**ASSIGNMENT - 1**

**Q.1**

1. Write Hello World at address location 0x20000100 in your STM32

controller

1. Store this address into pointer

2. Dereference the pointer to store characters - &#39; H&#39 ;

3. Increment pointer

4. Dereference the pointer to store characters - &#39; e&#39 ;

5. Repeat

Solution:

main.c file

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 \*/

int i=0;  
char \*ptr;  
char str[] = "Hello World";

/\* USER CODE END Init \*/

/\* Configure the system clock \*/  
SystemClock\_Config();

/\* USER CODE BEGIN SysInit \*/

/\* USER CODE END SysInit \*/

/\* Initialize all configured peripherals \*/  
MX\_GPIO\_Init();  
MX\_DFSDM1\_Init();  
MX\_I2C2\_Init();  
MX\_QUADSPI\_Init();  
MX\_SPI3\_Init();  
MX\_USART1\_UART\_Init();  
MX\_USART3\_UART\_Init();  
MX\_USB\_OTG\_FS\_PCD\_Init();

/\* USER CODE BEGIN 2 \*/  
ptr = (char \*)0x20000100;

/\* USER CODE END 2 \*/

/\* Infinite loop \*/  
 /\* USER CODE BEGIN WHILE \*/  
 while (str[i] != 0)  
 {  
 /\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/  
 \*ptr = str[i];  
 ptr++;  
 i++;  
 }  
 /\* USER CODE END 3 \*/  
 }

Output:

Graphical user interface, application

Description automatically generated

**Q.2** Modify the program to turn ON and OFF continuously in a loop with human observable software delay.

Answer:

main.c

**void** delay\_sec(**unsigned** **int** seconds)    // delay in terms of seconds

{

**unsigned** **int** i,j;

**for**(i=0; i<seconds; i++)

    {

**for**(j=0;j<8000;j++)

        {

        }

    }

}

/\* 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 \*/

  MX\_GPIO\_Init();

  MX\_DFSDM1\_Init();

  MX\_I2C2\_Init();

  MX\_QUADSPI\_Init();

  MX\_SPI3\_Init();

  MX\_USART1\_UART\_Init();

  MX\_USART3\_UART\_Init();

  MX\_USB\_OTG\_FS\_PCD\_Init();

  /\* USER CODE BEGIN 2 \*/

  /\* USER CODE END 2 \*/

    /\* USER CODE BEGIN 3 \*/

  uint32\_t \*pClkctrlreg = (uint32\_t\*)0x4002104C;

uint32\_t \*pPortDModeReg = (uint32\_t\*)0x48000000;

uint32\_t \*pPortDOutReg = (uint32\_t\*)0x48000014;

//1. Enable clock control register GPIOA enable (RCC\_AHB2ENR)

\*pClkctrlreg |= 0x01;

//2.Configure the mode of the IO pin as output

//a. Clear 10th and 11th bit position

\*pPortDModeReg &= 0xFFFFF3FF;

//b. make 10th & 11th bit position as 01 for output(SET)

\*pPortDModeReg |= 0x00000400;

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

**while**(1)

{

/\* USER CODE END WHILE \*/

//3. GPIOA output data register (used to write)

\*pPortDOutReg |= 0x0020;

//HAL\_Delay(2000);

delay\_sec(200);

**printf**("Led on1 \n");

// GPIOA output data register (clear the 5th bit)

\*pPortDOutReg &= 0x00000000;

//HAL\_Delay(2000);

delay\_sec(200);

**printf**("Led off1 \n");

\*pPortDOutReg |= 0x0020;

//HAL\_Delay(2000);

delay\_sec(200);

**printf**("Led on2 \n");

\*pPortDOutReg &= 0x00000000;

//HAL\_Delay(2000);

delay\_sec(200);

**printf**("Led off2 \n");

}

  /\* USER CODE END 3 \*/

}