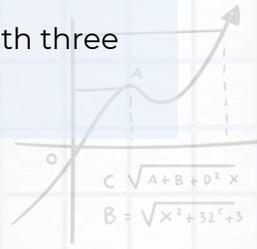


Build IoT Products (Energy Meter, Water Meter and an RF ID access control system)



Introduction

The Internet of Things (IoT) describes the network of physical objects “things”, that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. In this course we will teach you to build IoT system with three things Energy Meter, Water Meter and an RF ID access control system.



What does this course aim to achieve?

This course aims to teach how to build an energy meter, water meter and RFID tag reader as an IOT device to monitor energy, water consumption and the number of people and cars that have entered in a given lane. It also enables this data to be captured with an IOT system so simple reports can be generated from the information.

Energy meter

In an exceedingly power-hungry world, and in a time where we are keeping energy efficiency as one of the highest priorities in product development, the ability to accurately measure & calculate energy consumption at every point, becomes vital. Not only does this enable us to improve the product development phase, but also edge us towards a greener tomorrow.



Tasks for Energy meter



1. The build should detect the total energy being consumed
2. Values of V, I, Power & Power Factor to be figured.
3. Should find the cumulative energy consumed from the given input
4. Total power consumed in each period to be calculated.
5. Should send the data over Wi-Fi to the mobile.



What is being built in this course:

An energy meter to measure various parameters of the electricity passing through a given line. (Power, voltage, Currents etc.)

How is it being tested:

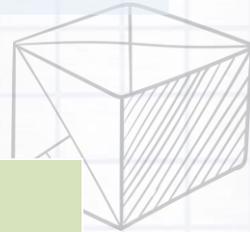
Give a specific predefined current input to one end of the line and check if the Energy meter can measure it accurately.

$$IX = A+B^x \\ = AB^x$$

Course Prerequisites

Basics of EEE, Basic Programming Language (C/Embedded C)

Component required for Energy Meter



S.no	Components	Quantity	Usage (one time or reusable)	Store
1	Dexter board	1	reusable	
2	Voltage sensor 250 V (ZMPT101B)	1	reusable	Turbo Enterprises
3	Current sensor 20 A (ACS712)	1	reusable	Turbo Enterprises
4	Bulb (load) 40,60 W	2	reusable	
5	Power adapter 12V, 2A	1	reusable	Turbo Enterprises
6	Esp 32 module	1	reusable	Turbo Enterprises



7	Bulb holder	1	reusable	
8	Soldering lead	1	reusable	Turbo Enterprises
9	Acrylic sheet	1	reusable	Anandha Stationery Stores
10	USB to TTL converter	1	reusable	Turbo Enterprises
11	Teflon wires 2 colours 2m	2	reusable	Turbo Enterprises
12	Brass stud 3x25 (optional)	24	reusable	Mercy Electronics
13	Brass stud 3x10 (optional)	20	reusable	Mercy Electronics
14	Male to female connecting wire	10	reusable	Turbo Enterprises

Where to source your components

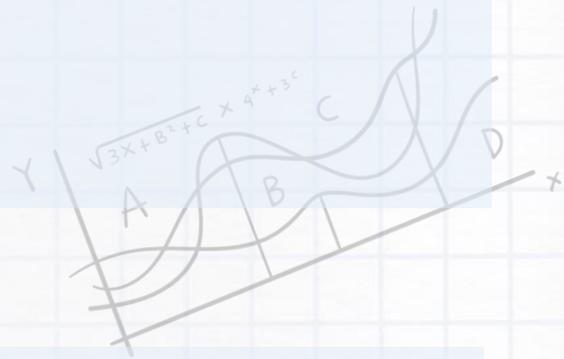
Turbo Electronics - +91 98404 19099 | www.robu.in

"Kindly check the above sources for components procurement. Do check for components availability & prices from local electronic component shops as well. Sourcing good components is the first step towards a successful Build, so do spend time identifying local vendors who will be able to supply you with components & accessories."

Step-by-step Procedure - Hardware Connection

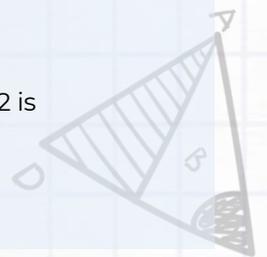
- For Current Measurement ACS712 current sensor is used
- Current sensor ACS712 is powered using the VCC, which is connected to Dexter board 3.3 V pin.
- Output pin of current sensor ACS712 is connected to the ADC pin (6) J4 of Dexter board.
- The input pins of ACS712 (1 & 2) are connected to the mains supply.
- Connect the phase of supply (230 V) with pin 1.
- Connect the load (bulb) with the pin 2 of ACS712 and complete the supply by connecting to the neutral.
- For Voltage Measurement ZMPT01B voltage sensor is used
- Voltage sensor ZMPT01B is connected to the supply (230V) through pins 1 & 2.
- Voltage sensor is powered using VCC pin which is connected to the 5 V pin in the Dexter board.
- Output pin of ZMPT01B is connected to pin (5) J4 of the Dexter board.
- For communicating the data from Dexter board to cloud, ESP 32 Wi-Fi module is used
- Connect Rx pin of ESP 32 module to Tx (JS-1) pin of Dexter board.

- ESP 32 module is powered using 5V supply from Dexter board
- Connect all the grounds
- Open Cubemx
- Save file and generate code in cube-ide
- Open Cube-ide
- Selected peripherals are initialized in default by Cubemx
- Go to stm32L4xx_it.c file
- Get the real time data from RTC
- Set the real time clock
- Enable timer and ADC
- Process the ADC readings through filter and calculate the offset voltage and current
- Calculate the rms voltage and current
- Calculate power factor
- Average the calculated rms values
- Send data to ESP 32 through UART
- Create string to receive data from Dexter board.
- Build the project
- Click on build project
- Select the Debug Configuration



Ubidots

- Open Ubidots for Education and Create new Account.
- Click on your profile and check for API credentials
- Copy the default token key and replace it in code.
- Set the Wi-Fi-SSID and password as the users mobile/Wi-Fi username and password
- For setting device label, go to Ubidots, click on devices and then click on add new device, the click on add new variable then copy variable and device label.
- After initialization, receive data through UART from the dexter board and extract the required string and convert value to float.
- Publish frequency means how frequently data is sent to the server (e.g., 2000ms - data is sent to the server once every 2 seconds)
- Send the data to the cloud using a specific API key. For any publish frequency (say 2 seconds), it is recommended to have frequency more than 5 seconds
- Create a variable ID and send the data to it
- Publish the data over MQTT
- Clear the serial buffer
- Now compile the code and upload to Esp32.
- Note : While uploading code to Esp32, Make sure that Uart Rx pin of Esp32 is disconnected .
- Send the data to ubidots



$$1 \times = A + B^x \\ = AB^x$$

Water flow meter

Flow meters act as diagnostic tools to check the consumption of any given substance flowing through a given line. In today's world where freshwater scarcity is immense, it's essential to know our own consumption and being able to build a reliable metering system that gives an accurate calculation of the water consumption would be vital.



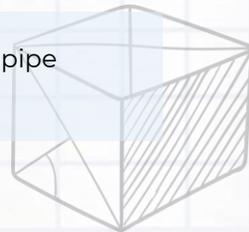
Tasks for Water flow meter

1. Build a small pipeline that connects to the Water Flow sensor's inlet/input end. [$\frac{1}{2}$ inch pipe]
2. Find the total volume of Liquid passing through the meter
3. Should calculate the cumulative of total liquid passed through in each time period.
4. Should send the read data over [Bluetooth] to the mobile

(Ensure dexter board is well shielded to not let any liquids damage the board)

What is being built in this course:

A water flow meter which can measure the amount of water flowing through a pipe and get the results displayed on an app using IoT.

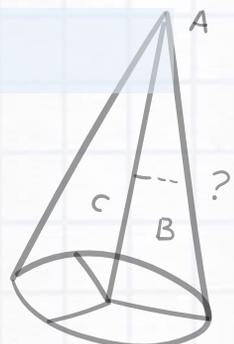


How is it being tested:

Give a specific predefined water input to the input of flow sensor and check if the meter measures accurately.

Course Prerequisites

Basics of EEE, Basic Programming Language (C/Embedded C)



Component required for water flow meter

S.no	Components	Quantity	Usage (one time or reusable?)	Online links
1	Dexter board	1	reusable	
2	HC-05 bluetooth module	1	reusable	Turbo Enterprises
3	Flow sensor YF-S201	1	reusable	Turbo Enterprises
4	ESP 32 module	1	reusable	Turbo Enterprises
5	$\frac{3}{4}$ PVC hose	1	reusable	NAVARANG electrical & hardware
6	0.5 inch PVC pipe	2	reusable	NAVARANG electrical & hardware
7	L -bend	2	reusable	
8	Dot board 4x6 cm ²	1	reusable	Turbo Enterprises
9	5mm nuts and bolt	4	reusable	NAVARANG electrical & hardware
10	Teflon wires 2 colour 2m	2	reusable	
11	Brass stud 3x25 (optional)	24	reusable	Mercy Electronics
12	Brass stud 3x10 (optional)	20	reusable	Mercy Electronics
13	Acrylic sheet (optional)	1	reusable	Anandha Stationery Stores
14	Male to female connecting wire	10	reusable	Turbo Enterprises
15	Female to female connecting wire	10	reusable	Turbo Enterprises



+B^x
B^x



Where to source your components

Turbo Electronics - +91 98404 19099 | www.robu.in

"Kindly check the above sources for components procurement. Do check for components availability & prices from local electronic component shops as well. Sourcing good components is the first step towards a successful Build, so do spend time identifying local vendors who will be able to supply you with components & accessories."



Step-by-step Procedure - Hardware Connection

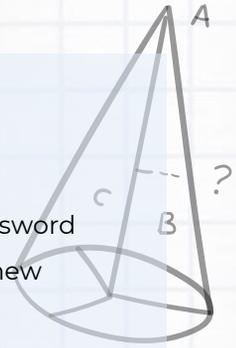
- Flow sensor YF-S201 powered using the 5 V (Vin) pin of Dexter board.
- Output pin of the flow sensor is connected to the GPIO pin (PC0) of the Dexter board.
- Connect the inlet of the water flow meter using a pipe to a water source.
- Connect the outlet of the water flow meter using a pipe to a container, in which the water would be collected. For communicating the data from Dexter board to cloud, ESP 32 Wi-Fi module is used
- Connect the Vin pin of ESP 32 module to the 5 V supply from Dexter board.
- Connect (Rx) pin of ESP 32 module to (Tx) - (PC4) pin of Dexter board.
- Connect the push button to the (PC1) pin of the Dexter board. The push button is used for clearing memory.
- Connect all the ground pins together.
- Open Cubemx
- Configure clock
- Disable external clock in RCC
- Select the pin which is used to read signal from flow sensor (PC0)
- Configure the sensor Pin as external Interrupt pin
- Configure another interrupt pin for memory clear (PC1)
- Configure SPI3 with the mentioned parameters as shown in the figure
- Configure UART3 with 9600 Baud Rate
- Save file and generate code in cube-ide
- Open Cube-ide
- Selected peripherals are initialized in default by Cubemx
- Go to stm32L4xx_it.c file
- Find external interrupt handler. Increment the counter whenever new interrupt comes, this counter represents the flow count.
- CALIBRATION
- Pour 1 liter of water through the water flow sensor and count the number of pulses
- (e.g., 1 liter = 400 pulses, then 1 pulse = 1/400 liter)
- Send data through UART to ESP 32.
- Finally Build the project.
- ESP 32
- Open Arduino
- Click file and go to references and paste the URL at additional board
- Go to tools, click on board, board manager, click on type ESP32 and install.
- Add the Wi-Fi library for Ubidots for Wi-Fi Connection

$$1X = A + B^x \\ = AB^x$$



Ubidots

- Open Ubidots for Education and Create new Account.
- Click on your profile and check for API credentials
- Copy the default token key and replace in code.
- Set the Wi-Fi-SSID and password as the users mobile/Wi-Fi username and password
- For setting device label, go to Ubidots, click on devices and then click on add new device, the click on add new variable then copy variable and device label.
- Paste the variable and device name
- After initialization, receive data through UART from the dexter board and extract the required string and convert value to float
- Publish frequency means how frequently data is sent to the server (e.g., 2000ms - data is sent to the server once in every 2 seconds)
- Send the data to the cloud using specific API key. For any publish frequency (say 2 seconds), it is recommended to have frequency more than 5 seconds
- Create a variable ID and send the data to it
- Publish the data over MQTT
- Clear the serial buffer
- Compile the code and upload to Esp32.
- Note : While uploading code to Esp32, Make sure that Uart Rx pin of Esp32 is disconnected



RF ID access control system

$$1 X = A + B^x \\ = AB^x$$

A radio frequency (RF) signal refers to a wireless electromagnetic signal used as a form of communication if one is discussing wireless electronics. Used in various applications across industries, we take a focus on access control mechanisms that are enabled using RF ID recognition.

Tasks for RF ID access control system

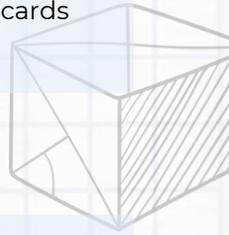
1. RFID reader should detect the RF tag and
 - a. Access granted display in LCD when correct tag
 - b. Access denied in display when wrong tag
2. Build a stepper motor-based Gate and program it to make a 90° turn to open gate and 90° turn in opp. direction to close the gate.
3. When Correct tag, the Toll gate should open
4. It should calculate the total no. of correct entries for the given period.
5. Should send the data over [GSM 2G/3G] to the mobile



What is being built in this course:



Build a RF based fast tag reader which can grant and deny access using the RFID cards and display the number of entered/denied cars.



How is it being tested:

Enable access to registered tags and deny access to unregistered tags and move the tollgate accordingly.

Course Prerequisites

Basics of EEE, Basic Programming Language (C/Embedded C)



Component required for RF ID control system

S.no	Components	Quantity	Usage (one time or reusable)	Store
1	Dexter board	1	reusable	
2	MFRC522 reader module	1	reusable	Turbo Enterprises
3	RFID tags	6	reusable	Turbo Enterprises
4	Power Adaptor 12V 2A	1	reusable	Turbo Enterprises
5	Voltage Regulator 7805	1	reusable	Turbo Enterprises
6	16*2 LCD display	1	reusable	Turbo Enterprises
7	SIM 900A GSM module	1	reusable	Turbo Enterprises
8	Sim 800 L	1	reusable	Turbo Enterprises
9	Dot board 6x8 cm ²	1	reusable	Turbo Enterprises
10	Push buttons	2	reusable	Turbo Enterprises

11	Micro servo motor SG 90	1	reusable	Turbo Enterprises
12	Soldering lead	1	one -time	
13	Brass stud 3x25(optional)	24	reusable	Mercy Electronics
14	Brass stud 3x10(optional)	20	reusable	Mercy Electronics
15	Acrylic sheet	1	reusable	Anandha Stationery Stores
16	Male to female connecting wire	10	reusable	Turbo Enterprises
17	Female to female connecting wire	10	reusable	Turbo Enterprises

Where to source your components

Turbo Electronics - +91 98404 19099 | www.robu.in

"Kindly check the above sources for components procurement. Do check for components availability & prices from local electronic component shops as well. Sourcing good components is the first step towards a successful Build, so do spend time identifying local vendors who will be able to supply you with components & accessories."

Step-by-step Procedure - Hardware Connection

- LCD connections
- Connect D4, D5, D6, D7, of LCD to pins 7, 8, 9, 10 of Dexter board.
- Connect the Enable pin of LCD to pin 6 of Dexter board.
- Connect RS pin of LCD to J5 of Dexter board.
- Connect R/W pin to ground pin of Dexter board.
- EM18 reader module
- Connect Tx pin of EM18 reader module to Rx (PC5) of dexter module
- GSM
- Connect Rx pin of GSM module to Tx (PA2) pin 11 (J4) of Dexter board.
- Servomotor
- Connect the VCC pin of servo motor to the 5V supply pin of Dexter board.
- Connect the output of servo motor to PC6 of dexter board.
- Voltage Regulator 7805
- Connect the input (+12 V) of 7805 to GSM.
- Connect (+5V) of 7805 to VCC pin of servo motor.
- Push-button
- Connect one pin of the push button1 to PA0 and other pin to GND for registering.
- Connect one pin of push button2 to PA4 and other pin to ground for erasing memory.

- Open Cubemx
- Configure Internal Clock
- Configure USART
- Configure SPI external flash
- Configure Timer
- Configure timer 1 for generating 1 microsecond delay for LCD display
- Configure GPIO
- Check for registrations interrupt
- Rise_flag in IRQ
- Register function – When the push button is pressed, the card is registered
- Check if the card is registered in the local and display the result in the LCD
- If access is granted open the gate using the servomotor
- Send the data to server
- Number of cards granted access and denied can be viewed using Thingspeak portal
- Create an account in Thingspeak
- Go to channel settings and setup a new channel for data logging
- Copy the API key and add it in the code.\
- The data can be logged in every 2 minutes and visualised using Thingspeak portal

