlı olmalıdır.

OLED ekranını I2C bağlantılarıyla bağlayın (SDA ve SCL pinleri).

STM32 kartını bilgisayara bağlayın.

Yazılım:

STM32CubeIDE veya STM32CubeMX kullanarak proje oluşturun.

Projeye ssd1306.h ve fonts.h dosyalarını ekleyin. Bu dosyalar, OLED ekranını kontrol etmek için kullanılacaktır.

main.c dosyasındaki ilgili kodları aşağıdaki gibi güncelleyin:

```c
/* USER CODE BEGIN Header */
/**
  ******************************************************************************
  * @file           : main.c
  * @brief          : Main program body
  ******************************************************************************
  * @attention
  *
  * Copyright (c) 2023 STMicroelectronics.
  * All rights reserved.
  *
  * This software is licensed under terms that can be found in the LICENSE file
  * in the root directory of this software component.
  * If no LICENSE file comes with this software, it is provided AS-IS.
  *
  ******************************************************************************
  */
/* USER CODE END Header */
/* Includes ------------------------------------------------------------------*/
#include "main.h"

/* Private includes ----------------------------------------------------------*/
/* USER CODE BEGIN Includes */
#include "fonts.h"
#include "ssd1306.h"


/* USER CODE END Includes */

/* Private typedef -----------------------------------------------------------*/
/* USER CODE BEGIN PTD */
```

```c
/* Private variables ---------------------------------------------------------*/
ADC_HandleTypeDef hadc1;

I2C_HandleTypeDef hi2c1;

/* USER CODE BEGIN PV */

/* USER CODE END PV */

/* Private function prototypes -----------------------------------------------*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_I2C1_Init(void);
static void MX_ADC1_Init(void);
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code ---------------------------------------------------------*/
/* USER CODE BEGIN 0 */
uint32_t ADC_deger;
uint32_t karanlik;
char buffer[16];

void ADC_Read()
{
    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, 10000);
    ADC_deger=HAL_ADC_GetValue(&hadc1);
    HAL_ADC_Stop(&hadc1);

}
```

```
}

/* USER CODE END 0 */

/**
  * @brief  The application entry point.
  * @retval int
  */
int main(void)
{
  /* USER CODE BEGIN 1 */

  /* USER CODE END 1 */

  /* MCU Configuration--------------------------------------------------------*/

  /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
  HAL_Init();

  /* USER CODE BEGIN Init */

  /* USER CODE END Init */

  /* Configure the system clock */
  SystemClock_Config();

  /* USER CODE BEGIN SysInit */

  /* USER CODE END SysInit */

  /* Initialize all configured peripherals */
```

```
/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_I2C1_Init();
MX_ADC1_Init();
/* USER CODE BEGIN 2 */
SSD1306_Init();
SSD1306_Fill(0);

/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{

        ADC_Read();
        HAL_Delay(10);
        karanlik = (ADC_deger / 2026.00) * 100;
        sprintf(buffer, "karanlik=%lu", karanlik);  // %lu kullanılmalıdır, çünkü karanlik bir unsigned long değiskeni
        SSD1306_GotoXY(0, 25);
        SSD1306_Puts(buffer, &Font_11x18, 1);
        SSD1306_UpdateScreen();




  /* USER CODE END WHILE */

  /* USER CODE BEGIN 3 */
}
```

```c
  * @brief I2C1 Initialization Function
  * @param None
  * @retval None
  */
static void MX_I2C1_Init(void)
{

  /* USER CODE BEGIN I2C1_Init 0 */

  /* USER CODE END I2C1_Init 0 */

  /* USER CODE BEGIN I2C1_Init 1 */

  /* USER CODE END I2C1_Init 1 */
  hi2c1.Instance = I2C1;
  hi2c1.Init.ClockSpeed = 400000;
  hi2c1.Init.DutyCycle = I2C_DUTYCYCLE_2;
  hi2c1.Init.OwnAddress1 = 0;
  hi2c1.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
  hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
  hi2c1.Init.OwnAddress2 = 0;
  hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
  hi2c1.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
  if (HAL_I2C_Init(&hi2c1) != HAL_OK)
  {
    Error_Handler();
  }
  /* USER CODE BEGIN I2C1_Init 2 */

  /* USER CODE END I2C1_Init 2 */
```

```c
/**
  * @brief System Clock Configuration
  * @retval None
  */
void SystemClock_Config(void)
{
  RCC_OscInitTypeDef RCC_OscInitStruct = {0};
  RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
  RCC_PeriphCLKInitTypeDef PeriphClkInit = {0};

  /** Initializes the RCC Oscillators according to the specified parameters
  * in the RCC_OscInitTypeDef structure.
  */
  RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;
  RCC_OscInitStruct.HSEState = RCC_HSE_ON;
  RCC_OscInitStruct.HSEPredivValue = RCC_HSE_PREDIV_DIV1;
  RCC_OscInitStruct.HSIState = RCC_HSI_ON;
  RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
  RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
  RCC_OscInitStruct.PLL.PLLMUL = RCC_PLL_MUL9;
  if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
  {
    Error_Handler();
  }

  /** Initializes the CPU, AHB and APB buses clocks
  */
  RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                              |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
  RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
  RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
```

```c
/**
  * @brief GPIO Initialization Function
  * @param None
  * @retval None
  */
static void MX_GPIO_Init(void)
{
/* USER CODE BEGIN MX_GPIO_Init_1 */
/* USER CODE END MX_GPIO_Init_1 */

  /* GPIO Ports Clock Enable */
  __HAL_RCC_GPIOD_CLK_ENABLE();
  __HAL_RCC_GPIOA_CLK_ENABLE();
  __HAL_RCC_GPIOB_CLK_ENABLE();

/* USER CODE BEGIN MX_GPIO_Init_2 */
/* USER CODE END MX_GPIO_Init_2 */
}

/* USER CODE BEGIN 4 */

/* USER CODE END 4 */

/**
  * @brief  This function is executed in case of error occurrence.
  * @retval None
  */
void Error_Handler(void)
{
  /* USER CODE BEGIN Error_Handler_Debug */
  /* User can add his own implementation to report the HAL error return state */
  __disable_irq();
```

```c
/**
  * @brief  This function is executed in case of error occurrence.
  * @retval None
  */
void Error_Handler(void)
{
  /* USER CODE BEGIN Error_Handler_Debug */
  /* User can add his own implementation to report the HAL error return state */
  __disable_irq();
  while (1)
  {
  }
  /* USER CODE END Error_Handler_Debug */
}

#ifdef  USE_FULL_ASSERT
/**
  * @brief  Reports the name of the source file and the source line number
  *         where the assert_param error has occurred.
  * @param  file: pointer to the source file name
  * @param  line: assert_param error line source number
  * @retval None
  */
void assert_failed(uint8_t *file, uint32_t line)
{
  /* USER CODE BEGIN 6 */
  /* User can add his own implementation to report the file name and line number,
     ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
  /* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */
```

Açıklamalar:

fonts.h ve ssd1306.h dosyalarını eklemek, OLED ekranını kontrol etmek için gerekli kütüphaneleri sağlar.

ADC_Read fonksiyonu, ADC okuma işlemlerini gerçekleştirir.

Ana döngü içinde karanlık sensör değeri ölçülür ve OLED ekranda görüntülenir.

Notlar:

Devre bağlantılarını doğru yapmak önemlidir.

STM32CubeIDE veya STM32CubeMX kullanarak proje oluşturun ve gerekli kütüphaneleri ekleyin.

Yazılımı STM32 kartına yükleyerek deneyi başlatın.

Ekran üzerinde karanlık sensör değerinin doğru bir şekilde görüntülendiğinden emin olun.

KAYNAKÇA

- Ekip Arkadaşları: Burhan Üstübi, Ebubekir Kartal, Furkan Aslan(Bu kişilerden yardım aldım.)
- Chat CPT