Microcontroller Lab 02

Submitted by:

Kazi Shadman Sakib
FH-97
3rd Year
Computer Science and Engineering, University of Dhaka

USART communication and Message format

In the initial stage we used USART communication to set the initial configuration. The format goes like ”Config Init temp temperature water water level humidity humidity lights rooms in which light will be turned on.” The command must end with a dot. Next during the runtime of the controller. Using interrupt handlers two types of USART communication will be handled. One is Config another is read. Config will set the configuration of the variables to a given value. For example ”Config light 1101.” will disable the light of 3rd room.

Read command will send back the configuration. For example: ”Read light.” will send the configuration of the rooms.

GPIO and USART2 pin table

Input

Temperature/Humidity	PC0 and PC1	PC0 and PC1 is used to show the state of temperature or humidity. 00:normal 01:low 11:high
Lights	PB0,PA0,PA1 and PA4	Each bit represents the input of 4 rooms(living room, dining room, bedroom and office respectively.
Water Pump	PA3	Water pump is handled by USART communication. By sending a new configuration of Water pump the new water level will be set

Output

Temperature/Humidity	PA5 and PA6	PA5 is represented as AC/Dehumidifier and PA6 as Heater/Humidifier.Based on the input the output LED will be lit. And they will work for 30 second(assuming it will take 30 second to return them to normal)
Lights	PA7,PB6,PC7 and PA9	Each bit represents the input of 4 rooms(living room, dining room, bedroom and office respectively. According to the configuration and input the respective lights
Water Pump	PA5	If by USART communication the water level becomes below 45 the the pump will be on,ie, the LED will be lit and for 10 seconds until it reaches 55 litre and stops

USART2

Config and Read	PA2 and PA3	Using PA2 and PA3 we used USART communication to configure and read the system

List of register configures

RCC_CR: we write HSE ON to 1 and PLL ON to 1.

RCC_APB1ENR: We power on enable clock. We also enable USART2 and GPIOA,GPIOB and GPIOC enable clock.

PWR_CR: We power on enable clock

RCC_CFGR: We set the prescaler values to get the system clock speed.

RCC_PLLCFGR: We configure the main PLL.

GPIOx_MODER: We configure the MODER register with GPIO and alternate functions according to our need in the above table.

GPIOA_AFRL: We configure the alternate function for PA2 and PA3 to USART2. The value is AF7 for USART2. So we put the value 0111 on (11:10:9:8) and (15:14:13:12).

USART2_CR1: We use this to enable USART2 on 13th bit. And define the word length 8 bit by setting 0 on 12th bit. We enable transmitter and receiver on 2nd and 3rd bit by setting them to 1. We enable Receive interrupt by setting 1 on 5th bit.

USART2_SR: We use this in the runtine to know the status of the registers. If Receiver interrupt is enabled. The 5th bit will be 1 and that's our queue to start receiving the data.

USART2_DR: We use the data register to get the byte data one at a time. We also use this to send the data using USART communication.

USART2_BRR: We use this to set the baud rate to 115200.

GPIOx_BSSR: We use this to set and reset the output of led we are using to represent the engines.

GPIOx_IDR: We use this to read the current input of the GPIO pins.

Check list

DONE:

- USART communication has been done using the terminal and PA2 and PA3.
- Temperature and humidity sensor turning on respective engines based on GPIO.
- Lights turning on and off based on USART communication and GPIO pins.
- Water pump turning on and off based on USART communication and interrupt handler to know the water level.

NOT DONE:

- Running all the sensors at once.
- Temperature and humidity sensor turning on respective engines based on USART communication.
- Water pump turning on and off based on GPIO pins to know the water level.