

Miroslav P. Dimitrov, Stamen I. Antonov,

PLOTTING DHT11 SENSOR READINGS ON MATLAB

Miroslav P. Dimitrov*, Stamen I. Antonov**

* *National Military University "Vasil Levski", Artillery, AAD and CIS Faculty, amirodimitrov@abv.bg;*

** *National Military University "Vasil Levski", Artillery, stamantonov@abv.bg.*

Abstract: Graphical representation is always helpful to visualize the data and it becomes very easy to find trends and patterns by looking at them. There are many software available to plot graphs based on the input values, but in embedded system MATLAB is one of the most popular software which can not only present the results in graphical format but also can be easily integrated with hardware and microcontrollers..

Keywords: MATLAB, Arduino UNO, Graphics, DHT11 Humidity sensor, temperature sensor.

ГРАФИЧНО ПРЕДСТАВЯНЕ НА ДАННИ В MATLAB ПОЛУЧЕНИ ПРИ ЕКСПЕРИМЕНТАЛНОТО ИЗПОЛЗВАНЕ НА МИКРОКОНТРОЛЕР ARDUINO UNO И СЕНЗОР DHT11

Мирослав П. Димитров*, Стамен И. Антонов**

*Национален военен университет „В. Левски“, Факултет „А, ПВО и КИС“
Шумен, България, *gancho.hristev@gmail.com, **akumanov13@gmail.com*

Въведение

Графичното представяне винаги е от особено значение за представянето на информация. С помощта на графичното представяне много лесно могат да се идентифицират промени и изменения единствено от изучаването на информацията от получените графики. Има налични много софтуерни продукти за построяване на графики въз основа на входни стойности. Във вградени системи MATLAB е един от най-популярните софтуерни продукти който могат да представят получените резултати в графичен вид. MATLAB също предоставя възможността получените резултати лесно да бъдат интегрирани с хардуер и микроконтролери.

Използването на платформата Arduino (Genuino) в процеса на обучение на студенти позволява по - лесното свързване на теоретичните концепции с хардуерната платформа, чрез която се илюстрират [1], [2].

В доклада е представен модел за запис и графично представяне на данни в MATLAB, получени от датчик за температура и влажност DHT11 свързан към Arduino Uno. Моделът е предназначен за изучаване на методите за въвеждане и визуализиране на данни в MATLAB, в реално време, с помощта на платформата Arduino.

Необходими компоненти [3]:

1. Arduino UNO;
2. MATLAB (версия R2016b или по нова);
3. Сензор за температура и влажност DHT11.

Технически характеристики на Arduino UNO[1]:

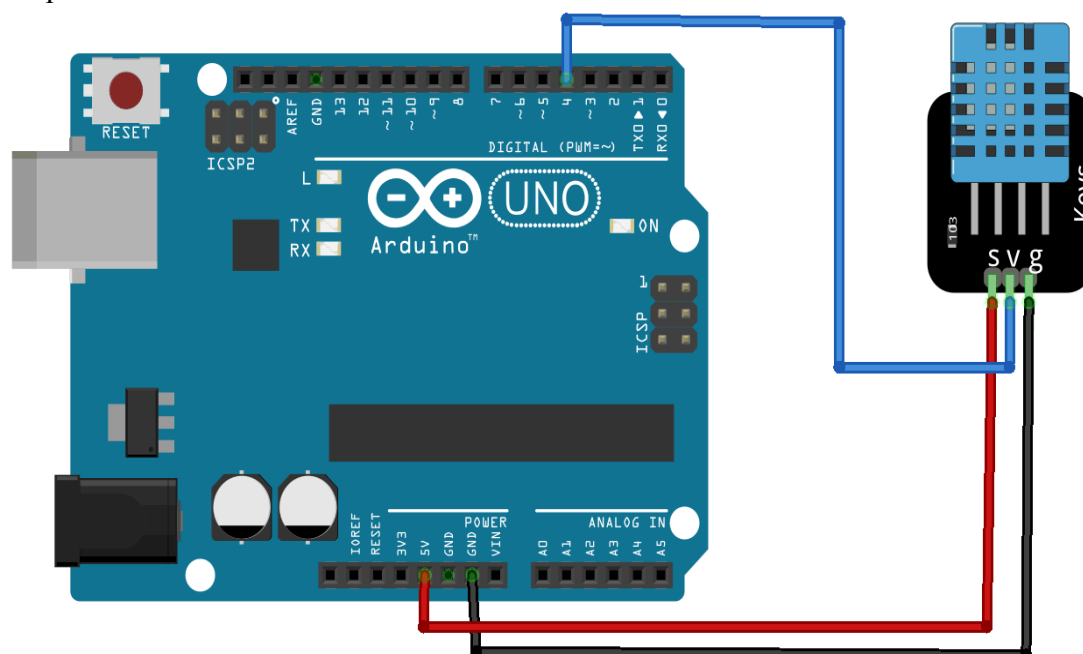
- Микроконтролер базиран на ATmega328P;
- Работно напрежение – 5v;
- Входно напрежение – 7 до 12v;
- Скорост на CLK – 16 Mhz;
- Флаш памет – 32 KB;
- Памет използвана от зареждащия механизъм – 0,5 Kb;
- SRAM – 2 Kb;
- EEPROM – 1 Kb.

Технически характеристики на DHT11 сензор за температура и влажност [2].

- Работен диапазон 20-90% RH 0-50 °C;
- Коефициент на точност при измерването на влажност $\pm 5\%RH$;
- Коефициент на точност при измерването на температура $\pm 2^{\circ}C$;
- Чип – 3 пинов едноредов.

3. Асемблиране на нужните компоненти

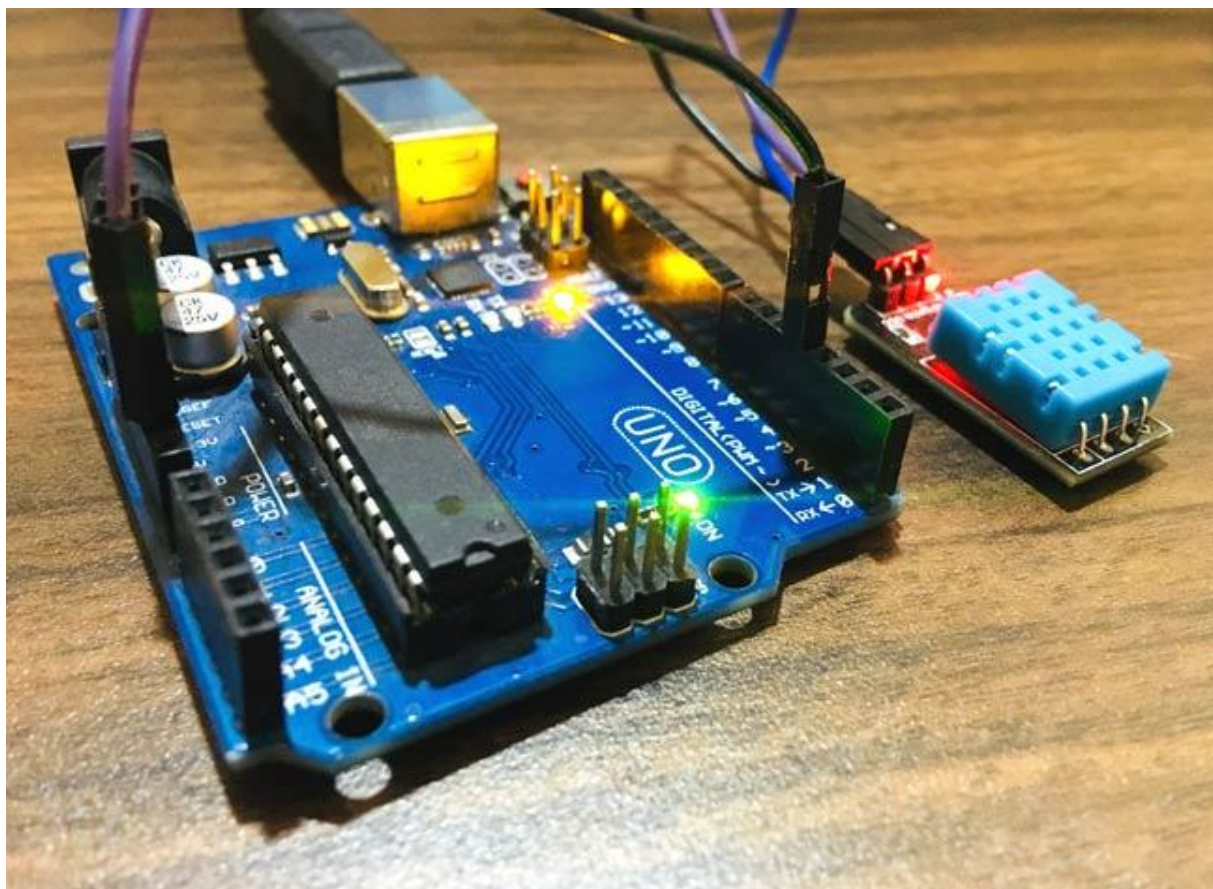
За да бъде изпълнена целта на доклада нужните компоненти за изследването следва да бъдат асемблирани правилно след което да бъдат проверени. Схемата на свързване на различните компоненти е представена на Фиг.1.



Фигура 1: Принципна схема на свързването на DHT11 с Arduino UNO.

Сензорът DHT11 за температура и влажност разполага с 3 пина обозначени с буквени знаци S, V и G. Пин S е свързан с 5-волтовия захранващ изход на Arduino UNO, като по този начин захранва сензора с 5 волта напрежение. Пин V е свързан с дигитален пин номер 4 на Arduino

UNO. Пин G е свързан към заземяващ пин на Arduino UNO. След като всички компоненти бъдат свързани според Фиг.1 проектът има следния вид Фиг.2;



Фигура 2: Физческо свързване на нужните компоненти.

4. Програмиране на софтуерният код на DHT11 взаимодействащ с MATLAB

След като всички компоненти са свързани правилно, трябва да бъде програмиран и зареден програмен код в Arduino посредством Arduino IDE. Пълният код за регистриране и визуализиране на информацията за температурата и влажността в MATLAB е представена в края на доклада. Програмирането на желания софтуерен код за нуждите на доклада преминава през няколко ключови етапа които са описани както следва:

1. Добавяне на нужната библиотека за DHT11;

```
#include <DHT.h>
```

2. Дефиниране на пина за трансфер на данни който е свързан към Arduino (пин номер 4);

```
#define DHTPIN 4
```

3. В 'void setup' се инициализира серийният и DHT11 сензор;

```
void setup() {
```

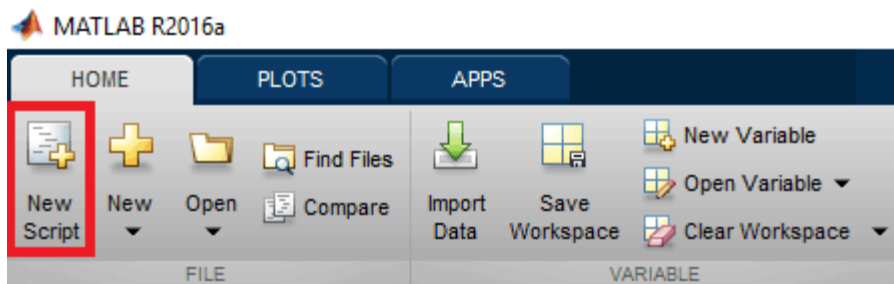
```
Serial.begin(9600);  
delay(2000);  
dht.begin(); // initialise DHT11 sensor  
}
```

4. В ‘void loop’ функцията се задават типа данни за стойностите на температурата и влажността тип *float*. Като за прочит на получените данни се използва командата *float temp = dht.readTemperature()* и *float humi = dht.readHumidity()*. След което за да могат тези данни да бъдат показани на графичен дисплей като по този начин MATLAB ще прочете дадените стойности серийно;

```
void loop() {  
    float temp = dht.readTemperature(); //данни за температурата  
    float humi = dht.readHumidity(); // прочит на данните за температурата  
    Serial.print(temp);  
    Serial.print(humi);  
    delay(2000);  
}
```

5. Код в MATLAB за графично изобразяване и запаметяване на данни;

Програмирането се извършва в софтуерната среда на MATLAB. Писането на кода се извършва в менюто „editor window“ показано на фиг.3:



Фигура 3: Бутон “New script” отваря полето за писане на нужният програмен код.

Програмният код започва с дефинирането на променлива за серийна комуникация между MATLAB и Arduino. COM18 е портът през който микроконтролера е свързан със MATLAB но той може да бъде променен при нужда;

```
s = serial('COM18');// код дефиниращ използването на COM18.
```

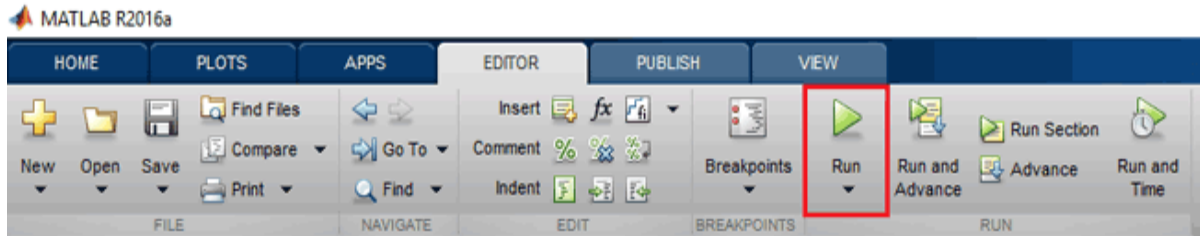
Изложението по долу код функцията „*fopen()*“ е използвана за серийна комуникация между Arduino и MATLAB. След което серийната информация е запаметена в променлива наречена „*out*“.

Променливата *out* е стринг от девет цифри където първите четири цифри съдържат данните за температурата а останалите данните за влажността. Следователно при използването на *Temp(i)=str2num(out(1:4))* and *Humi(i)=str2num(out(5:9))* съответно софтуерът прочита данните за температура и влажност;

```
fopen(s)  
out = fscanf(s)  
Temp(i)=str2num(out(1:4));
```

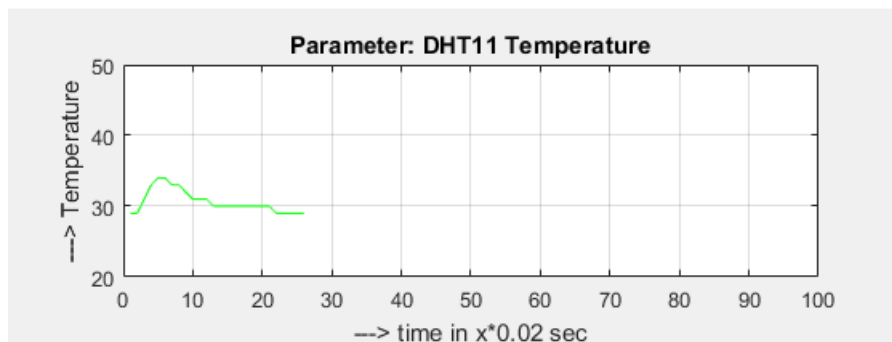
```
Humi(i)=str2num(out(5:9));
```

След като пълният код е зареден в MATLAB след натискането на бутон “Run” кодът е компилиран фиг.4;

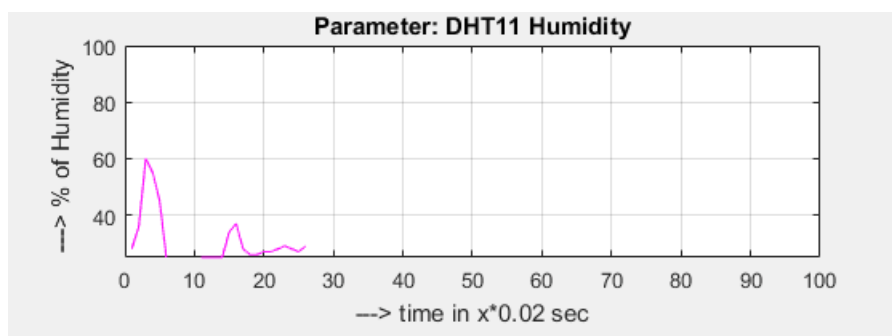


Фигура 4: Бутон “Run” за компилиране и зареждане на програмния код.

Ако кодът е компилиран и изпълнен без грешки, графика в реално време ще бъде показана на екрана. Информацията на тази графика ще бъде обновявана на всеки две секунди. Забавянето от две секунди в продиктувано от факта че времето което е нужно на DHT11 сензора да изпрати нужната информация до серийния порт фиг.5 и фиг.6;

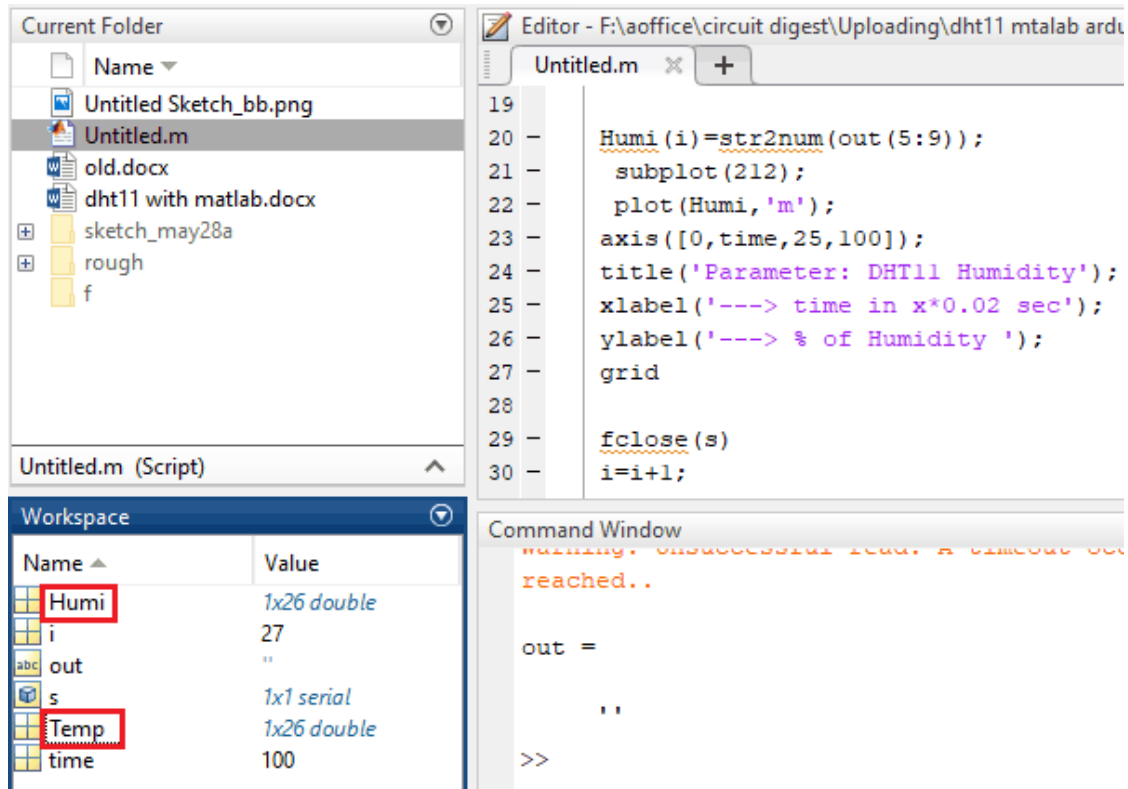


Фигура 5: Диаграма показваща температурните стойности в реално време.



Фигура 6: Диаграма показваща стойностите на влажност в реално време.

За да може запазената информация в реално време за температура и влажност да бъде проверена трябва да бъде отворен работният прозорец показан на фиг.7;



Фигура 7: Диалогов прозорец за запис на получените данни.

6. Пълнен обем на използваният програмният код;

Пълният обем на използваният програмен код на Arduino и MATLAB:

- Програмен код на Arduino:

```
#include <DHT.h>  
  
#define DHTPIN 4                                     #define DHTTYPE DHT11  
  
DHT dht(DHTPIN, DHTTYPE);  
  
void setup() {  
  Serial.begin(9600);  
  delay(2000);  
  dht.begin(); // initialise DHT11 sensor  
}  
  
void loop() {  
  float temp = dht.readTemperature(); //read temperature data  
  float humi = dht.readHumidity(); //read temperature data  
  Serial.print(temp);  
  Serial.print(humi);  
  delay(2000);  
}
```

- Програмен код на MATLAB:

```
s = serial('COM18');
time=100;
i=1;
while(i<time)

fopen(s)
fprintf(s, 'Your serial data goes here')
out = fscanf(s)

Temp(i)=str2num(out(1:4));
subplot(211);
plot(Temp,'g');
axis([0,time,20,50]);
title('Parameter: DHT11 Temperature');
xlabel('---> time in x*0.02 sec');
ylabel('---> Temperature');
grid

Humi(i)=str2num(out(5:9));
subplot(212);
plot(Humi,'m');
axis([0,time,25,100]);
title('Parameter: DHT11 Humidity');
xlabel('---> time in x*0.02 sec');
ylabel('---> % of Humidity ');
grid

fclose(s)
i=i+1;
drawnow;
end
delete(s)
clear s
```

Заклучение

Микроконтролерите започват да заемат неизменна част от нашия всекидневен живот. Изложеният в доклада метод за графично представяне на получена информация за температура и влажност с помощта на микроконтролер Arduino и софтуерната платформа MATLAB е изключително ефективен. Използването на таблици за представянето на информация в реално време предоставя възможността за улеснен анализ на информацията. Използваният програмен код е достатъчен за изпълнението на поставените задачи на доклада. Допълнително изследване може да бъде направено при използването на допълнителни сензори и компоненти за извличане на широк спектър от информация.

References

1. Чобанов Д. Г., “Using geniuno zero arm based platform in computer arcitecture courses” SocioBrains, International Scientific Refereed Online Journal With Impact Factor, Issue 59, July 2019, pp. 114 – 117, ISSN 2367-5721
2. Чобанов Д. Г., “Using Arduino based experiments in computer programming education”, International Scientific Refereed Online Journal With Impact Factor, Issue 60, August 2019, pp. 61 – 63, ISSN 2367-5721
3. Khatri P., Jun 18, 2019, “MATLAB Data Logging, Analysis and Visualization: Plotting DHT11 Sensor readings on MATLAB”, Retrieved from: <https://www.instructables.com/id/Charlieplexing-the-Arduino>