Embedded Systems - Arduino UNO demonstration

Kaja Słomska , Richard Chomjak

March 2018

1 Project description

- Counter (Time) must be displayed on LED Display in HHMMSS format, where HH stands for hours MM, stands for minutes and SS stands for seconds.
- Mode for time changing can be activated with potentiometer. You can increment or decrement parts in the counter. This mode behaves the following way:
  - If there is a change on potentiometer, you can set the counter.
  - If you can set the counter, first you can change the hours part, after five seconds of inactivity (hours had not been set to different value) you can change the minutes part, after the five second of inactivity, counter works normally.
- Data about measurements of temperature, time should be saved into memory, and after interruption their have to be sent via serial interface.

2 Hardware

Arduino UNO and AVT1615 are used. Arduino UNO is used as base platform and AVT1615 is used as LED display and sensors platform.

2.1 Platforms connections

The AVT1615 is full compatible with Arduino UNO, connections are made by user intelligence - AREF on AREF, 0 on 0, GND on GND...

2.2 Sensors connections

AVT1615 provides four switches (Sn), one potentiometer and a voltage sensor (ADC), few LEDs, LED display, contrast controller for LED, and piezo sensor.

There is mapping between AVT1615 platform to Arduino UNO:

S1 $->$ PD3 , Sends measured data trough serial interface.
S2 $->$ PD2 , Increments counter by one.
S3 $->$ PD1 , Decrements counter by one.
PR2 $->$ PCO , Voltage on PR2.
U1 $->$ PC1 , Temperature sensor.

3 Software

3.1 Time counter

Changing counter states is provided by automata (TIME_STATE). State E_NORMAL means that the counter is not possible to be changed from outside. E_HOUR state allows change of hours parts in the counter. E_MINUTE state allows change minutes part in the counter. State E_EPSILON is empty state which refers to the E_NORMAL state.

Non formal automata description (current state, input, next state) looks as follows:

E_NORMAL *voltage change on potentiometer* $->$ E_HOUR
E_HOUR *more than 5 seconds of idle* $->$ E_MINUTE
E_HOUR *activity on switches S2 or S3* $->$ E_HOUR
E_MINUTE *more 5 then seconds of idle* $->$ E_EPSILON
E_MINUTE *activity on switches S2 or S3* $->$ E_MINUTE
E_EPSILON $->$ E_NORMAL

Time compare interruption is used for counter incrementation. The interruption TIMSK1 uses prescale factor of size 1024, where counter TCN1 is used for internal time counter for comparing with data register OCR1A.

Structure *time_rep* is used for time data representation. The structure contains: *total_time*, *hours*, *minutes* and second variables. Variable *total_time* is used for saving entire time in seconds while other variables are used only for time manipulating.

3.2 Sensors

Reading from sensors is obtained by function called analogRead from corresponding ports.

3.2.1 Temperature

Temperature sensor is linear and connected to 5 V. Function *analogRead* returns value from range 0 to 1023 and result is multiplicated by 100 for scaling purpose, therefore the equation of it is following: *(analogRead(VOLTAGE_PORT) * 5 / 1023) * 100*

3.2.2 Voltage

Voltage is read from potentiometer and connected to 5V. Function *analogRead* returns value from range 0 to 1023, so the equation of it is following: *(analogRead(VOLTAGE_PORT) * 5 / 1023)*

3.3 Serial communication

Purpose of serial communication is sending measured data. Measured data is saved in structure *structure* the structure has 3 arrays for temperature, time and voltage measurements. Counter , counter, in structure responsible for pointing to the memory. The size of arrays depends of macro *MEASUREMENT_SIZE*

Activation of serial communication is provided via switch S1.