Advanced Research in Electrical and Electronic Engineering
p-ISSN: 2349-5804; e-ISSN: 2349-5812 Volume 4, Issue 3 April-June, 2017, pp. 113-117
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/Publication.html

Real Time Data Acquisition and Encrypted Data Transmission using Microcontroller

Shri Swagato Guin[1], Shri Partha Banerjee[2] and Shri Pickon Majumdar[3]

[1,2]National Institute of Electronics and Information Technology, Kolkata
[3]School of Illumination Science, Technology and Design, Jadavpur University
E-mail: [1]swagatoguin@yahoo.com, [2]parthadataexe01@googlemail.com, [3]pickonmajumdar0635@gmail.com

Abstract—*Real time data acquisition and encrypted data transmission using low-cost Microcontroller Devices like Arduino UNO has become very popular and effective in recent past. The real time data does gather may be readily processed using low cost handheld devices to arrive at important decisions in various fields e.g. Telemedicine, Industrial Process Control , Environmental Studies to name a few. Whereas such low cost solution may be effectively put to use in many important areas as noted above, the issue of data security during transmission is of paramount importance. If such sensitive data is intercepted and modified during transmission the result may be catastrophic.*

Any monitoring system is an automated system, designed to monitor different environmental parameters i.e. temperature, humidity, intensity of light, intensity of carcinogenic particles in air etc. in different locations at different point of time and continuously record them properly for future analysis. Monitoring temperature and humidity is gaining importance in different quality applications. Any deterioration will create hazard and will give an early warning in ensuing problems in proper planning and rescheduling of applications.

The present paper aims at integration of a suitable 256 bits Rijndael data encryption algorithm at the microcontroller level with a suitable decryption algorithm to be made available at receiver's end. The system will monitor temperature and humidity at any time. It requires a digital composite sensor for measuring temperature and relative humidity. Also Arduino Microcontroller is used for performing necessary computations and recording temperature and humidity from the sensor and record inside relational database system in encrypted fashion. This real data can be viewed through web as well as graphically using MATLAB for better monitoring.

Keywords*: Microcontroller, Arduino UNO, Rijndael Encryption and Decryption, MATLAB*

1. INTRODUCTION

The major industries in India involving agricultural, pharmaceutical, industrial process control forming the backbone of countries economy. The continuous monitoring of temperature and humidity is a major criteria in all aforesaid industries. Any kind of deviation in the environmental conditions or the preset parameters can cost heavy financial losses due to alterations in productivity. A precise monitoring of humidity and temperature is required.

In any developing country like India, Sensors are becoming an integral part in the society. Nowadays sensors are used mostly for monitoring applications and also measuring real data which can be analyzed properly in near future.

The paper presents a system on temperature-cum-humidity monitoring using Arduino Microcontroller, DHT 11 sensor and PC. This system uses Rijndael data encryption algorithm at Microcontroller level and suitable decryption algorithm at the receiving end for measuring the details of current temperature and humidity and also monitor them graphically using MATLAB and also through web.

2. METHODOLOGY ADOPTED

The system monitors temperature and humidity at any time. A work plan is carried out to develop the system using Arduino Microcontroller with DHT 11 sensor. Arduino IDE (Integrated Development Environment) is used to perform the necessary coding for Arduino Microcontroller to perform the necessary computations on the data real data available from the sensor. Arduino IDE is an open source software for easily writing necessary coding to upload it inside microcontroller. The program uploaded in Microcontroller will convert the analog output of temperature and humidity into digital form and it can be viewed either using serial monitor or serial plotter of Arduino IDE or using GUI program written in MATLAB or through web with a specific URL. A script will transform the real data (temperature and humidity) into an encrypted data and store them in relational database.

3. ARCHITECTURE

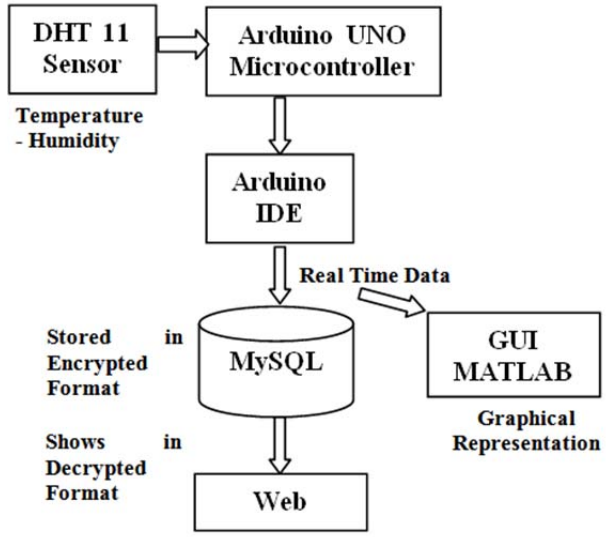


Fig. 1: Microcontroller Sensor Architecture

4. PROCESS DESCRIPTION

I. Proper Connectivity of DHT 11 Sensor with Arduino Microcontroller

II. Capturing real temperature and humidity data through serial monitor of Arduino IDE

III. Conversion of real data into csv format

IV. View real data graphically through MATLAB

V. Storing encrypted data in MySQL Database

VI. View decrypted data through Web in real time for future analysis

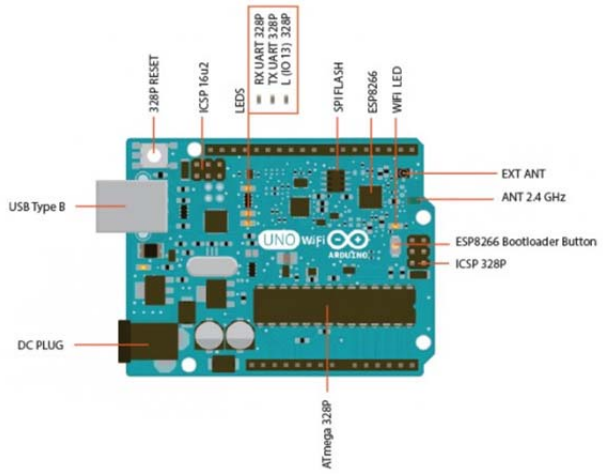


Fig. 2: Arduino UNO

5. DEVICE DESCRIPTION

I. Arduino UNO Microcontroller

Arduino is an open-source platform used for building digital devices and interactive objects that can sense and control physical devices. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on PC / Laptop, used to write and upload computer code to the physical board.

Arduino UNO is a microcontroller board, based on ATMega328. It consists of 14 digital I/O pins in which Pulse Width Modulation (PWM) pin count is 6. For analog inputs there are 6 analog pins. Also it contains 16 MHz quartz crystal, a USB connection, a power jack, and a reset button.

II. DHT 11 Sensor

The DHT11 Temperature & Humidity Sensor consists of a temperature & humidity sensor which is calibrated against a digital signal output. The DHT11 ensures reliability, high efficiency and stability for a long time which is present with the help of this digital-signal-acquisition exclusive technique. This temperature and humidity sensor have an Negative Temperature Co-efficient (**NTC**) temperature component for measuring the temperature and a very high-performance 8-bit microcontroller connected for humidity, which is cost effective and provides an excellent quality and fast response ability with anti-interference.

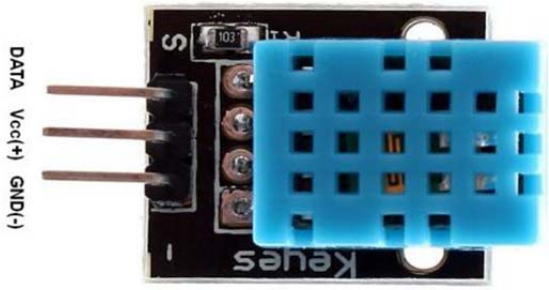


Fig. 3: DHT 11 Sensor

Table 1: Sensor Details

Item	DHT11
Measurement Range	20-90%RH 0-50 ℃
Humidity Accuracy	±5%RH
Temperature Accuracy	±2℃
Resolution	1
Package	3 Pin Single Row

6. SOURCE CODE

```
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>
#define DHTPIN 12 // Pin which is connected to the DHT sensor.
uint32_t delayMS;
void setup() {
 Serial.begin(9600);
 dht.begin();
 sensor_t sensor;
 …
 }
void loop() {
 sensors_event_t event;
…..
 Serial.print("Temperature: ");
 Serial.print(event.temperature);
 Serial.println(" degree C");
 }
.….
{
 Serial.println("Error reading humidity!");
 }
 else {
 Serial.print("Humidity: ");
 Serial.println(event.relative_humidity);
 Serial.println("%");
}}
```

7. ENCRYPTION AND DECRYPTION TECHNIQUES

Rijndael Algorithm is an Advanced Encryption Standard (AES). It's a block cipher which works iteratively.

- Block size: 128 bit (but also 192 or 256 bit)
- Key length: 128, 192, or 256 bit
- Number of rounds: 10, 12 o 14
- Key scheduling: 44, 52 or 60 subkeys having length = 32 bit

Each round (except the last one) is a uniform and parallel composition of 4 steps:

- SubBytes (byte-by-byte substitution using an S-box)
- ShiftRows (a permutation, which cyclically shifts the last three rows in the State)
- MixColumns (substitution that uses Galois Fields, corps de Galois, GF(28) arithmetic)
- AddRound key (bit-by- bit XOR with an expanded key)

I. Rijndael Design

- Operations performed on State (4 rows of bytes)
- The 128 bit key is expanded as an array of 44 entries of 32 bits word; 4 distinct words serve as a round key for each round; key schedule relies on the S-box
- Algorithms composed of three layers:

  Linear Diffusion

  Non-Linear Diffusion

  Key Mixing

II. Rijndael: High-Level Description

State = X

1. AddRoundKey(State, Key_0)
2. for r=1 to (Nr – 1)
   a. SubBytes(States, S-box)
   b. ShiftRows(State)
   c. MixColums(State)
   d. AddRoundKey(State, Key_r)

end for

1. SubBytes(State, S-box)
2. ShiftRows(State)
3. AddRoundKey(State, Key_{Nr})

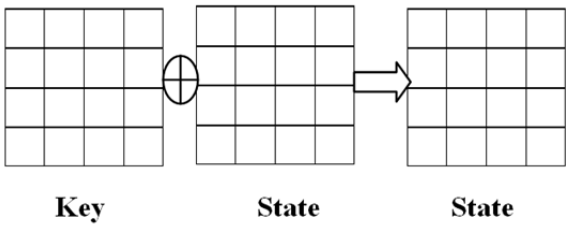
Y=State

A. AddRound Key

- **State** is represented as follows (16 bytes)

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,2}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

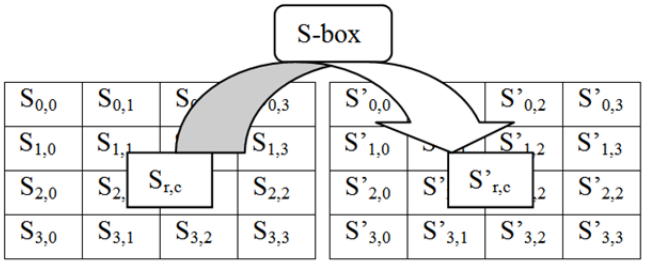
- AddRoundKey(State, Key)



B. SubBytes Transformation

Bytes are transformed using a non-linear S-box

$$S'_{r,c} \leftarrow \text{S-box}(S_{r,c})$$



C. Rijndael S-box Table

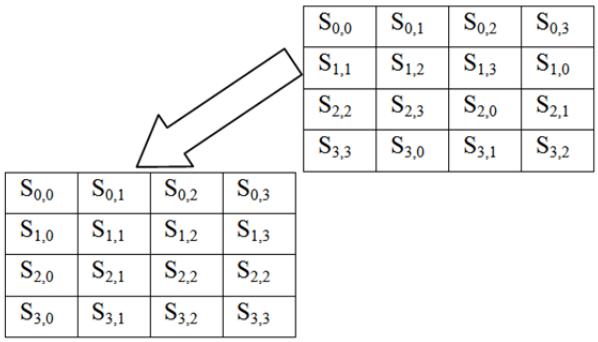
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	3	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Example: hexa 53 is replaced with hexa ED

(The first 4 bits in the byte(the first hexadecimal value, hence) individuate the row, the last 4 bits individuate the column)

D. Shift Rows

Circular Left Shift of a number of bytes equal to the row number



E. MixColumns

- Interpret each column as a vector of length
- Each column of State is replaced by another column obtained by multiplying that column with a matrix in a particular field (Galois Field).

III. Decryption

- The decryption algorithm is not identical with the encryption algorithm, but uses the same key schedule.
- There is also a way of implementing the decryption with an algorithm that is equivalent to the encryption algorithm (each operation replaced with its inverse), however, in this case, the key schedule must be changed.

IV. Code Fragment

A. Encryption

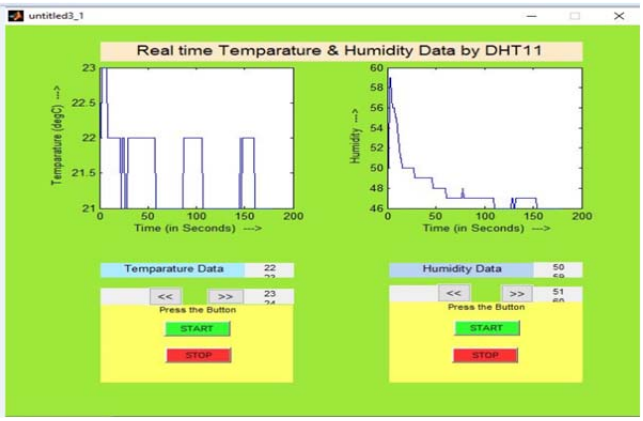
```
$key=md5('$sensor');
function encrypt($string, $key)
        {
$string=rtrim(base64_encode(mcrypt_encrypt(MCRYPT_
RIJNDAEL_256, $key, $string,
MCRYPT_MODE_ECB)));
                return $string;
        }
```

B. Decryption

```
$key=md5('$sensor');
function decrypt($string, $key)
        {
$string=rtrim(mcrypt_decrypt(MCRYPT_RIJNDAEL_25
6, $key, base64_decode($string),
MCRYPT_MODE_ECB));
                return $string;}
```

8. RESULTS

The measurement of humidity and temperature were carried out during remote area Canning (22.3104 Degree North, 88.6579 Degree East), South 24 PGS, West Bengal on 11th March 2017 respectively during day time. The results are shown in graphically:



Location: remote area Canning (22.3104 Degree North, 88.6579 Degree East), South 24 PGS, West Bengal

Fig. 4: Plot of Temperature and Humidity through MATLAB

Fig. 5: Serial Monitor view Temperature and Humidity through ARDUINO IDE

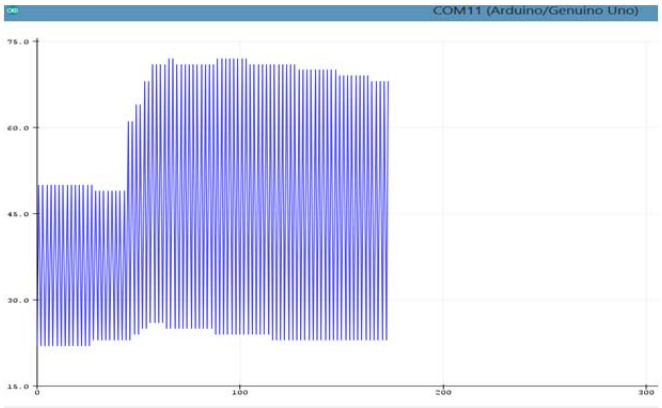


Fig. 6: Serial Plotter view Temperature and Humidity through ARDUINO IDE

Fig. 7: Web based encrypted and decrypted view of Temperature and Humidity

9. CONCLUSION

Temperature and Humidity monitoring system monitors temperature and humidity in efficient manner without involvement of any complex tools and technologies. Using this approach temperature as well as humidity both can be monitored in web as well as graphically. This real time data can act as an input in complex data analysis process. Also this data can be used in Knowledge Based System in areas like Agriculture, Telemedicine, and Power Plant etc.

REFERENCES

[1] Y.-m. HAN and J.-p. ZHAO, "Design of temperature humidity wireless sensor network node based on DHT11," Journal of Jinggangshan University (Natural Science), vol. 32, pp. 67-70, 2011.

[2] N. H. A. Aziz, W. N. W. Muhamad, N. A. Wahab, A. J. Alias, A. T. Hashim, and R. Mustafa, "Real time monitoring critical parameters in tissue culture growth room with SMS alert system," in Intelligent Systems, Modelling and Simulation (ISMS), 2010 International Conference on, 2010, pp. 339-343.

[3] M. Margolis, Arduino cookbook: "O'Reilly Media, Inc.", 2011.

[4] Temperature and humidity monitoring systems for fixed storage areas WHO Technical Report Series, No. 961, 2011 August 2014.

[5] D. Yu-fang, "nRF905 & DHT11 based wireless temperature & humidity logger," Information Technology, vol. 8, p. 057, 2012.

[6] Buenfeld N, Davis R,Karmini A,Gilbertson A. Intelligent monitoring of concrete structures. 666th ed. UK: CIRIA; 2008. p. 150.

[7] N. Tianlong, "Application of Single Bus Sensor DHT11 in Temperature Humidity Measure and Control System [J]," Microcontrollers & Embedded Systems, vol. 6, p. 026, 2010.

[8] Daeman Joan, Rijmen Vincent, "The Design of Rijndael: AES - The Advanced Encryption Standard (Information Security and Cryptography)" ISBN-13: 978-3642076466, 1st ed. 2002