ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 3, Issue 8, August 2016

# Wireless Daq System

[1] Mr. Sachin Hosamani, [2] Prof. Laxmikant Bordekar, [3] Prof. Vijaya Hadfadkar
[1][2][3] Department Of Electronics And Communications Engineering,
Agnel Institute Of Technology And Design, Assagao-Goa, India

***Abstract:* -- With the advancement of technology, the processes are becoming more and more complex. Due to this increase in complexity, for efficient analysis of process the number of parameters required for data acquisition also increases. Data Acquisition is simply the gathering of information about a system or process. It is the process of collecting data in an automated fashion from analog sources and process using digital measurement system. Before the computer age, most data was recorded manually or on strip chart recorders. Many new generation data acquisition products have been developed due to emergence of microcontroller that enables real-time gathering, analysis, logging and viewing of data. To meet these requirements the demand of an improved, efficient and up-to-date data acquisition is increasing.**

**Index Terms: — Thermocouple, Arduino, MAX31855, ESP8266**

## I. INTRODUCTION

The world is advancing, technologies are getting upgraded and at the same time the processes are getting more and more complex, and therefore to go about the same efficiently number of parameters required for data acquisition also increase. The practical meaning of acquisition is to acquire, so here data acquisition is just acquiring the data from the physical world, for example temperature, pressure, humidity etc. This was about acquisition, now to execute all these processes manually is a tiring and tedious task. This device will take care of it, as its a wireless device. This feature enables a user to read the data from this device sitting at his home or any workplace. The acquired data from the physical world cannot be read by the hardware devices directly as it is analog in nature, therefore suitable signal conditionings are used to process the analog outputs of the sensors. This data is fed to the micro controller which will give the desired value as output. This device makes use of a wireless transceiver that will transmit/receive data. Here in the partial fulfillment of the project we are dealing with temperature measurement only. The data will be collected in an automated fashion from analog and digital measurement devices such as sensors. This collected data will undergo required conditioning for the next block of devices to read. Thermocouples are used to convert this temperature value into corresponding electrical voltage. To convert the analog voltage into digital data ADCs are used. Furthermore, the Wi-Fi modules take care of the transmission and reception of the data.

## II.HARDWARE IMPLEMENTATION

### *a. Components*

Temperature measurement using microcontroller based data acquisition includes the following components:

1. Thermocouple (k- type)
2. Cold-junction compensated thermocouple-to-digital converter (max31855)
3. Microcontroller (arduino)
4. Liquid crystal display (hd 44780)
5. Relays
6. Wi-fi module (esp 8266)

### *b.Components section and description*

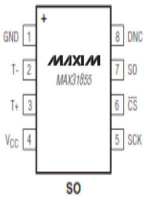
#### *1. Thermocouple*

A thermocouple is a temperature-measuring device consisting of two dissimilar conductors that contact each other at one or more spots, where a temperature differential is experienced by the different conductors (or semiconductors). It produces a voltage when the temperature of one of the spots differs from the reference temperature at other parts of the circuit. Thermocouples are a widely used type of temperature sensor for measurement and control, and can also convert a temperature gradient into electricity. Commercial thermocouples are inexpensive, interchangeable, are supplied with standard connectors, and can measure a wide range of temperatures. In contrast to most other methods of temperature measurement, thermocouples are self-powered and require no external form of excitation. The main limitation with thermocouples is accuracy; system errors of less than one degree Celsius (C) can be difficult to achieve. Any junction of dissimilar metals will produce an electric potential related to temperature.

Thermocouples for practical measurement of temperature are junctions of specific alloys which have a predictable and repeatable relationship between temperature and voltage. Different alloys are used for different temperature ranges. Properties such as resistance to corrosion may also be important when choosing a type of thermocouple. Where the measurement point is far from the measuring instrument, the intermediate connection can be made by extension wires which are less costly than the materials used to make the sensor. Thermocouples are usually standardized against a reference temperature of 0 degrees Celsius; practical instruments use electronic methods of cold-junction compensation to adjust for varying temperature at the instrument terminals. Electronic instruments can also compensate for the varying characteristics of the thermocouple, and so improve the precision and accuracy of measurements.

## 2. *MAX31855*

The MAX31855 performs cold-junction compensation and digitizes the signal from a K-, J-, N-, T-, S-, R-, or E-type thermocouple. The data is output in a signed 14-bit, SPI-compatible read-only format. This converter resolves temperatures to 0.25$^{o}$ C, allows readings as high as +1800$^{o}$ C and as low as -270$^{o}$ C, and exhibits thermocouple accuracy of 2$^{o}$ C for temperatures ranging from -200$^{o}$ C to 700$^{o}$ C or K-type thermocouples.



| PIN | NAME | FUNCTION |
|---|---|---|
| 1 | GND | Ground |
| 2 | T- | Thermocouple Input. See Table 1. Do not connect to GND. |
| 3 | T+ | Thermocouple Input. See Table 1. |
| 4 | $V_{CC}$ | Power-Supply Voltage |
| 5 | SCK | Serial-Clock Input |
| 6 | $\overline{CS}$ | Active-Low Chip Select. Set $\overline{CS}$ low to enable the serial interface. |
| 7 | SO | Serial-Data Output |
| 8 | DNC | Do Not Connect |

***Fig 1: Pin Diagram and Discription***

The device includes signal-conditioning hardware to convert the thermocouple's signal into a voltage compatible with the input channels of the ADC.

$$V_{OUT} = (41.276\mu V/^{o}C) \times (T_R - T_{AMB})$$

The device senses and corrects for the changes in the reference junction temperature with cold-junction compensation. The reference junction, or "cold" end (which should be at the same temperature as the board on which the device is mounted). The device senses and corrects for the changes in the reference junction temperature with cold-junction compensation. It does this by first measuring its internal die temperature, which should be held at the same temperature as the reference junction. It then measures the voltage from the thermocouple's output at the reference junction and converts this to the non-compensated thermocouple temperature value. This value is then added to the device's die temperature to calculate the thermocouple's "hot junction" temperature.

***Steps to get data:***

- Drive CS low
- Apply a clock signal at SCK to read the results at SO.

Drive CS low to output the first bit on the SO ***pin.***

- A complete serial-interface read of the cold-junction compensated thermocouple temperature requires 14 clock cycles.
- 32 clock cycles are required to read both the thermocouple and reference junction temperatures

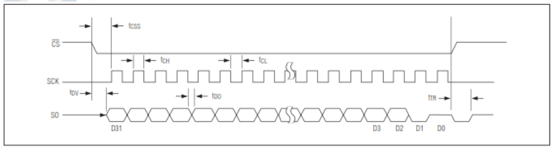
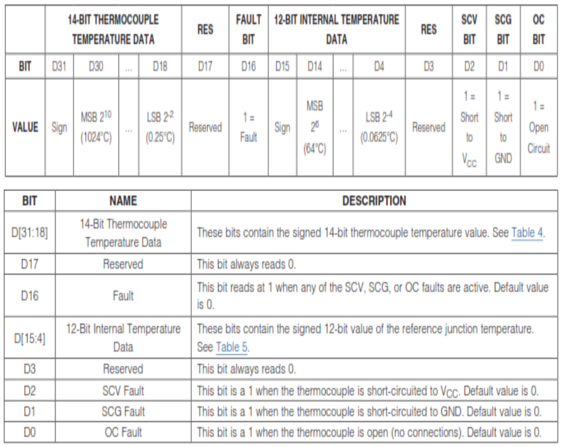


***Fig 2: Timing Diagram***

The first bit, D31, is the thermocouple temperature sign bit, and is presented to the SO pin within tad of the falling edge of CS. Bits D[30:18] contain the converted temperature in the order of MSB to LSB, and are presented to the SO pin with the falling edge of SCK. Bit D16 is normally low and goes high when the thermocouple input is open or shorted to GND or V. The reference junction temperature data begins with D15. CS can be taken high at any point while clocking out conversion data. If T+ and T- are unconnected, the thermocouple temperature sign bit (D31) is 0, and the remainder of the thermocouple temperature value (D[30:18]) is 1.



|  | 14-BIT THERMOCOUPLE TEMPERATURE DATA |  |  |  | RES | FAULT BIT | 12-BIT INTERNAL TEMPERATURE DATA |  |  |  | RES | SCV BIT | SCG BIT | OC BIT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BIT | D31 | D30 | ... | D18 | D17 | D16 | D15 | D14 | ... | D4 | D3 | D2 | D1 | D0 |
| VALUE | Sign | MSB $2^{10}$ (1024°C) | ... | LSB $2^{-2}$ (0.25°C) | Reserved | 1 = Fault | Sign | MSB $2^{6}$ (64°C) | ... | LSB $2^{-4}$ (0.0625°C) | Reserved | 1 = Short to $V_{CC}$ | 1 = Short to GND | 1 = Open Circuit |

| BIT | NAME | DESCRIPTION |
|---|---|---|
| D[31:18] | 14-Bit Thermocouple Temperature Data | These bits contain the signed 14-bit thermocouple temperature value. See Table 4. |
| D17 | Reserved | This bit always reads 0. |
| D16 | Fault | This bit reads at 1 when any of the SCV, SCG, or OC faults are active. Default value is 0. |
| D[15:4] | 12-Bit Internal Temperature Data | These bits contain the signed 12-bit value of the reference junction temperature. See Table 5. |
| D3 | Reserved | This bit always reads 0. |
| D2 | SCV Fault | This bit is a 1 when the thermocouple is short-circuited to $V_{CC}$. Default value is 0. |
| D1 | SCG Fault | This bit is a 1 when the thermocouple is short-circuited to GND. Default value is 0. |
| D0 | OC Fault | This bit is a 1 when the thermocouple is open (no connections). Default value is 0. |

*Fig 3*: **Bit Weight and Function**

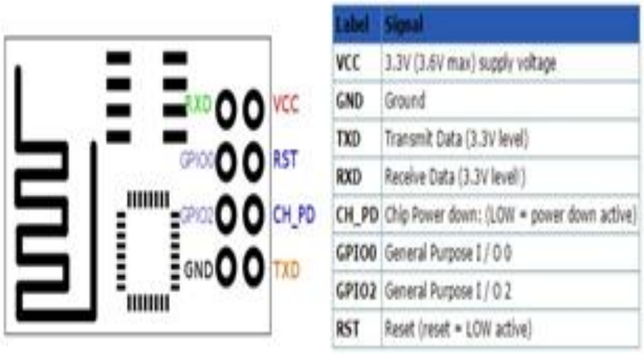
### 3. *ATMEGA 328P*

The Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno can be programmed with the Arduino Software (IDE).

***MEMORY:*** The ATmega328 has 32 KB . It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

***COMMUNICATION:*** The Uno has a number of facilities for communicating with a computer, another Uno board, or other microcontrollers. The ATmega328 provides UAR TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will ash when data is being transmitted via the USB- to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino Software (IDE) includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

### 4. *Wi-Fi Module*



| Label | Signal |
|---|---|
| VCC | 3.3V (3.6V max) supply voltage |
| GND | Ground |
| TXD | Transmit Data (3.3V level) |
| RXD | Receive Data (3.3V level) |
| CH_PD | Chip Power down: (LOW = power down active) |
| GPIO0 | General Purpose I / O 0 |
| GPIO2 | General Purpose I / O 2 |
| RST | Reset (reset = LOW active) |

***Fig 4: ESP8266 Module***

This WiFi serial transceiver module is based on the ESP8266 chipset. The ESP8266 modules offer a very cheap way to expand microcontroller projects with Internet connection. ESP8266 has an Integrated TCP/IP protocol stack. In short, the ESP8266 module is a TTL "Serial to Wireless Internet" device. The module runs at 3.3 Volt and the serial port is set to a data rate of 115200 baud.

Processing and storage capacity on ESP8266 powerful piece, it can be integrated via GPIO ports sensors and other applications specific equipment to achieve the lowest early in the development and operation of at least occupy system resources. The ESP8266 highly integrated chip, including antenna switch balun, power management converter, so with minimal external circuitry, and includes front-end module, including the entire solution designed to minimize the space occupied by PCB. The system is equipped with ESP8266 manifested leading features are: energy saving VoIP quickly switch between the sleep / wake patterns, with low-power operation adaptive radio bias, front-end signal processing functions, troubleshooting and radio systems coexist characteristics eliminate cellular / Bluetooth / DDR / LVDS / LCD interference.

The hardware connections required to connect to the ESP8266 module are fairly straight-forward but there

are a couple of important items to note related to power: The ESP8266 requires 3.3V power–do not power it with 5 volts! The ESP8266 needs to communicate via serial at 3.3V and does not have 5V tolerant inputs, so you need level conversion to communicate with a 5V microcontroller like most Arduinos use.
Characteristics

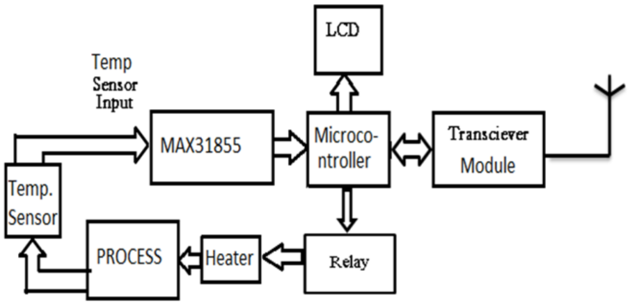
- 802.11 b / g / n
- Wi-Fi Direct (P2P), soft-AP
- Built-in TCP / IP protocol stack
- Built-in PLL, voltage regulator and power management components
- Built-in temperature sensor
- Support antenna diversity
- off leakage current is less than 10uA
- Built-in low-power 32-bit CPU: can double as an application processor
- SDIO 2.0, SPI, UART
- Integrated TR switch, balun, LNA, power amplifier and matching network

***Hardware connection***

- The ESP8266 requires 3.3V power.
- The ESP8266 needs to communicate via serial at 3.3V and does not have 5V tolerant inputs, so you need level conversion to communicate with a 5V microcontroller like most Arduino use.

## III. IMPLEMENTATION

***Hardware details of system designed***



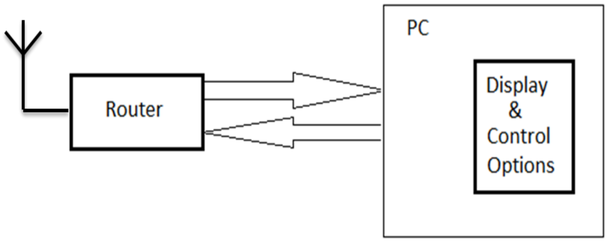


***Fig 5**: Block Diagram*

Temperature sensor: temperature sensor will simply acquire analog temperature value from the physical world and convert it into an equivalent voltage value.

Max31855: the data obtained from temperature sensor is in the order of micro volts, this data won't be read by the microcontroller directly as it is analog in nature. That is why we use the max31855 ic that converts the micro volts data into desired volts range as well as digitize the output.

Microcontroller: microcontroller will compute the data and control its flow.

LCD: LCD will simply display the output of the microcontroller. Transeiver module: this module will transmit as well as receive data.

PC: computer at the receiver side will display data received from the micro controller. It will also send data to the micro controller if needed**.**

***Interfacing MAX31855 with Arduino***

Pin Configuration of MAX31855:
(a) Connect D0 to Digital Pin 3
(b) Connect CS to Digital Pin 4
(c) Connect CLK to Pin 5

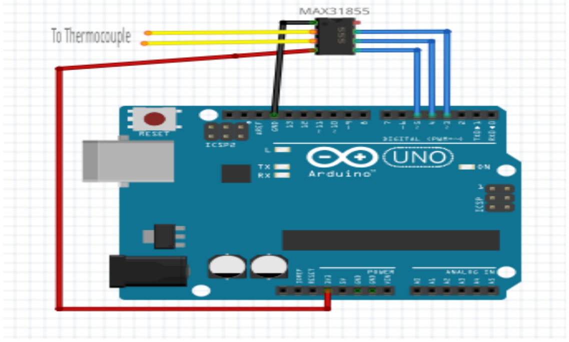


***Fig 6: Interfacing MAX31855 with Arduino***

***LCD interfacing with Arduino***

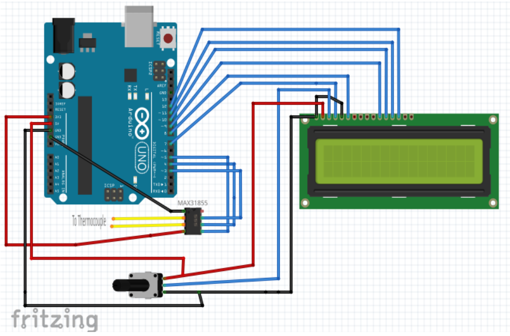


*Fig 7: LCD interfacing with Arduino*

***Pin Configuration:***

(a) LCD RS pin to digital pin 7

(b) LCD Enable pin to digital pin 8

(c) LCD D4 pin to digital pin 9

(d) LCD D5 pin to digital pin10

(e) LCD D6 pin to digital pin 11

(f) LCD D7 pin to digital pin 12

***ESP8266-01 Interfacing with Arduino***

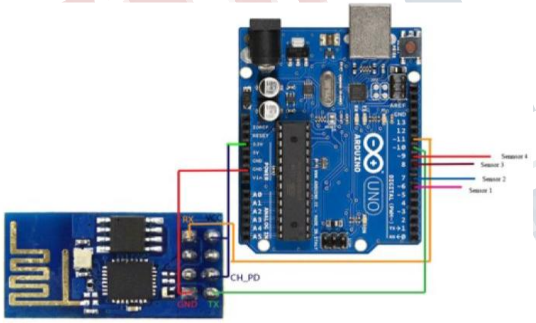


*Fig 8: Interfacing ESP8266 to Arduino*

***Pin Configuration of ESP:***

(a) Connect TX to Digital Pin 2
(b) Connect RX to Digital Pin 6

## IV. SOFTWARE DESIGN AND DEVELOPMENT

***Working***: Following process goes on step by step when hardware is powered.

- Configuring of Wi-Fi is done and is indicated on LCD display has CONFIGURE WIFI on first line and after configuration process done is indicated by OK message on second line of LCD display.

- IP address is displayed on LCD display.

- Then configuration settings is complete and system comes to online and LCD display.

- Temperature is measured and indicated on LCD display TEMPERATURE {DEG}.

- Now the temperature has to be sent to the webpage through Wi-Fi cloud (Thingspeak).The data sent can be viewed on html webpage.

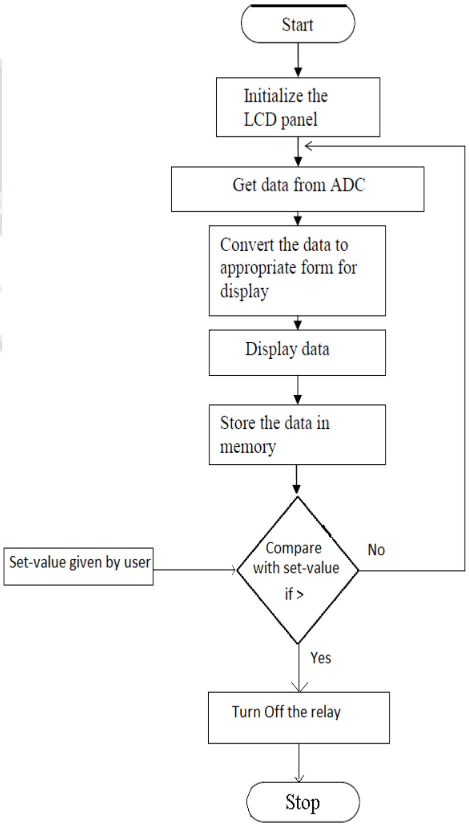
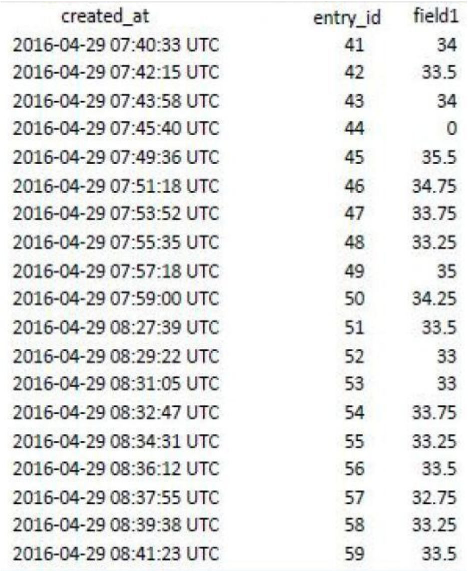


*Fig 9: Flow Diagram*



| created_at | entry_id | field1 |
|---|---|---|
| 2016-04-29 07:40:33 UTC | 41 | 34 |
| 2016-04-29 07:42:15 UTC | 42 | 33.5 |
| 2016-04-29 07:43:58 UTC | 43 | 34 |
| 2016-04-29 07:45:40 UTC | 44 | 0 |
| 2016-04-29 07:49:36 UTC | 45 | 35.5 |
| 2016-04-29 07:51:18 UTC | 46 | 34.75 |
| 2016-04-29 07:53:52 UTC | 47 | 33.75 |
| 2016-04-29 07:55:35 UTC | 48 | 33.25 |
| 2016-04-29 07:57:18 UTC | 49 | 35 |
| 2016-04-29 07:59:00 UTC | 50 | 34.25 |
| 2016-04-29 08:27:39 UTC | 51 | 33.5 |
| 2016-04-29 08:29:22 UTC | 52 | 33 |
| 2016-04-29 08:31:05 UTC | 53 | 33 |
| 2016-04-29 08:32:47 UTC | 54 | 33.75 |
| 2016-04-29 08:34:31 UTC | 55 | 33.25 |
| 2016-04-29 08:36:12 UTC | 56 | 33.5 |
| 2016-04-29 08:37:55 UTC | 57 | 32.75 |
| 2016-04-29 08:39:38 UTC | 58 | 33.25 |
| 2016-04-29 08:41:23 UTC | 59 | 33.5 |

*Fig 10: Recorded Data on Excel*

## V. CONSTRUCTION AND TESTING

PCB or a printed circuit board is a component made out of layers of insulating materials with electrical conductors. The insulator can be made of many materials mainly plastic, FIBRE glass and ceramics. During manufacture, the extra conductors that are not required are etched off, leaving printed circuit made of those required conductors where we can attach our electronic components. An actual-size, single-side PCB layout of the wi-fi temperature logger is shown in fig below**.**

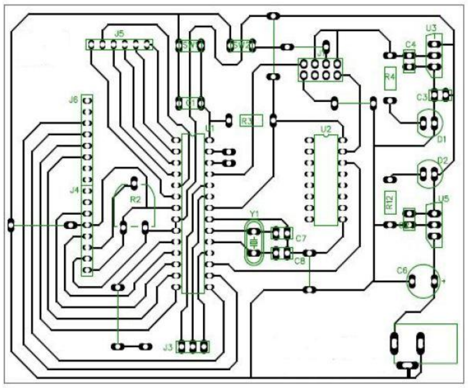


***Fig 11*: PCB Layout**

Now to about the testing of this circuit, power on the circuit, then the program will initialize the LCD and show the starting message on the LCD. If the LCD is showing the message properly go to www.thingspeak.com. First you need to make your own channel on the website by signing in the same. After signing in and saving your channel, you will receive 2 API keys and channel id. A write API key and a read API key, it is used to check the data from website on your smart phone or PC).

Before using the system, ensure that esp8266 module baud rate is changed to 9600 as most of these have 115200. Also, a wi-fi modem with an internet connection should be kept near your circuit board. Next, make the following changes in the espiot. ccode:

(a) Put your ssid-wi_ ssid .
(b) password-wi-fi password
(c) Replace API key with your own API key received

When you open your channel under data import/export section, you will get your API key. Call the link there and insert any field value that you want to upload. In private view section, you will see a graph of the data as shown in fig 12. For checking the results on a smart phone or pc, a link is provided in the respective channel. There API key is replaced by read API key and channel-id is replaced by the users channel id. Copy and paste this link in search bar of the browser in your smart phone or pc. You will get the recorded temperature data information. If temperature data has been uploaded, you will be able to see the graph here. Temperature data on www.thingspeak.com is shown in fig5.8**.**

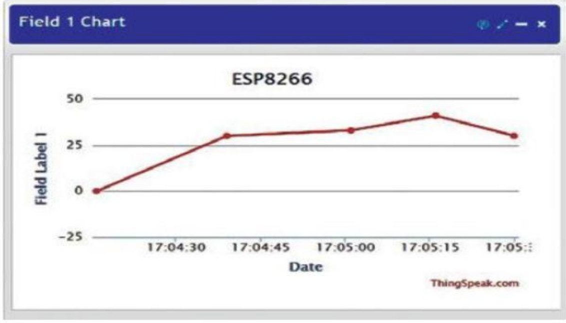


***Fig 12*: *Temperature Data On Website***

## VI. CONCLUSION AND FUTURE WORK

A cost effective Data Acquisition System was implemented. Whenever the supply is given to the system, the sensors connected to the system will be initialized and reads status of sensors continuously with real time, and these are displayed in our webpage using the microcontroller and these are updated by server through Wi-Fi module interfaced to controller. Data will be made available online on the webpage. The very existence of such a cheap and easy to use module is an indicator that soon the whole world will have a lot of data to analyze and make life easier and much more comfortable. Flexibility of this system allows you to include other sensors such as pressure, humidity etc which would simplify the problem of having a different system for the implementation of other parameter measurement systems.

## REFERENCES

1) Andrew J Thompson, John L Bahr and Neil R Thomson,”Low power data Logger”, proceedings of conference department of physics, university of otago,Dunedin.

2) Data logger and remote monitoring system for Multiple parameter measurement applicationsG.s. Nhivekar, r.r.mudholkerpublished on e -Journal of Science Technology (e-JST)

3) S. J. Perez, M. A. Calva, R. Castaneda, A microcontroller based data logging System Instrumentation and Development Vol. 3 Nr. 8, copyright 1997,Journal of the Mexican Society of Instrumentation.

4) H S kalsi, Electronic instrumentation, Tata McGraw-Hill Ltd., New Delhi,1999.