```
///*****************************************************************************
// Program Description: Batch Moisture "Speedy" and PI Interface.
// FileName:  speedy_w6100.ino
// Author:    Jim Abraham
// Date:      5-22-19
// rev:
//*****************************************************************************
#include <SPI.h>
#include <Ethernet.h>
#include <EEPROM.h>
#include "MgsModbus.h"  // modbus TCP IP interface
//-----------------------------------------------------------------------------
MgsModbus Mb;
//-----------------------------------------------------------------------------
char inByte;     // incoming serial byte (characters)
char chr[10];
String inByte1;  // incoming serial byte (string)
String inByte2;  // incoming serial byte (string)
int charcnt=0,charcnt1=0,charflag=0;
int bmi00=0,bmf00=0;
int start_measure=0,gotdata=0,in_data=0,cont_read=0;
int measure=0,measure_time=0,measure_count=0,reading=0,reading_done=0;

//-----------------------------------------------------------------------------
// Ethernet settings (depending on MAC and Local network)
//-----------------------------------------------------------------------------
  byte mac[] = {0x90, 0xA2, 0xDA, 0x0E, 0x94, 0xB5 };
  IPAddress ip(10,53,10,166);
  IPAddress myDns(10,60,11,130);
  IPAddress gateway(10,53,11,254);
  IPAddress subnet(255,255,254,0);
//-----------------------------------------------------------------------------

//=============================================================================
// SET-UP
//=============================================================================
void setup()
{
//-----------------------------------------------------------------
// AND MF-50 Moisture Analyzer
//-----------------------------------------------------------------
// Moisture content = ST,+00000.00spsp%crlf
// MF-50 Commands:
   // Q   = Output the current data.
   // SIR = Output data continuously.
   // C   = Stop data output by SIR command.
   // QM  = Output data during measurement only.
//-----------------------------------------------------------------
// Serial setup for MF-50
//-----------------------------------------------------------------
  Serial.begin(2400,SERIAL_7E1);   // To Analyzer RS-232 port.

//-----------------------------------------------------------------
// initialize the ethernet device
//-----------------------------------------------------------------
  Ethernet.begin(mac, ip, myDns, gateway, subnet);   // start ethernet interface
//  Mb.MbData[0] = 0,  Mb.MbData[1] = 0,  Mb.MbData[2] = 0,  Mb.MbData[3] = 0;
//  Mb.MbData[4] = 0,  Mb.MbData[5] = 0,  Mb.MbData[6] = 0,  Mb.MbData[7] = 0;
//  Mb.MbData[8] = 0,  Mb.MbData[9] = 0,  Mb.MbData[10] = 0, Mb.MbData[11] = 0;

//-----------------------------------------------------------------
// Set-up I/O
//-----------------------------------------------------------------
  pinMode(2, OUTPUT);    // sets the digital pin 2 as output, WDT
  pinMode(3, OUTPUT);    // sets the digital pin 3 as output, Measure

//-----------------------------------------------------------------
// Get Previous Moisture Reading.
//-----------------------------------------------------------------
  bmi00=EEPROM.read(1);
  bmf00=EEPROM.read(2);
```

```
//---------------------------------------------------------------------
// Start-up delay
//---------------------------------------------------------------------
  delay(1000);

//---------------------------------------------------------------------
// Check if Measuring Moisture. (Send data only when measuring)
// Send command to analyzer to check if measuring batch moisture.
//---------------------------------------------------------------------
  Serial.println("C"); //Stop
  Serial.println("QM"); // print QM with CR and LF.

//---------------------------------------------------------------------
}// end of SET-UP function.

//=============================================================================
// MAIN LOOP
//=============================================================================
void loop()
{
  WDT_LED_Update();                 // Update WDT LED
  Chk_Start_Measuring();            // Check if Measurements started.
  Get_Measurement_Readings();       // Get Analyzer readings.
  Send_Data_to_PI();                // Send data to PI.
}


//=============================================================================
// ROUTINES
//=============================================================================

//---------------------------------------------------------------------
// WDT led (blink on/off)
//---------------------------------------------------------------------
 void WDT_LED_Update()
 {
  digitalWrite(2, HIGH); // WDT LED on
  delay(500);            // delay 1/2 second
  digitalWrite(2, LOW);  // WDT LED off
  delay(500);            // delay 1/2 second
 }

//---------------------------------------------------------------------
// Check if Measuring Moisture
//---------------------------------------------------------------------
 void Chk_Start_Measuring()
 {
  charcnt=0, charcnt1=0, charflag=0;
  if (start_measure==0)
    {
      Serial.println("QM");
      digitalWrite(3, LOW);      // Measuring LED off
      measure_time = 0, measure_count = 0;
    }

  if (start_measure==1)
    {
      digitalWrite(3, HIGH);     // Measuring LED on
      if (cont_read==0)          // Measurement Done.
        {
          Serial.println("QM");
        }

      if (cont_read==1)
        {
          measure_time = measure_time + 1;
          if (measure_time<=3)
            {
              Serial.println("Q");
            }
          else
```

```
            {
              start_measure=0;
              cont_read=0;
              Serial.println("QM");
              gotdata=3; // measure end.
            }
        }
    }
 }

//-------------------------------------------------------------------------
// Send command to analyzer to check if measuring batch moisture.
// Check if analyzer is sending Measurements
//-------------------------------------------------------------------------
 void Get_Measurement_Readings()
 {
  if(Serial.available())
   {
//-------------------------------------------------------------------------
// keep reading from Serial until there are bytes in the serial buffer
//                       123456789012 3  4  5 6  7
// Check for Data Stream: ST,+00000.00 sp sp % cr lf
//-------------------------------------------------------------------------
  while (Serial.available()>0)
        {
           reading=1, reading_done=0, start_measure=1;
           inByte = Serial.read();

           if (inByte == 'S' and charflag==0)
               {
                  charflag=1;
               }
           // Get reading
           if (charflag==1)
               {
                  charcnt=charcnt+1;
                  if (charcnt>7 and charcnt<13)
                  {
                      charcnt1=charcnt1+1;
                      chr[charcnt1] = inByte; // get current reading
                  }
               }
         }//while
           measure_count = measure_count + 2;
     }//if
    else // no bytes in serial buffer.
     {
      //Mb.MbsRun();
      if (reading==1) // check if measurement done.
      {
        reading_done=reading_done+1;
        if (reading_done>=4)
        {
          cont_read=1;
          reading=0;
          reading_done=0;
        }
      }
//-------------------------------------------------------------------------
// Get Batch Moisture data (0-10%)
// Measurement Complete!
//-------------------------------------------------------------------------
        if (gotdata==3)
          {
            start_measure=0, in_data=0, gotdata=0;
            //-----------------------------------------------------
            // Save Final Moisture Reading.
            //-----------------------------------------------------
            inByte1 ="";
            inByte1 += chr[1];        // x0.00
            inByte1 += chr[2];        // 0x.00
```

```
          //inByte1 += chr[3];
            inByte2 ="";
            inByte2 += chr[4];        // 00.x0
            inByte2 += chr[5];        // 00.0x

            if (measure_count>75 and inByte1 != "00")
              {
                bmi00 = inByte1.toInt();  // int part
                bmf00 = inByte2.toInt();  // fractional part
                EEPROM.update(1,bmi00);
                EEPROM.update(2,bmf00);
              }
          }
     }//else
 }

//---------------------------------------------------------------------------
// Batch Moisture data to send to PI.
//---------------------------------------------------------------------------
 void Send_Data_to_PI()
 {
   Mb.MbData[0] = (bmi00*100)+bmf00;  // make an integer to send, /100 in PI.
   Mb.MbsRun(); // need to run as often as possible...otherwise get i/o timeout.
 }
```