



# ПРОЈЕКТОВАЊЕ ЕЛЕКТРОНСКИ СИСТЕМА

## НАЗИВ ПРОЈЕКТА:

Мерење температуре и влажности ваздуха и слање података на *WolkAbout* интернет платформу

## ТЕКСТ ЗАДАТКА:

Путем *Raspberry Pi3* развојног окружења и сензора температуре и влажности ваздуха извршити мерења. Измерене величине приказати на *WolkAbout* интернет платформи.

## МЕНТОРИ ПРОЈЕКТА:

Др Владимир Рајс

## ПРОЈЕКАТ ИЗРАДИО:

Предраг Вуковић, Е1 33/2018

## ДАТУМ ОДБРАНЕ ПРОЈЕКТА:

25.04.2020.

# Мерење температуре и влажности ваздуха и слање података на *WolkAbout* интернет платформу

Увод .....	2
Реализација .....	2
1. Архитектура хардвера .....	2
2. Архитектура софтера.....	4
Закључак.....	6
Додатак.....	7
Литература.....	11

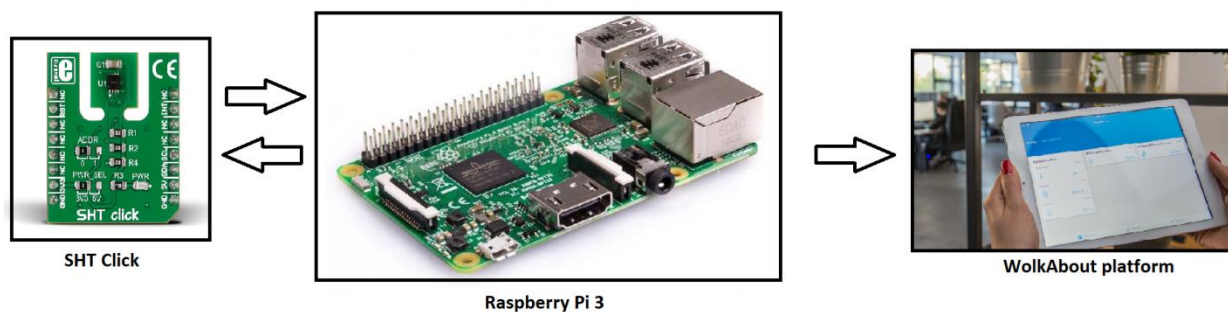
## Увод

Уређај за мерење температуре и влажности ваздуха, те постављање измерених величина на интернет платформу реализован је путем *Raspberry Pi3* развојног окружења, *SHT Click* сензора температуре и влажности ваздуха и *WolkAbout* интернет платформе на којој се приказују мерни резултати. Мерење задатих величина и ажурирање резултата се константно обавља док је уређај активан, односно док је покренута апликација на *Raspberry Pi3* рачунару и док је омогућен приступ интернет платформи.

## Реализација

### 1. Архитектура хардвера

Развојно окружење *Raspberry Pi3* је главни део уређаја. *Raspberry Pi3* представља мали рачунар са скромним перформансама на коме се налази Linux оперативни систем. Као такав, погодан је за различите пројекте писане у C или Python програмском језику. Сензор температуре и влажности ваздуха је *SHT3*, који се, са својим пропратним колом, налази на плочици *SHT Click*. Комплетан сензор је малих димензија, а производ је фирме Микроелектроника. Измерени подаци се могу пратити на *WolkAbout* интернет платформи, путем мобилне апликације или интернет претраживача.



Слика 1. Архитектура хардвера

Главне компоненте кориштене за реализацију:

- **Raspberry Pi3 model B** – трећа генерација рачунара *Raspberry Pi*, која је заменила *Raspberry Pi2* у фебруару 2016. Године. Покреће га 64 битни *Broadcom* процесор са четири језгра и тактом од 1,2 GHz. Поседује 1GB радне меморије, *wireless LAN* и *Bluetooth* модуле, четири USB прикључка, као и HDMI прикључак. Поседује 40 улазно/излазних пинова, преко којих подржава различите комуникационе протоколе као што су I<sup>2</sup>C, SPI, UART, итд.
- **SHT Click** – Микроелектроничка плочица која на себи има дигитални сензор температуре и влажности ваздуха SHT-3x DIS. Комуникација између *Raspberry Pi*-а и *SHT Click*-а базира се на I<sup>2</sup>C (*Inter-Integrated Circuit*) комуникационом протоколу, путем SDA (*Serial Data Line*) и SCL (*Serial Clock Line*) пинова. Путем џампера постоји могућност избора адресе сензора. Ако је пин ADDR везан на масу, као што је у овом случају, адреса у хексадецималној интерпретацији износи 0x44. Ако је пин ADDR везан на Vcc, адреса је 0x45. Преко другог џампера је могућ избор да ли ће напајање плочице бити 3,3V или 5V. Приликом ове употребе, сензор се напаја преко *Raspberry Pi*-а са 3,3V. Сам сензор SHT-3x DIS је потпуно калибрисан, линеаризован и поседује температурно компензован дигитални излаз. Опсег мерења температуре ваздуха је од -40°C до 125°C са резолуцијом од 0,015°C те од 0% до 100% са резолуцијом од 0,01% када је у питању влажност ваздуха.
- **WolkAbout IoT** платформа – интернет платформа, за чије коришћење је неопходан мобилни телефон или рачунар, осмишљена да помогне у развоју веб и мобилних апликација које могу да контролишу и управљају различитим уређајима. Постоји могућност интегрисања са другим системима, што омогућава изградњу комплетног пословног окружења додавањем компоненти специфичних за индустрију и различите апликације. Користећи кључне модуле, лако је повезати се на платформу, надгледати и контролисати различите уређаје и апликације.

## 2. Архитектура софтера

Код за ову апликацију написан је програмском језику *Python*.

Захваљујући готовим функцијама, које се налазе у одговарајућим *Python* библиотекама, сама апликација није комплексна.

Приликом иницијализације врши се:

1. Дефинисање I<sup>2</sup>C порта, на који је прикључен сензор температуре и влажности ваздуха. У овом случају, кориштен је I<sup>2</sup>C1 модул на пиновима 3 (*SDA*) и 5 (*SCL*). Дефинисање I<sup>2</sup>C порта се извршава функцијом **smbus.SMBus(1/0)** – 1 или 0, зависно од порта.
2. Дефинисање уређаја који приступа *WolkAbout* интернет платформи, на коју ће слати измерене податке. Функција **wolk.Device(key, password)** дефинише уређај који жели да приступи интернет платформи. Параметри *key* и *password* су предефинисани од стране платформе за појединачне уређаје.
3. Само повезивање дефинисаног уређаја се обавља извршењем функције **wolk.WolkConnect(device)**.

У бесконачној петљи се извршава мерење и читање вредности са сензора температуре и влажности ваздуха, те ажурирање датих вредности на интернет платформи. Овај процес се понавља сваких 60 секунди, из разлога што температура и вчажност ваздуха нису брзо променљиве величине.

У бесконачној петљи се извршавају следеће функције:

1. **write\_i2c\_block\_data(ADDRESS, 0x24, 0x0B)** – ова функција уписује на I<sup>2</sup>C магистралу података адресу сензора коме се упућује команда, те двобајтну команду којом се окида мерење температуре и влажности ваздуха. Команда 0x24 означава мерење без истезања сигнала на *SCL* линији, док 0x0B команда означава поновљивост мерења. Након издавања команде за мерење, сензору је потребно око 20ms како би обавио мерења.
2. **read\_i2c\_block\_data(ADDRESS, 0x00, 6)** – ова функција уписује на I<sup>2</sup>C магистралу података адресу сензора коме се упућује команда за испис измерене вредности температуре и влажности ваздуха. Команда 0x00 означава читање вредности, док параметар 6 означа број бајтова података које очекујемо да сензор врати. Тих шест бајтова означавају вредност температуре, двобајтна вредност праћена контролним CRC (*Checksum*)

бајтом, те двобајтна вредност влажности ваздуха, такође праћена контролним CRC бајтом. Коначне вредности температуре и влажности ваздуха се рачунају по формулама датим од стране произвођача сензора.

3. **wolk\_device.add\_sensor\_reading**("Temperature", temperature) – ова функција врши додавање измерене вредности са сензора. Први параметар функције представља назив референце која је дефинисана на интернет платформи, док други параметар је измерена вредност која се уписује на платформу.
4. **wolk\_device.publish()** – након додавања измерених вредности са сензора, ова функција ажурира додате вредности на интернет платформу.

Алгоритам рада апликације налази се на слици 2. Комплетан Python код се налази у додатку овог рада, на страни 7.



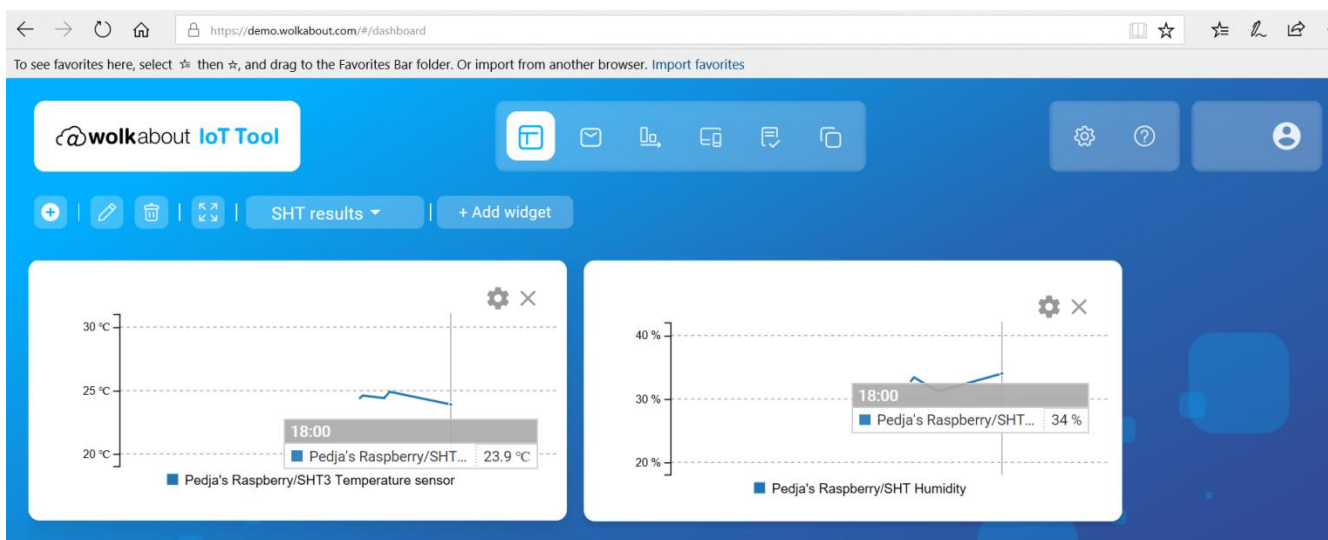
Слика 2. Алгоритам рада програма

## Закључак

Апликација поуздано мери вредности температуре и влажности ваздуха, те измерене вредности ажурира на *WolkAbout* интернет платформу, где је лако пратити поменуте параметре. Пример изгледа података на *WolkAbout* платформи се налази на слици 3.

Python програмски језик се показао као приступачно, једноставно и јефтино решење за ову примену. Готове програмске функције су корисне за једноставе апликације, док би за комплексније примене затвореност функција могла представљати проблем.

Могућа побољшања могу бити постигнута у виду пролонгирања времена потребног за један циклус мерења, јер температура и влажност ваздуха нису величине које се брзо мењају. С обзиром на то, време мерења и ажурирања података се може повећати на сваких 10 минута. Такође, једно од побољшања би могло бити додавање још сензора, или неки актуатор којим би се управљало путем мобилне апликације.



Слика 3. Пример ажурирања података

## Додатак

У додатку се налази комплетак Python код са коментарима.

```
# Project: cloud_project
# Author: Predrag Vukovic
# Date: 25.04.2020
# Revision: 1.0

# Read temperature and humidity value from SHT3 sensor.
# These values are measured every 60s and published to WolkAbout IoT
platform.

#import neccessary libraries
import os
import sys
import smbus
import time
import datetime
from io import StringIO
import wolk

#define I2C bus (1 or 0, depending where is connected sensor on Raspberry Pi)
and I2C adreses of sensor
BUS = 1
ADDRESS = 0x44

#define contants used to calculate temperature and humidity
C1 = -45.0
```



```

C2 = 175.0
C3 = 65535.0

#sensor connected on port I2C1
bus = smbus.SMBus(BUS)

# define measurement period in seconds
MEASUREMENT_PERIOD = 60

def main():

    # Define device which will be connected to colud
    # key and password defined by WolkAbout platform
    device = wolk.Device(key="1bz62mgd1dxqhn4n", password="c660584f-cac4-
41bf-9240-5c6907d30ba5")

    # Assign our deviced to connecting to WolkAbout platform
    wolk_device = wolk.WolkConnect(device)

    # Start program
    print("Connecting to WolkAbout IoT Platform")

    # Connect to device to WolkAbout Platform, in case that is not possible, exit
system
    try:
        wolk_device.connect()
    except RuntimeError as e:
        print(str(e))
        sys.exit(1)

```

```

while True:
    try:
        # write i2c address of SHT sensor, and commands (0x24 - disabled clock
        stretching, 0x0B - Medium repeatability)
        # to trigger temperature and humidity measurement
        bus.write_i2c_block_data(ADDRESS,0x24,[0x0B])

        # sensor needs 20ms to measure values
        time.sleep(0.2)

        # trigger reading data from sensor, we expect 6 data bytes
        data = bus.read_i2c_block_data(ADDRESS, 0x00, 6)

        # store data bytes from sensor
        temp_H = data[0]    # temperature MSB
        temp_L = data[1]    # temperature LSB
        HUMI_H = data[3]    # humidity MSB
        HUMI_L = data[4]    # humidity LSB

        # calculate measured temperature value and round float value to 2
decimal positions
        temp = (temp_H << 8) + temp_L
        temperature = C1 + C2*(temp / C3)
        temperature = float("{:.2f}".format(temperature))

        # calculate measured humidity value and round float value to 2 decimal
positions
        humi = (HUMI_H << 8) + HUMI_L

```

```

humidity = 100.0 * humi / C3
humidity = float("{:.2f}".format(humidity))

# upload measured values to WolkAbout platform
wolk_device.add_sensor_reading("Temperature", temperature)
wolk_device.add_sensor_reading("Humidity", humidity)
wolk_device.publish()

# show measured values
print("Temperature: ", temperature)
print("Humidity:", humidity)

# wait until next cycle of measurement
time.sleep(MEASUREMENT_PERIOD)

# if program execution is interrupted, then disconnect device from
WolkAbout platform
except KeyboardInterrupt:
    print("Received KeyboardInterrupt, quitting")
    wolk_device.disconnect()
    sys.exit()

if __name__ == "__main__":
    main()

```

## Литература

1. Raspberry Pi3, Documentation

<https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

2. SHT Click, Data Sheet

[https://download.mikroe.com/documents/datasheets/Sensirion\\_Humidity\\_SHT3x\\_Datasheet\\_digital.pdf](https://download.mikroe.com/documents/datasheets/Sensirion_Humidity_SHT3x_Datasheet_digital.pdf)

3. WolkAbout, Documentation

<https://wolkabout.com/>