Lesson 4

Programming Arduino Examples 9.3 and 9.4

Examples 9.3 and 9.4

- Programming the serial reading from an ADC output from an analog sensor with the ADC
- Programming Arduino for using of timer library-functions

Example 9.3 Sensor ADC output reading using serial input SPI

- A temperature sensor used for measuring between 0 degree and 100 degree Celsius
- The sensor sends analog output at an analog input of a 10 bit ADC
- An RH% sensor can also be used in similar manner where measured value is in RH% in place of degree Celsius

Example 9.3 Sensor ADC Output Reading Every Hour Using Serial Input SPI

- Set interfacing circuit such that the ADC output for sensor at 100 degree or 100% = decimal 1023 (=binary 1111111111) and = decimal 0 (=0000000000) for 0 degree or %
- ADC output coverts to serial by a parallel input to serial-output (PISO) converter
- The serial output connects to serial SPI input pin at Arduino Uno board

Test of Running state of the program

- One hour wait loop programming for sensor reading every hour
- Test performed by LED On-Off states using a blinking program for blink at each 3 s interval

Example 9.3

- Declaring the data types, constants, variables and functions used.
- #include <SPI.h> for using SPI pins
- #include <util.h> for IO utility functions. Includes UART interface, which connects to computer for display of messages on computer-screen
- IDE software provides the functions for display using serial interface output of board

Declaring functions and variables

- initial value = 0
- #define TempSensorADCinput 0
- /*calibCoeff = output change in parameter per unit rise in temperature by 1 degree Celsius.
- #define calibCoeff = 0.097752 when sensed parameter analog value is between 0 degree Celsius to 100 degree Celsius and digital ADC out is between 0 and 1023

Declaring functions and variables

- float observedV, parameter; /*observedV = Vinput from Sensor. The parameter = sensed parameter value 0….1023*/
- int internalLED = 13; /* initialise internal Port 13 Digital IO Pin LED for test Function. */
- char [ ] unit; /* degree Celsius, %, any other for sensed parameter*/

Setup ( )

- Same as in earlier Examples
- Add initial values
- parameter = 0.0; // Declare the initial value of the parameter ADC output
- observedV= 0.0; Initial value observe = 0 mV

loop ( )

- Run the following functions
- observedV = analogRead (TempSensorADCinput);
- parameter = calibCoeff*obesrvedV *1023/3.3;
- Serial.print (“Temperature =”); //Assume sensed parameter is temperature
- Serial.println (parameter, unit);
- test ( );

test ( )

- Run the following functions
- digitalWrite (internalLED, HIGH);}
- for (int i = 1, i<=600) , i++) { delay (3000);
- digitalWrite (internalLED, LOW); delay (3000); // Wait 6 s
- digitalWrite (internalLED, HIGH); for internal LED blink every 6 s during hour loop after which again read sensed value.

Example 9.4

- Arduino Timer functions required in number of applications
- A number of timer libraries available
- MsTimer ( ); A set of timer functions library available at website

Example 9.4

- Millisecond timer MsTimer two states, running or timeout after preset time
- Two functions set() and start().
- First one to set the timer for interrupt after a preset interval
- Second one to start running of the timer

Example 9.4

- Declaring the data types, constants, variables and functions used.
- #include <MsTimer2.h>
- Declare void action ( ) { /* Write statements for actions on preset time over, for example change of output at an IO pin*/

Setup ( )

- /* Set the millisecond timer to execute the function action ( ) after 3000 ms and start the timer*/
- MsTimer2: : set (3000, action);
- MsTimer2: : start ();

loop ( )

- Run the following functions
- loop ( ); Define statements for execution after interrupt action function finishes

Summary

We learnt

- Programming the serial reading hourly of an ADC output from an analog sensor
- Testing using internal LED blinking at every 6 s at the Arduino
- Programming Arduino for using of timer library-function MsTimer and actions on timer time-outs

End of Lesson 4 on
Programming Arduino Examples 9.3 and 9.4