Microcontroller Lab Assignment 2: Develop a building Controller using STM32 USART and GPIO for Building/Apartment Light, Temperature, Water Supply, and Humidity Control.

Computer Science and Engineering, University of Dhaka,

Version 1.1

Any inconsistency – please report

February 24, 2022

1 Objective

The objectives of the lab assignment are to understand and have hands-on training to build confidence and working knowledge in the use of USART (`https://www.youtube.com/watch?v=JuvWbRhhpdI&t=49s`, I put a random video: find other videos/Doc) and GPIO port to communicate with the Microcontroller to develop firmware for the practical control system.

2 Description

Use USART2 to set up the initial configuration, change a specific control configuration and get the current setup of the building control system. You can use TeraTerm, Hercules, Putty, or any other serial terminal program to communicate with Microcontroller (STM32F446re). Let the initial operating temperature be 25^oC, humidity 45%, and the minimum water level is 50 liter. The controller switches on the light bulbs when the luminance is less than 500 lumens in a room. However, switching on/off a light bulb will be dependent on the current setting. A user may keep living light control enabled until 10:00 pm. This arrangement implies that the owner can configure the light bulb for any devices/equipment to be sensitive to the control system or not. The controller switch-on the water pump when the water level in the upper tank is less than 45 liters and off when the level is 55 liter. These are the initial configuration of the controller. See fig. 1.

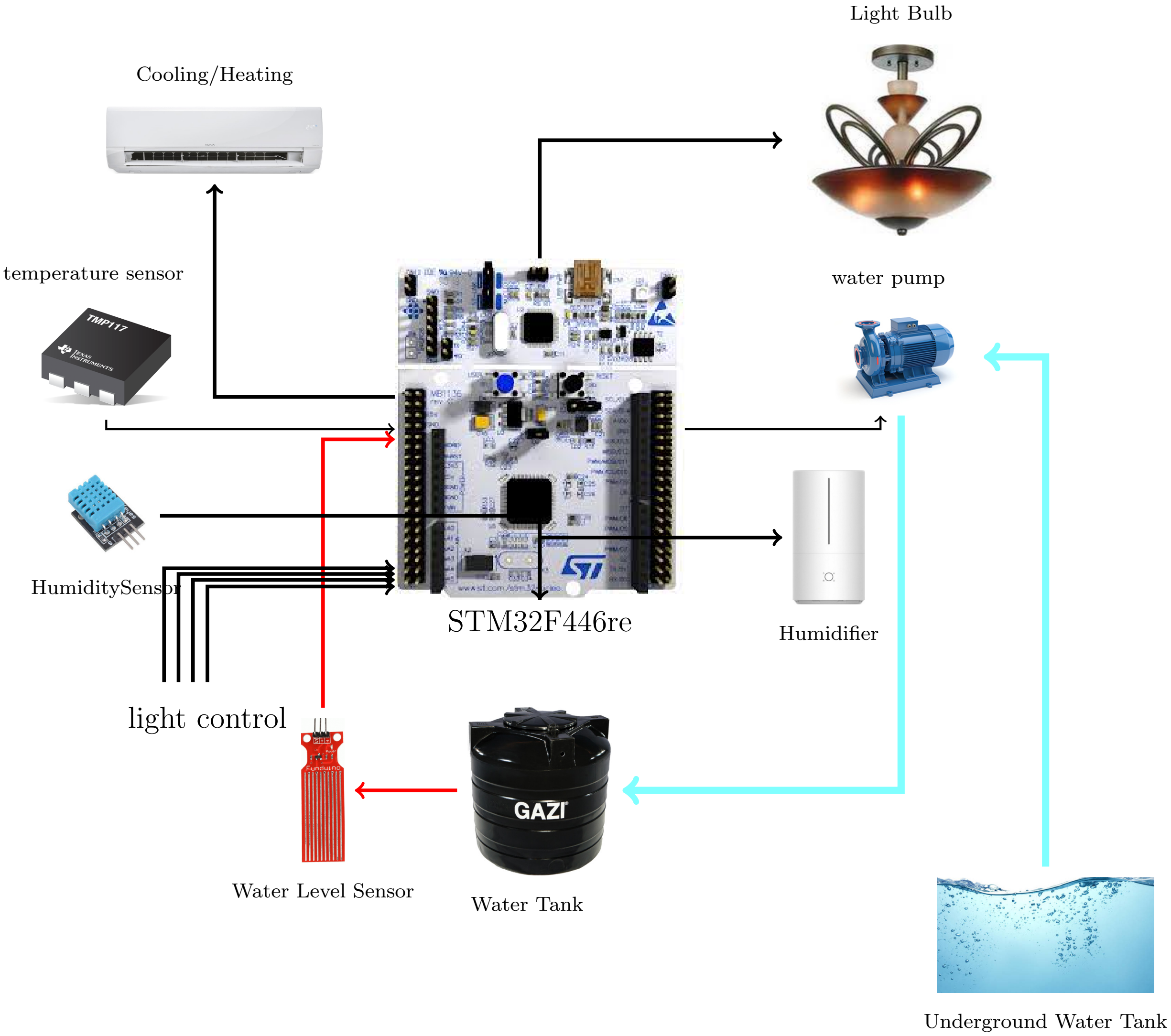

Figure 1: Building Control System

2.1 Lab Workable Interpretation of the Above Diagram

Use USART2 with the send and receive interrupt to set the initial configuration, change any specific configuration and get the configuration of the building control system. You can use TeraTerm, Hercules, Putty, or any other serial terminal to communicate with Microcontroller (STM32F446re).

In the real-world building control system, you may need electronic switches which will receive control to on or off from the controller and respond accordingly. For the current lab, you can use switches or jumper cables connected to VCC to indicate the input parameters to process and send a control command to the devices. See the detailed description as follows:

2.1.1 Temperature Control

The controller can obtain the actual temperature value by employing a temperature sensor. However, for our lab, we will use two jumper wires or toggle switches where 00 (both grounded/off) indicate the temperature is within the range of 23^oC C to 27^oC C. 10 or 01 (one jumper cable connected to VCC or switch-on) indicates the temperature is below 23^{o}C and 11 indicates the temperature is above 27^{o}C. In this case, the operation or expected temperature is 25^oC, and the tolerance level is ± 2^oC. The tolerance level defines that if the operating temperature is T^oC, then heating should be switch-on when the temperature is less than $T^oC - 2$ and the cooling system switch-on when the temperature is above $T^oC + 2$. We can replace the air conditioner or heating system by connecting two LEDs (two GPIO ports). We assume that the cooling/heating system can adjust the temperature within 30 seconds, and therefore, any of the two LEDs will switch on for 30 seconds. In the existing (real) system, the input temperature of the controller will change, and the controller will switch on/off the heating and cooling system. Also, the controller output will trigger an electronic switch on/off instead of directly controlling the A/C switch.

2.1.2 Humidity/Water Level Control

Similar to the temperature control, we can take two inputs to indicate current humidity below 40% and higher than 50%. The controller will switch off the pump for the water level above 55 liters and switch on the pump if the water level is below 45 liters (here, a LED).

2.1.3 Room Light Control

Let us have three inputs (Jumper cables or switches), one of the inputs indicates the night or daytime. At night the input is 1, and day time it is off (0). The other two inputs control the light bulb in the four rooms (Living (00), Dining (01), Bedroom (10), and office room (11))—the light bulb-switch on/off based on the input. However, the controller (terminal program) must send a light bulb configuration to set on for the control. All light bulbs are on at nighttime, depending on the configuration. The terminal program must send the configuration using the USART2. Now, let the light-bulbs configuration be '0001', then the office room light reacts to the controller's action, and all other three rooms' light-bulbs remain non-responsive to the controller actions whether it is day or night.

2.2 Control Command – Design Need

The computer terminal program must send the initial configuration to the MCU, and the MCU controller program will store it to a variable known as 'config.' You must use the USART interrupt in the MCU to send or receive any configuration. (See the interrupt handler in the USART lecture material uploaded microprocessor and micro-controller theory course website – google classroom)

```
Config Init temp 25 water 50 hum 45 light 1111
```

The above initial configuration command instruct controller to set and store the initial configuration unless otherwise changed. To change the configuration to the light bulbs or any other devices, the command sample is as follows: Initial configuration must be sent to the MCU from your computer and will be stored to a variable known as 'config'

```
 Config temp 27
```

The temperature set to 27^{o}C. Similarly, for light blub

```
Config light 1010
```

Living and Office rooms lights will respond for the day and night. To read the current configuration use the following command from the terminal program:

```
Read Config
```

For all configuration. To read the current configuration of the light bulb use the following command

```
Read Config light
```

3 To Do

First Select the GPIO port and pins for the input and output (Data sheet). Then use reference manual and data sheet to (Follow the Uploaded Videos `https://www.youtube.com/playlist?list=PLfIJKC1ud8ghc4eFhI84z_3p3Ap2MCMV-`)

- Configure The clock of MCU (Using External Crystal)
  a. Enable HSE (High Speed External Clock) and wait for HSE to become ready
  b. HSE (i) External Crystal/ceramic resonator (ii) HSE external user clock
  c. Set the power enable clock and volatge regulator
  d. Configure the FLASH PREFETCH and the LATENCY Related Settings
  e. Configure the PRESCALERS HCLK, PCLK1, PCLK2
  f. Configugure the main PLL
  g. Enable PLL and wait for it to become ready
  h. Select clock source and wait for it to be set
- Configure a timer
  a. Enable timer clock (RCC→ APBxENR)
  b. Set the prescaler and ARR (TIMx→PSC, TIMx→ARR)
  c. Enable the Timer and wait for update flag to ready (TIMx→CR1)
- Configure USART2 and GPIO port (for input and output)
  a. Enable UART clock and GPIO clock
  b. Configure UART pin for Alternate function
  c. Enable UART on USART˙CR1 rgister
  d. Program M bit in USART CR1 to define the word length
  e. Select the baud rate using the USART˙BRR register.
  f. Enable transmission TE and recieption bits in USART˙CR1 register

4 Report Needed

The report must contains

- USART communication and message format
- Tables for GPIO pins for input/output and USART2 and comments on the use.
- List of registers configure for the lab and their values
- Experiments check list (do or not done) with comments.

5 End Comments

For the real-life control system design for building management, you can try it with the actual sensor and device; however, you must follow the standard safety requirement.