

DR. BOB DAVIDOV

Многоканальное устройство ввода и накопления аналоговых данных на базе MS Excel

Цель работы: познакомиться с macros технологиями для решения задач управления и мониторинга в неспециализированной среде MS Excel.

Задача работы: построение системы ввода, обработки и накопления аналоговых данных на базе электронных таблиц и USB/RS-232 интерфейса Arduino.

Приборы и принадлежности: Персональный компьютер, MS Excel, платформа Arduino UNO, МатЛАБ.

ВВЕДЕНИЕ

Электронные таблицы MS Excel можно использовать не только как средство обработки и накопления данных но и для управления реальными объектами и мониторинга, например, через интерфейс USB/RS-232. В этой работе раскрываются особенности системы чтения, обработки и накопления реальных аналоговых данных на базе Excel и платформы Arduino структурная схема которой показана на Рис. 1.

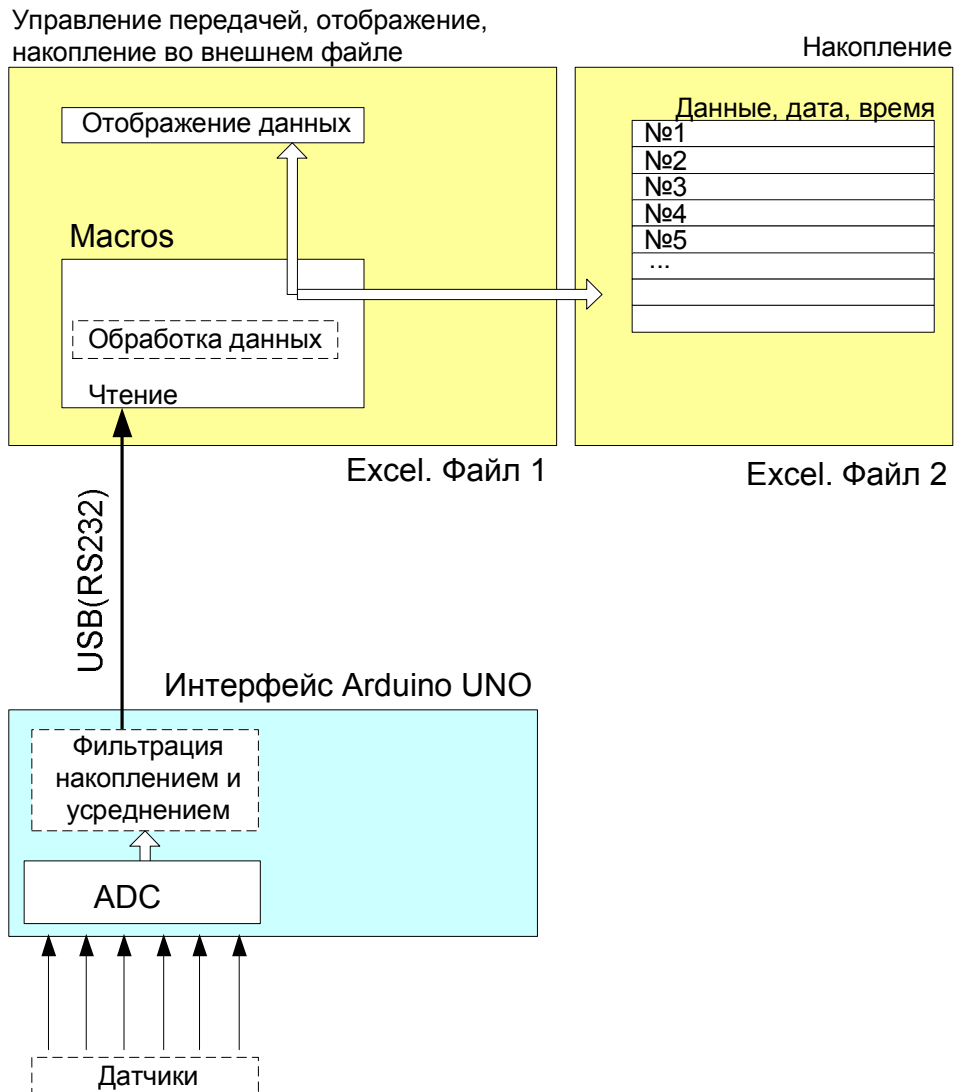


Рис. 1. Блок-схема системы накопления аналоговых данных, их фильтрации, обработки и накопления на базе MS Excel и платформы Arduino.

ОБЩИЕ СВЕДЕНИЯ

Платформа Arduino UNO

В этой работе Arduino UNO работает как устройство ввода аналоговых сигналов на максимальной частоте, накопления и усреднения данных перед их отправкой в последовательный канал передачи.



Рис. 2. Периферия платформы ARDUINO (DFRduino) UNO.

Платформа Arduino UNO имеет следующие характеристики.

- Микроконтроллер ATmega 328, частота 16 MHz,
- Напряжение питания: 7-12 В (6-20 В - предел). Вход используется для подачи питания от внешнего источника (в отсутствие 5 В от разъема USB).
- 32 Кб флэш память (2К занято загрузчиком),
- 2Кб ОЗУ
- 1Кб EEPROM
- 6 аналоговых (0-5В, 10бит, 0.1мс) вводов и 14 цифровых вводов/выводов с 6 PWM (ШИМ) выходами (~490 Гц, 0 .. 255)
- Входной ток ADC: 0 .. 1 мкА
- Встроенные USB-COM (300, ..., 115200 бод), SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), и I2C: 4 (SDA) и 5 (SCL) каналы связи;
- 2 внешних прерывания

Примечание. Порядок подключения драйвера платформы, среды разработки, и базовых библиотек можно найти в “Создание интерактивных объектов и сред на базе платформы Arduino. Часть 1” на <http://www.vr-online.ru/blog/sozдание-interaktivnyh-obektov-i-sred-na-baze-platformy-arduino-chast-2-7969> и в “Создание интерактивных объектов и сред на базе платформы Arduino. Часть 2” на <http://www.vr-online.ru/blog/sozдание-interaktivnyh-obektov-i-sred-na-baze-platformy-arduino-chast-2-7969>

Датчик температуры LM335

Особенности датчика температуры LM335:

- Непосредственное измерение температуры в град.К
- Диапазон измеряемых температур -40 .. +100 °С
- Точность 1°С
- Отношение выходного напряжения к температуре 10мВ/°С

- Выходное рабочее напряжение при 25С и токе 1мА 2.92 .. 2.98 В
- Постоянная времени в стоячем воздухе 80 сек
- движущемся воздухе (~5 мм/сек) 10 сек
- движущемся масле 1 сек
- Динамический импеданс < 1 Ом
- Низкая стоимость ~18 руб.(август 2013 г.)

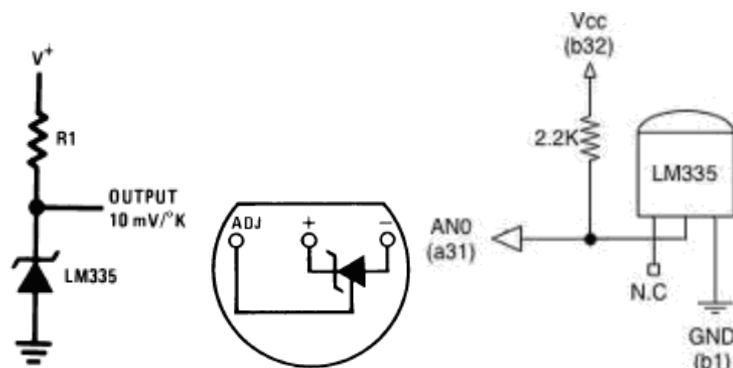


Рис. 3. Схемы включения датчика температуры LM335 без регулировки (контакт ADJ, N.C.).

Датчик влажности НН-4000-002

Датчик влажности НН-4000-002 имеет следующие характеристики.

- Диапазон рабочих температур -40 °С to 85 °С
- Диапазон измеряемой влажности 0% RH to 100 % RH,
здесь и далее **RH** – относительная влажность
- Погрешность ±5 %RH (0 .. 59% RH), ±8 %RH (60 .. 100% RH)
- Гистерезис ±3 %RH
- Время отклика 15 сек (tau) в медленно движущемся воздухе
- Повторяемость ±0.5 % RH
- Время включения 70 мсек макс.
- Питание +5.8 В / 500 μА
- Дрейф ±1.2 %RH за 5 лет; ±0.25 %RH за год
- Стабильность при 50% RH ±1.2 %RH

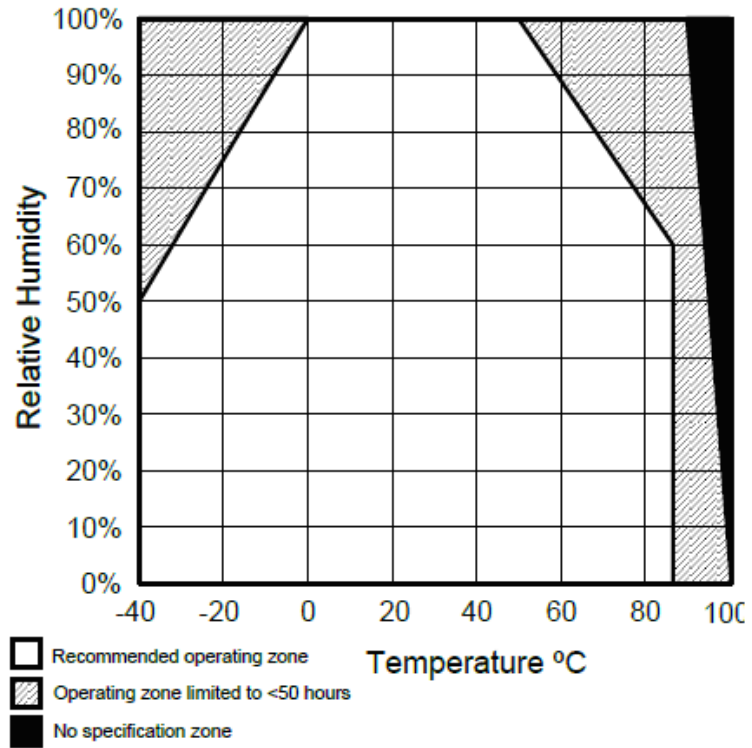


Рис. 4. Рекомендуемая рабочая зона температуры и влажности.

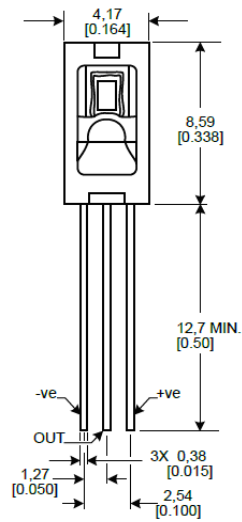


Рис. 5. Размеры датчика HIH-4000-002.

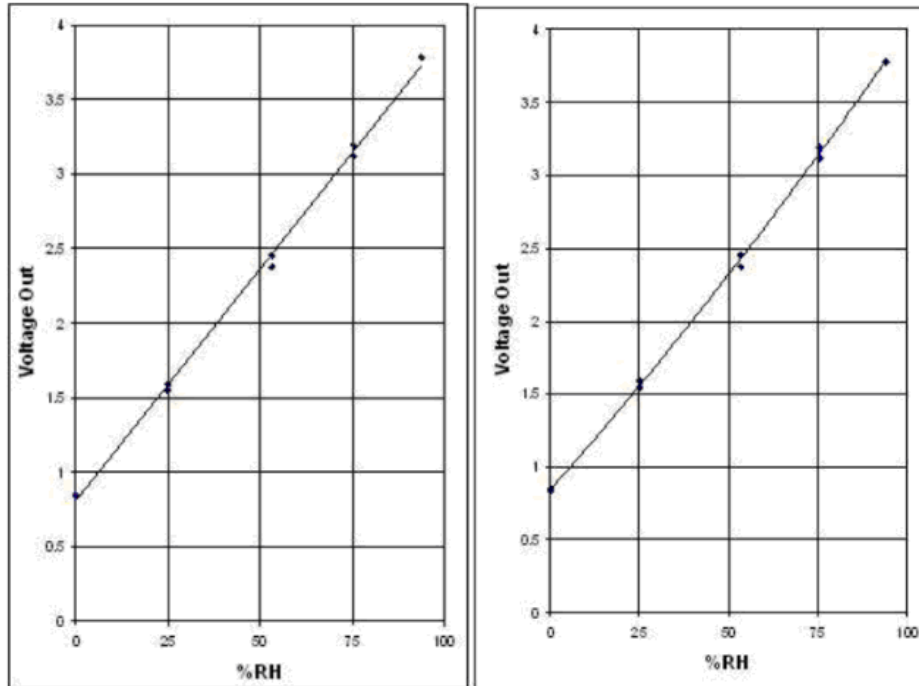


Рис. 6. Зависимость выходного напряжения датчика от влажности аппроксимированная полиномом первой (слева) и второй (справа) степени.

Для расчета влажности по выходному напряжению датчика и введению поправок по напряжению питания и температуре датчика необходимо использовать следующие отношения.

Выходное напряжение датчика:

$$V_{out} = V_{supply} (0.0062 (\text{sensor RH}) + 0.16), \text{ полином 1-ой степени}$$

$$V_{out} = 0.00003 * (\text{sensor RH})^2 + 0.0281 * (\text{sensor RH}) + 0.820 \text{ для } 25\text{C}, \text{ полином 2-ой степени}$$

Температурная компенсация:

$$V_{out} = (1.0305 + 0.000044T - 0.0000011T^2) * (\text{Sensor RH}) + (0.9237 - 0.0041T + 0.000040T^2), \text{ где } T - \text{ температура в } ^\circ\text{C}$$

Точка росы

Для вычисления **точки росы** (температуры, до которой должен охладиться воздух, чтобы достичь состояния насыщения водяным паром при данном влагосодержании и неизменном давлении) можно использовать показания датчиков температуры и влажности и графики, приведенные на Рис. 7.

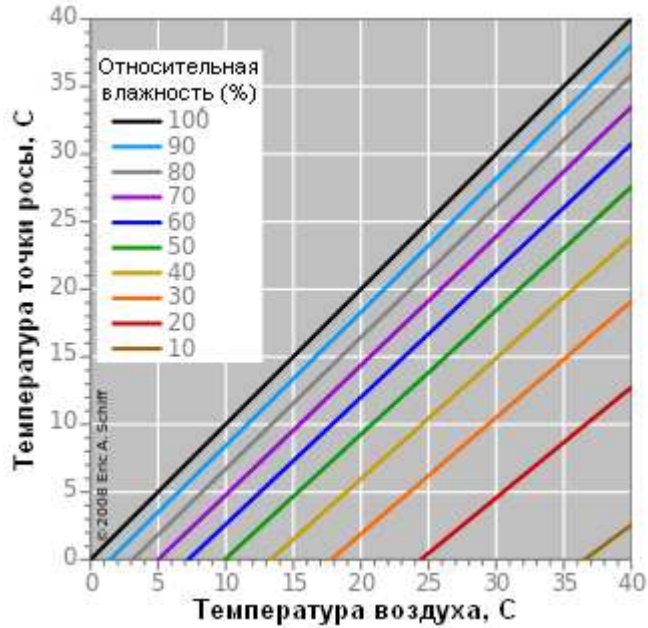


Рис. 7. Вычисление точки росы по температуре и относительной влажности.

Подключение датчиков к ADC платформы Arduino UNO

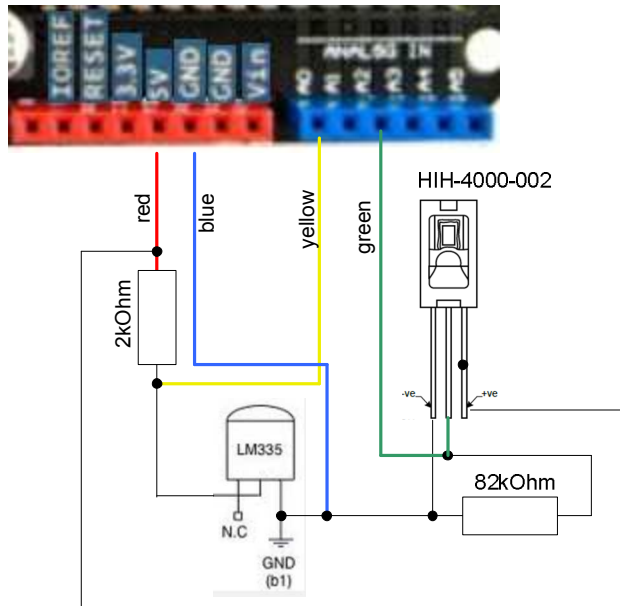


Рис. 8. Подключение датчика температуры LM335 к 1-му входу ADC (A0) и датчика влажности НН-4000-002 к 3-му входу ADC (A2).

Сигнальный провод

Для подключения удаленных (до 15 метров) датчиков используется 4-х жильный сигнальный провод с сопротивлением 28 Ом / 50 м.

Калибровка датчиков.

Для калибровки датчиков можно использовать показания аттестованных калиброванных датчиков, установленных в той же точке измерения.

Для приближенного контроля показаний датчика температуры можно использовать следующие характерные температуры.

- 0°C – температура подтаявшего льда. Внимание: температура льда из морозильной камеры холодильника может быть значительно ниже.
- 36.6°C – температура тела под языком.
- ~100°C – температура кипящей воды. Держите контакты датчика сухими. Вода может изменить межконтактное сопротивление и, соответственно, показания датчика.

Для приближенного контроля показаний датчика влажности можно использовать дыхание параметры которого обсуждаются в статье “Parameters of the exhaled breath” <http://www.sciencebits.com/exhalecondense>: “Распространенным заблуждением является то, что выдыхаемый воздух имеет 100% влажность при 37 °C. Если бы это было так, мы бы увидели конденсацию при любых условиях наружного воздуха холоднее, чем 37 °C. Очевидно, что это не так!”, “... Таким образом, **воздух, выходящий из наших уст имеет температуру около 35°C и 95% влажность.**”

Полезные команды Arduino UNO

Команда	Описание
<code>Serial.begin(9600);</code>	Установка скорости передачи данных через COM порт (RS-232) из ряда: 300, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 57600, 115200 бод (бит/с)
<code>Serial.flush();</code>	Очистка входного буфера COM порта
<code>int led = 13;</code> <code>pinMode(led, OUTPUT);</code>	Установка режима работы TTL порта Arduino (OUTPUT - на выход). К 13 порту подключен светодиод который используется для демонстрации цикличной работы Arduino
<code>digitalWrite(led, LOW);</code> <code>digitalWrite(led, HIGH);</code>	Включение и выключение светодиода платформы подключенного к 13-му TTL-порту
<code>void loop() {}</code>	Циклическое выполнение операторов помещенных в фигурные скобки
<code>int adc_0;</code> <code>adc_0 = analogRead(0);</code>	Чтение первого (нулевого) канала 10 разрядного 5В 0.1 мс АЦП
<code>Serial.write(adc_0_Lo);</code>	Запись в COM-порт байта 0 .. 127
<code>unsigned long adc_0_sum = 0;</code> <code>int loop_num = 0;</code> <code>adc_0_sum = adc_0_sum + adc_0;</code>	Накопление и подсчет ADC данных

loop_num = loop_num + 1;	
if (loop_num > 30000) { adc_0_sum = 0; }	Защита от переполнения накопления данных
if (Serial.available() > 0) {}	Проверка прибытия данных через COM порт . Используется для синхронизации Excel средой процессов в Arduino
ctrl_num = Serial.read();	Считывание данных из COM порта
adc_0 = (adc_0_sum * 16) / loop_num;	Вычисление среднего значения, используется для фильтрации АЦП сигналов и повышения виртуальной разрядности АЦП до 14 бит
adc_0_Hi = ((adc_0 >> 7) & 0x7F);	Выделение младшей половины 14-разрядного среднего
adc_0_Lo = ((adc_0 << 7) & 0x7F);	Выделение старшей половины 14-разрядного среднего
Serial.write(adc_0_Hi); // in 0 .. 127 data range Serial.write(adc_0_Lo)	Передача в COM-порт старшей и младшей половины 14-разрядного значения усредненного результата накопления 10-разрядных данных АЦП

Полезные команды Macros и MS ActiveX control

Команда	Описание
Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)	Подключение системного таймера для выполнения задержек. Пример команды “задержка на одну секунду”: Sleep 1000&
Sub CommandButton1_Click() End Sub	Начало и конец подпрограммы
Dim StopMacro As Boolean	Инициализация глобальной / локальной переменной в зависимости от того где расположен оператор Dim - за пределами или внутри программного блока.
For i = 1 To 10000 Next i	Пример оператора цикла
DoEvents If StopMacro Then Exit For	Пример выхода из цикла For. Оператор DoEvents необходим перед использованием глобальных переменных
MSComm1.CommPort = 3 MSComm1.Settings = "9600,N,8,1" MSComm1.InputLen = 0 On Error Resume Next MSComm1.PortOpen = True	Подключение к COM порту и настройка его параметров

<pre>If Err Then MsgBox "Com" & MSComm1.CommPort & ": not available. Change CommPort property to another port." Exit Sub End If</pre>	Выдача сообщения об ошибке соединения с COM-портом
<pre>MSComm1.PortOpen = False</pre>	Закрытие COM порта
<pre>Set oXL = New Excel.Application Set oBook = oXL.Workbooks.Open("D:\pc\LONG_ADC_Arduino_Read\DataBase_1.xls") Set oSheet = oBook.Worksheets("Sheet1")</pre>	Пример открытия страницы внешней (по отношению к macros) электронной таблицы Excel
<pre>Set oSheet = Nothing oBook.Close True ' without False displays Message Set oBook = Nothing oXL.Quit Set oXL = Nothing 'ActiveWorkbook.Close True 'ActiveWorkbook.Close False</pre>	Пример закрытия внешнего Excel файла
<pre>MSComm1.Output = Chr\$(1)</pre>	Запись единицы в COM порт для синхронизации процессов Arduino
<pre>b = MSComm1.Input</pre>	Считывание данных из буфера COM порта
<pre>Sheet1.Cells(5, 1).Value = i</pre>	Запись в ячейку Excel (в которой работает macros) переменной i
<pre>exampleDate = Now</pre>	Чтение системной даты и времени
<pre>Sheet1.Cells(5, 8).Value = Hour(exampleDate) Sheet1.Cells(5, 12).Value = Month(exampleDate)</pre>	Выделение из переменной exampleDate значений текущего часа и месяца
<pre>oSheet.Cells(Offset, 1).Value = Offset - 4</pre>	Запись в ячейку внешней Excel значения Offset - 4
<pre>Application.DisplayAlerts = False ThisWorkbook.Save Application.DisplayAlerts = True Application.Quit</pre>	Закрытие Excel с macros
<pre>UserForm1.Show ThisWorkbook.Save</pre>	Команда Workbook для автоматического запуска macros

Примечание. Команда <Ctrl + Break> останавливает работу macros.

Примеры ActiveX control можно найти в работе “COM/DCOM клиент и сервер автоматизации Microsoft ActiveX control” [4].

ПРИМЕРЫ ПОЛУЧЕНИЯ ПРОВЕРЕННЫХ РЕЗУЛЬТАТОВ И ВАРИАНТЫ ДЛЯ САМОКОНТРОЛЯ

Задание 1. Подключение и настройка платформы Arduino UNO.

1. Подключите два датчика температуры к первому и второму входам АЦП (А0 и А1) и четыре датчика влажности к остальным входам по схеме показанной на Рис. 8.

Примечание. Можно подключить любые другие датчики в любом порядке или не подключать их вовсе. Эти варианты потребуют соответствующей коррекции скрипта обработки и отображения аналоговых данных.

2. Загрузите интегрированную среду разработки и настройте драйвер платформы Arduino, например, как показано в разделе ПОДГОТОВКА К РАБОТЕ С ARDUINO ЧЕРЕЗ ИНТЕГРИРОВАННУЮ СРЕДУ РАЗРАБОТКИ работы “Создание интерактивных объектов и сред на базе платформы Arduino. Часть 1” <http://www.vr-online.ru/blog/sozдание-interaktivnyh-obektov-i-sred-na-baze-platformy-arduino-chast-1-7947>
3. Используя команды Arduino, приведенные в таблице выше, разработайте программу которая для каждого канала 6-канального 10-разрядного АЦП считывает показания, накапливает и подсчитывает их, а в момент поступления запроса (числа) от Excel через СОМ-порт, вычисляет среднее значение АЦП увеличенное в 16 раз (4 разряда), округляет его до 14-разрядов, и двумя байтами посылает среднее значение в СОМ-порт.

Ниже дан пример реализации заданного алгоритма.

/*

Communication with Excel in RT mode

created 24 August 2013

modified

by Bob Davidov

Reads from 10-bit ADC (6 channels)

Passes the ADC averages of 14 bits to COM port

by two bytes each in 0 .. 127 range

Order of sensors connected to ADC ports:

adc_0, adc_1, adc_2, adc_3, adc_4, adc_5,

temp1, temp2, hum1, hum2, hum3, hum4

This example code is in the public domain.

*/

byte ctrl_num; // control num of 0 .. 127 from Excel

int led = 13; // control LED

int adc_0; // 10 bit ADC 0 .. 5V

int adc_1;

int adc_2;

int adc_3;

int adc_4;

```

int adc_5;

unsigned long adc_0_sum = 0; // for sum of adc_i
unsigned long adc_1_sum = 0;
unsigned long adc_2_sum = 0;
unsigned long adc_3_sum = 0;
unsigned long adc_4_sum = 0;
unsigned long adc_5_sum = 0;

int loop_num = 0; // number of asc_i reading

byte adc_0_Hi; // high byte (in 0 .. 127 range) of 14 bit average
byte adc_1_Hi;
byte adc_2_Hi;
byte adc_3_Hi;
byte adc_4_Hi;
byte adc_5_Hi;

byte adc_0_Lo;
byte adc_1_Lo;
byte adc_2_Lo;
byte adc_3_Lo;
byte adc_4_Lo;
byte adc_5_Lo;

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600); //300, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 57600, 115200
  Serial.flush(); // clear input buffer

  pinMode(led, OUTPUT);
}

void loop() {
  // read ADC
  adc_0 = analogRead(0);
  adc_1 = analogRead(1);
  adc_2 = analogRead(2);
  adc_3 = analogRead(3);
  adc_4 = analogRead(4);
  adc_5 = analogRead(5);

  // sum of captured data
  adc_0_sum = adc_0_sum + adc_0;
  adc_1_sum = adc_1_sum + adc_1;
  adc_2_sum = adc_2_sum + adc_2;
  adc_3_sum = adc_3_sum + adc_3;

```

```

adc_4_sum = adc_4_sum + adc_4;
adc_5_sum = adc_5_sum + adc_5;

// number of adc reading
loop_num = loop_num + 1; // ~ 0 .. 180 loppes per 0.1 ms

if (loop_num > 30000) { // protection against long delay in data reading
  adc_0_sum = 0;
  adc_1_sum = 0;
  adc_2_sum = 0;
  adc_3_sum = 0;
  adc_4_sum = 0;
  adc_5_sum = 0;
}

// get any incoming bytes:
if (Serial.available() > 0) {
  ctrl_num = Serial.read(); // 0 .. 255 from excel
  switch (ctrl_num) {
  case 1:
    digitalWrite(led, HIGH); // switch on the LED

    adc_0 = (adc_0_sum * 16) / loop_num; // int: average of adc x 16
    adc_1 = (adc_1_sum * 16) / loop_num;
    adc_2 = (adc_2_sum * 16) / loop_num;
    adc_3 = (adc_3_sum * 16) / loop_num;
    adc_4 = (adc_4_sum * 16) / loop_num;
    adc_5 = (adc_5_sum * 16) / loop_num;

    adc_0_Hi = ((adc_0 >> 7) & 0x7F); // byte: high half (7 bit) of 14 bit adc data (int word)
    adc_1_Hi = ((adc_1 >> 7) & 0x7F);
    adc_2_Hi = ((adc_2 >> 7) & 0x7F);
    adc_3_Hi = ((adc_3 >> 7) & 0x7F);
    adc_4_Hi = ((adc_4 >> 7) & 0x7F);
    adc_5_Hi = ((adc_5 >> 7) & 0x7F);

    adc_0_Lo = (adc_0 & 0x7F); // byte: low half of 14 bit adc data (int word)
    adc_1_Lo = (adc_1 & 0x7F);
    adc_2_Lo = (adc_2 & 0x7F);
    adc_3_Lo = (adc_3 & 0x7F);
    adc_4_Lo = (adc_4 & 0x7F);
    adc_5_Lo = (adc_5 & 0x7F);

    Serial.write(adc_0_Hi); // in 0 .. 127 data range
    Serial.write(adc_0_Lo);
    Serial.write(adc_1_Hi);
    Serial.write(adc_1_Lo);

```

```

Serial.write(adc_2_Hi);
Serial.write(adc_2_Lo);
Serial.write(adc_3_Hi);
Serial.write(adc_3_Lo);
Serial.write(adc_4_Hi);
Serial.write(adc_4_Lo);
Serial.write(adc_5_Hi);
Serial.write(adc_5_Lo);

// reset sum of adc data
adc_0_sum = 0;
adc_1_sum = 0;
adc_2_sum = 0;
adc_3_sum = 0;
adc_4_sum = 0;
adc_5_sum = 0;

loop_num = 0;

digitalWrite(led, LOW); // switch off the LED

break;

default:
break;
}
}
}

```

4. При помощи интегрированной среды запишите программу в память платформы Arduino.

Задание 2. Настройка MS Excel на работу с COM портом.

1. Установите соединение Excel с COM-портом, например, как показано в разделе ОБЩИЕ СВЕДЕНИЯ и в задании 2 работы “ Управление из Excel через COM порт” <http://www.vr-online.ru/blog/upravlenie-iz-excel-cherez-com-port-8475>
2. Проверьте работу соединения

Задание 3. Считывание аналоговых данных в Excel.

1. Используя команды Macros, приведенные в таблице выше, разработайте программу которая циклически устанавливает связь с COM-портом, через COM порт передает платформе Arduino код синхронизации, принимает усредненные данные АЦП платформы, системную дату и время операционной среды и отображает принятые данные в ячейках электронной таблицы и разрывает соединение с COM-портом.

Примечание. Для получения корректного значения усредненного сигнала АЦП не забудьте поделить на 16 увеличенное в Arduino значение АЦП.

Ниже дан пример реализации заданного алгоритма.
Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Dim StopMacro As Boolean

Private Sub CommandButton1_Click()

Dim b() As Byte ' for COM port Input buffer reading
' for saving data in external xls file:

For i = 1 To 10000

MSComm1.CommPort = 3
MSComm1.Settings = "9600,N,8,1"
MSComm1.InputLen = 0
On Error Resume Next

MSComm1.PortOpen = True
Sleep 1000&'Delay for 1 second

'start Arduino sending of ADC 14 bit averages into COM port
MSComm1.Output = Chr\$(1) 'LED High
Sleep 500&'Delay for 0.5 second

b = MSComm1.Input 'reading ADC data
'ADC 10 bit average = (Hi byte x 128 + Lo byte) / 16
'Arduino averages ~730 ADC samples each excel loop with 0.5 s delay

Sheet1.Cells(5, 1).Value = i

'
VVVV Offset in voltage
Sheet1.Cells(5, 2).Value = (((b(UBound(b) - 23)) * 128 + b(UBound(b) - 21)) / 16) * (5 / 1024) - 2.92) * 100 ' Temperature 1
Sheet1.Cells(5, 3).Value = (((b(UBound(b) - 19)) * 128 + b(UBound(b) - 17)) / 16) * (5 / 1024) - 2.92) * 100 ' Temperature 2
Sheet1.Cells(5, 4).Value = (((b(UBound(b) - 15)) * 128 + b(UBound(b) - 13)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity

1

Sheet1.Cells(5, 5).Value = (((b(UBound(b) - 11)) * 128 + b(UBound(b) - 9)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity 2
Sheet1.Cells(5, 6).Value = (((b(UBound(b) - 7)) * 128 + b(UBound(b) - 5)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity 3
Sheet1.Cells(5, 7).Value = (((b(UBound(b) - 3)) * 128 + b(UBound(b) - 1)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity 4
'
^^^ sensor supply

exampleDate = Now
Sheet1.Cells(5, 8).Value = Hour(exampleDate)
Sheet1.Cells(5, 9).Value = Minute(exampleDate)
Sheet1.Cells(5, 10).Value = Second(exampleDate)
Sheet1.Cells(5, 11).Value = Day(exampleDate)

Sheet1.Cells(5, 12).Value = Month(exampleDate)

Sheet1.Cells(5, 13).Value = Year(exampleDate)

' close COM port

MSComm1.PortOpen = False

Next i

End Sub

2. Настройте отображение данных как показано ниже.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<i>Измерение и запись температуры и влажности</i>												
2													
3		Температура, С		Влажность, %				Время			Дата		
4		1	2	1	2	3	4	Час	Мин	Сек	День	Месяц	Год
5	7	12.5	12.7	87.4	88.3	86.2	82.1	8	11	42	3	9	2013
6													
7													

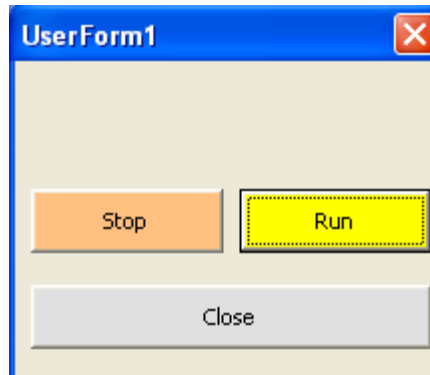
Задание 4. Запись данных в отдельный Excel файл.

1. Добавьте в программу задания 3 команды открытия внешнего Excel файла (см. таблицу Macros выше), записи данных во внешний файл и закрытия внешнего файла.
2. Настройте таблицу внешнего файла как показано ниже.

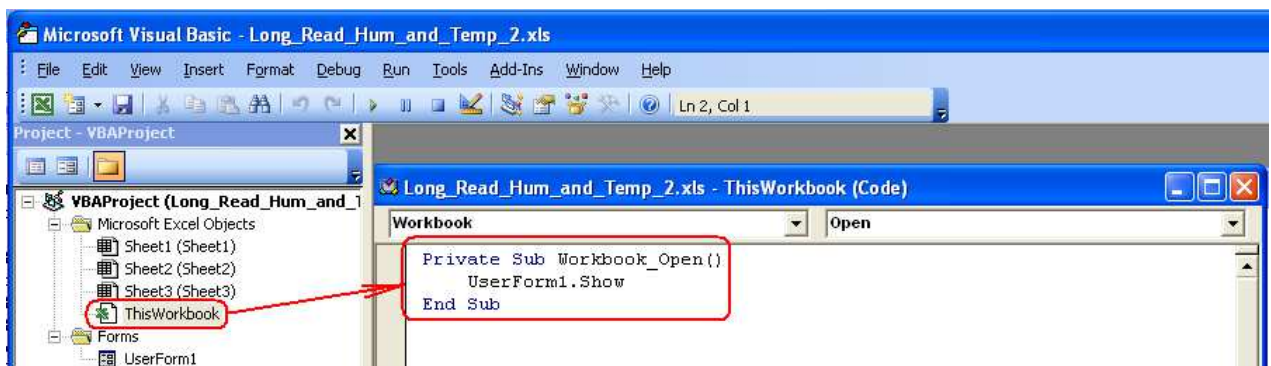
	A	B	C	D	E	F	G	H	I
1	<i>Наблюдение за температурой и влажностью</i>								
2									
3	Start cell	Дата, Время	Температура, С		Влажность, %				Время, сек
4	12		1	2	1	2	3	4	
5	1	9/3/2013 8:11	12.4	12.8	87.4	88.3	86.2	82.1	28
6	2	9/3/2013 8:11	12.5	12.8	87.4	88.3	86.2	82.1	30
7	3	9/3/2013 8:11	12.4	12.8	87.4	88.3	86.2	82.1	33
8	4	9/3/2013 8:11	12.4	12.8	87.4	88.3	86.2	82.1	35
9	5	9/3/2013 8:11	12.5	12.7	87.4	88.3	86.2	82.1	37
10	6	9/3/2013 8:11	12.4	12.7	87.4	88.3	86.2	82.1	40
11	7	9/3/2013 8:11	12.5	12.7	87.4	88.3	86.2	82.1	42
12									
13									

Задание 5. Быстрая загрузка, остановка мониторинга и выгрузка рабочей программы.

1. Используя команды Macros, приведенные в таблице выше, разработайте показанный ниже интерфейс, в котором кнопка “Run” запускает циклический процесс чтения, отображения и накопления аналоговых данных задания 3 и 4, кнопка “Stop” останавливает циклический процесс, а кнопка “Close” закрывает Excel файл с Macros.



2. Обеспечьте автоматический запуск Macros при открытии Excel файла следующей командой Workbook.



3. Сравните собственный код с примером кода выполнения задания:

```
Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Dim StopMacro As Boolean
```

```
Private Sub CommandButton1_Click()
```

```
Dim b() As Byte ' for COM port Input buffer reading
' for saving data in external xls file:
```

```
Dim oXL As Excel.Application, oBook As Excel.Workbook, oSheet As Excel.Worksheet, vValue As Variant
Dim Offset As Integer ' or Long
```

```
StopMacro = False
```

```
For i = 1 To 10000
```

```
DoEvents 'statement in the loop to click the other button
If StopMacro Then Exit For 'Sub
```

```
MSComm1.CommPort = 3
MSComm1.Settings = "9600,N,8,1"
MSComm1.InputLen = 0
On Error Resume Next
```

```
MSComm1.PortOpen = True
```

```
If Err Then
```

```
MsgBox "Com" & MSComm1.CommPort & ": not available. Change CommPort property to another port."
```

```
Exit Sub
```

```
End If
```

```
' using the Excel automation objects to open xls file
```

```
Set oXL = New Excel.Application
```

```
Set oBook = oXL.Workbooks.Open("D:\pc\LONG_ADC_Arduino_Read\DataBase_1.xls")
```

```
Set oSheet = oBook.Worksheets("Sheet1")
```

```
Sleep 2000& 'Delay for 2 second
```

```
'start Arduino sending of ADC 14 bit averages into COM port
```

```
MSComm1.Output = Chr$(1) 'LED High
```

```
Sleep 500& 'Delay for 0.5 second
```

```
b = MSComm1.Input 'reading ADC data
```

```
'ADC 10 bit average = (Hi byte x 128 + Lo byte) / 16
```

```
'Arduino averages ~730 ADC samples each excel loop with 0.5 s delay
```

```
Sheet1.Cells(5, 1).Value = i
```

```
,
```

VVVV Offset in voltage

```
Sheet1.Cells(5, 2).Value = (((b(UBound(b) - 23)) * 128 + b(UBound(b) - 21)) / 16) * (5 / 1024) - 2.92) * 100 ' Temperature 1
```

```
Sheet1.Cells(5, 3).Value = (((b(UBound(b) - 19)) * 128 + b(UBound(b) - 17)) / 16) * (5 / 1024) - 2.92) * 100 ' Temperature 2
```

```
Sheet1.Cells(5, 4).Value = (((b(UBound(b) - 15)) * 128 + b(UBound(b) - 13)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity
```

```
1
```

```
Sheet1.Cells(5, 5).Value = (((b(UBound(b) - 11)) * 128 + b(UBound(b) - 9)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity
```

```
2
```

```
Sheet1.Cells(5, 6).Value = (((b(UBound(b) - 7)) * 128 + b(UBound(b) - 5)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity 3
```

```
Sheet1.Cells(5, 7).Value = (((b(UBound(b) - 3)) * 128 + b(UBound(b) - 1)) / 16) * (5 / 1024) / 4.8 - 0.16) / 0.0062 ' Humidity 4
```

```
,
```

^^ sensor supply

```
exampleDate = Now
```

```
Sheet1.Cells(5, 8).Value = Hour(exampleDate)
```

```
Sheet1.Cells(5, 9).Value = Minute(exampleDate)
```

```
Sheet1.Cells(5, 10).Value = Second(exampleDate)
```

```
Sheet1.Cells(5, 11).Value = Day(exampleDate)
```

```
Sheet1.Cells(5, 12).Value = Month(exampleDate)
```

```
Sheet1.Cells(5, 13).Value = Year(exampleDate)
```

```
Offset = oSheet.Cells(4, 1).Value ' Index offset
```

```
oSheet.Cells(Offset, 1).Value = Offset - 4
```

```
oSheet.Cells(Offset, 2).Value = exampleDate
```

```
oSheet.Cells(Offset, 3).Value = Sheet1.Cells(5, 2).Value
```

```
oSheet.Cells(Offset, 4).Value = Sheet1.Cells(5, 3).Value
```

```
oSheet.Cells(Offset, 5).Value = Sheet1.Cells(5, 4).Value
oSheet.Cells(Offset, 6).Value = Sheet1.Cells(5, 5).Value
oSheet.Cells(Offset, 7).Value = Sheet1.Cells(5, 6).Value
oSheet.Cells(Offset, 8).Value = Sheet1.Cells(5, 7).Value
oSheet.Cells(Offset, 9).Value = Sheet1.Cells(5, 10).Value
```

```
Offset = Offset + 1
```

```
oSheet.Cells(4, 1).Value = Offset    ' Cell offset
```

```
' close external xls file
Set oSheet = Nothing
oBook.Close True ' without False displays Message
Set oBook = Nothing
oXL.Quit
Set oXL = Nothing
```

```
'ActiveWorkbook.Close True
'ActiveWorkbook.Close False
```

```
' close COM port
MSComm1.PortOpen = False
```

```
Next i
Exit Sub
```

```
Application.CommandBars(1).Reset 'reset the main Worksheet Menu bar
```

```
End Sub
```

```
Private Sub CommandButton2_Click()
    StopMacro = True
End Sub
```

```
Private Sub CommandButton3_Click()
'End 'Comment: It Exit all macros
'<Ctrl + Break> stops running macro
```

```
The following closes Excel Application:
Application.DisplayAlerts = False
ThisWorkbook.Save
Application.DisplayAlerts = True
Application.Quit
End Sub
```

4. Проверьте работоспособность разработанной системы мониторинга которая считывает аналоговые сигналы, фильтрует их, обрабатывает, отображает и накапливает. Система имеет простой интерфейс который может автоматически запускать при открытии Excel файла, имеет возможность ручной остановки процесса мониторинга и выхода из системы.

Задание 6. Накопление данных в xls файле под управлением МатЛАБ.

1. Замените XLS модуль управления передачей, обработки и записи данных (см. Рис. 1, Задание 3) аналогичным модулем на базе m-файла МатЛАБ (см. Рис. 9) использующего для записи данных в xls файл технологию Microsoft ActiveX control.

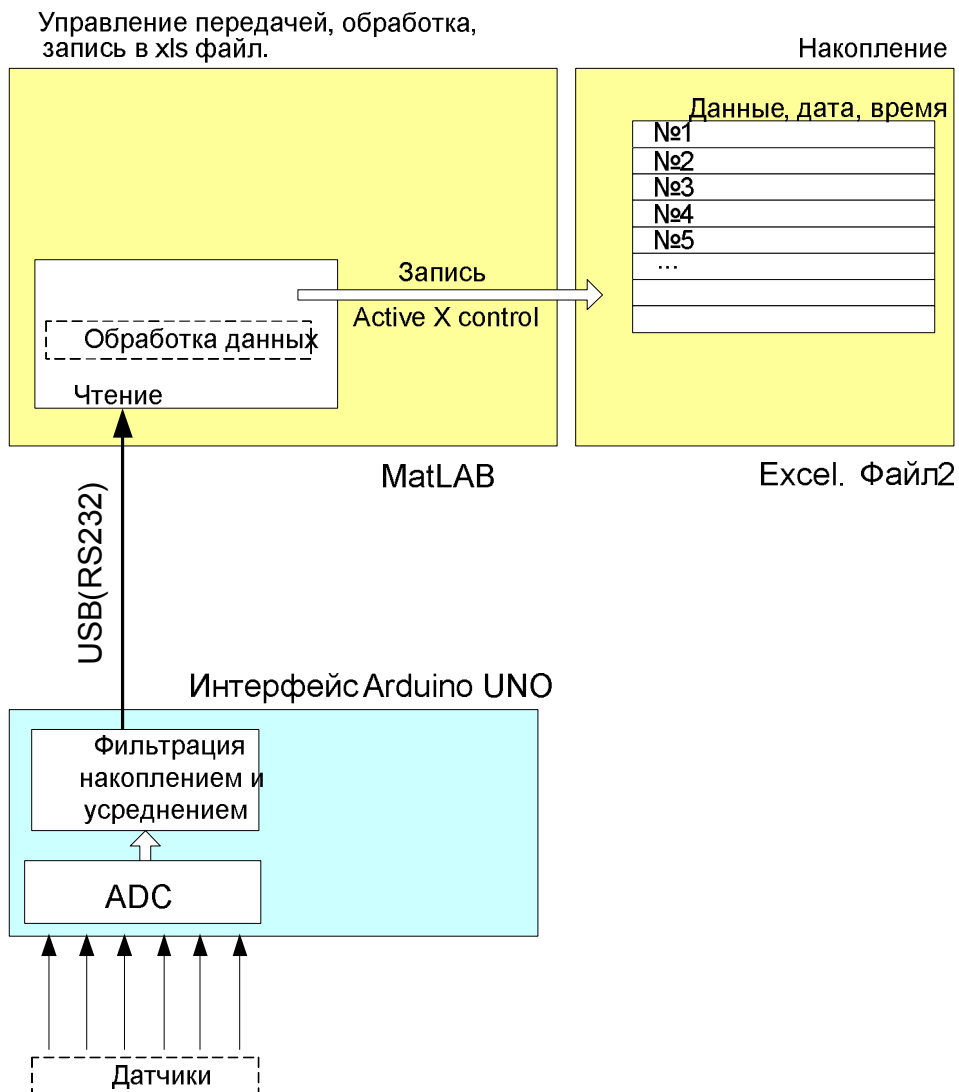


Рис. 9. Блок-схема системы накопления аналоговых данных, их фильтрации, обработки и накопления на базе МатЛАБ, MS Excel и платформы Arduino.

Пример m-кода модуля МатЛАБ использующего технологию ActiveX control для записи обработанных данных канала RS-232 в MS Excel:

```
% Arduino initializing
s = serial('COM3','BaudRate',9600,'DataBits',8,'Parity', 'none',
'FlowControl', 'none');
set (s, 'Timeout', 0.1);
fopen (s);

% xls initializing
filename = 'DataBase_1.xls'; % existing file
filename = fullfile (pwd, filename);
hExcel = actxserver ('excel.application');
%set(hExcel, 'Visible', 1);
Workbook = hExcel.Workbooks;

%Read xls data
invoke(Workbook,'Open',filename);
LineNum = hExcel.Activesheet.get ('Range', 'A4');
line_num = LineNum.value;

if isnan (line_num) % is A4 empty ?
    line_num = 7;
end
pause(1);

for i = 1:10
    fprintf(s,'1');
    In = fread(s);
    if ~isempty (In)
        temp_1 = (((In(1) * 128 + In(2)) / 16) * (5 / 1024) - 2.87) * 100;
        temp_2 = (((In(3) * 128 + In(4)) / 16) * (5 / 1024) - 2.87) * 100;
        hum_1 = (((In(5)*128+In(6))/16) * (5 / 1024) / 4.8 - 0.16) / 0.0062;
        hum_2 = (((In(7)*128+In(8))/16) * (5 / 1024) / 4.8 - 0.16) / 0.0062;
        hum_3 = (((In(9)*128+In(10))/16)* (5 / 1024) / 4.8 - 0.16) / 0.0062;
        hum_4 = (((In(11)*128+In(12))/16)* (5 / 1024) / 4.8 - 0.16) / 0.0062;

        data = [temp_1 temp_2 hum_1 hum_2 hum_3 hum_4];

        range = ['A' num2str(line_num)];
        records = hExcel.Activesheet.get ('Range', range);
        records.value = line_num - 4;

        range = ['B' num2str(line_num)];
        date_time = hExcel.Activesheet.get ('Range', range);
        date_time_now = [datestr(now, 23) ' ' datestr(now, 13)];
        date_time.value = date_time_now;

        range = ['C' num2str(line_num) ':H' num2str(line_num)];
        ADC_data = hExcel.Activesheet.get ('Range', range);
        ADC_data.value = data;

        range = ['I' num2str(line_num)];
        time_sec = hExcel.Activesheet.get ('Range', range);
```

```

        time_sec.value = date_time_now (length(date_time_now)-
1:length(date_time_now));

        line_num = line_num + 1;
        pause (1);
    end
    pause (0.1);
end
LineNum = hExcel.ActiveSheet.get ('Range', 'A4');
LineNum.value = line_num;

% close Arduino
fclose(s);
delete(s);

% close xls
invoke(hExcel.ActiveWorkbook, 'Save')
Workbook.Close;
invoke(hExcel, 'Quit');
hExcel.delete;

```

2. Проверьте работоспособность m-модуля сравнивая результат с xls-модулем (Задание 4 и 5) который читает данные платформы Arduino, обрабатывает их и записывает в другой xls файл.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как обеспечить управление объектом из Excel используя порты платформы Arduino?
2. Какое максимальное количество строковых записей можно сделать в Excel файле?
3. Какими средствами можно расширить объем записываемой информации в Excel файлы?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Создание интерактивных объектов и сред на базе платформы Arduino. Часть 1 <http://www.vr-online.ru/blog/sozдание-interaktivnyh-obektov-i-sred-na-baze-platformy-arduino-chast-1-7947>
2. Создание интерактивных объектов и сред на базе платформы Arduino. Часть 2 <http://www.vr-online.ru/blog/sozдание-interaktivnyh-obektov-i-sred-na-baze-platformy-arduino-chast-2-7969>
3. Управление из Excel через COM порт <http://www.vr-online.ru/blog/upravlenie-iz-excel-cherez-com-port-8475>
4. Dr. Bob Davidov. COM/DCOM клиент и сервер автоматизации Microsoft ActiveX control. <http://portalnp.ru/2013/10/1156>
5. Parameters of the exhaled breath. <http://www.sciencebits.com/exhalecondense>