International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206490

Software Defined Embedded Systems to Control Electronics Without Using Internet

Arunapriya R
Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamil Nadu, India

Article Info
Volume 6, Issue 4
Page Number: 501-508
Publication Issue :
July-August-2020

Article History
Accepted : 15 Aug 2020
Published : 22 Aug 2020

ABSTRACT

Thecontrol system is designed to assist and provide support in order to fulfill the needs of elderly and disabled in home. Automation of the surrounding environment of a modern human being allows increasing his work efficiency and comfort. There has been a significant development in the area of an individual's routine tasks and those can be automated. In the present times, we can find most of the people clinging to their mobile phones and smart devices throughout the day. Hence with the help of his companion-a mobile phone, some daily household tasks can be accomplished by personifying the use of the mobile phone. Analyzing the current smart phone market, most users are opting for Android based phones. It has become a second name for a mobile phone in layman terms. Home Automation System has been designed for mobile phones having Android platform to automate an WIFI interfaced Arduino which controls a number of home appliances like lights, fans, bulbs and many more using on/off relay. This paper presents the automated approach of controlling the devices in a household that could ease the tasks of using the traditional method of the switch. The most famous and efficient technology for short range wireless communication-WIFI is used here to automate the system. The HAS system for Android users is a step towards the ease of the tasks by controlling one to twenty-four different appliances in any home environment.

Keywords : Message Queuing Telementry Transport, Message Access Control, Transmission Control Protocol / Internet Protocol, Arduino IDE.

I. INTRODUCTION

Software-defined network technology is an approach to network management that enables dynamic, programmatically efficient network configuration in order to improve network performance and monitoring making it more like cloud computing than traditional network management. In our research we use SDN as a key to create an embedded system to control electronics without internet by creating a local MQTT server with an IoT devices and it will be controlled via mobile phone.

IoT is especially relevant to Smart Grid since It provides systems to gather & act on power-related information in an automated fashion within goal to

improve efficiency, reliability, economics, & sustainability of production & distribution of electricity. There are several planned or ongoing large-scale deployments of IoT, to enable best system of cities & systems. For example, Songdo, South Korea, first of its kind fully clean & wired smart city, is near completion. Ambient intelligence & autonomous control are not part of original facts of Internet of Things. In future Internet of Things might be a non-deterministic & open network in which auto-organized or intelligent entities Web services, SOA components, virtual objects also known as avatars will be interoperable & able to act independently pursuing their objectives & shared ones depending on context, circumstances or environments.

II. EMBEDDED SYSTEMS

An embedded system is a computer system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today.Ninetyeight percent of all microprocessors are manufactured as components of embedded systems.

Examples of properties of typically embedded computers when compared with general-purpose counterparts are low power consumption, small size, rugged operating ranges, and low per-unit cost. This comes at the price of limited processing resources, which make them significantly more difficult to program and to interact with. However, by building intelligence mechanisms on top of the hardware, taking advantage of possible existing sensors and the existence of a network of embedded units, one can both optimally manage available resources at the unit and network levels as well as provide augmented functions, well beyond those available. For example, intelligent techniques can be designed to manage power consumption of embedded systems. Modern embedded systems are often based on microcontrollers (i.e. CPU's with integrated memory or peripheral interfaces),but ordinary microprocessors (using external chips for memory and peripheral interface circuits) are also common, especially in more-complex systems. In either case, the processor(s) used may be types ranging from general purpose to those specialized in certain class of computations, or even custom designed for the application at hand.

III. EMBEDDED COMPUTER SYSTEMS

An Embedded System is one that has computer hardware with software embedded in it as one of its important components.

An embedded computer is frequently a computer that is implemented for a particular purpose. In contrast, an average PC computer usually serves a number of purposes: checking email, surfing the internet, listening to music, word processing, etc... However, embedded systems usually only have a single task, or a very small number of related tasks that they are programmed to perform.

An embedded computer system is an electronic system, which includes a microcomputer.It is configured to perform a specific dedicated application Software is programmed into ROM.This software is not accessible to the user of the device, and software solves only a limited range of problems.Here the microcomputer is embedded or hidden inside the system.

Each embedded microcomputersystem, accepts inputs, performs calculations, and generates outputs and runs in “real time”. A typical automobile current a day contains an average of ten microcontrollers. In fact, modern houses maycontain as many as 150

microcontrollers and on average a consumer now interacts with microcontrollers up to 300 times a day. General areas that employ embedded microcomputers encompass every field of engineering namely: Communications, automotive, military, medical, consumer, machine control etc...

MQTT Protocol

It is an open OASIS and ISO standard lightweight, publish subscribe network protocol that transports messages between devices. The protocol usually runs over TCP\IP. However, any network protocol that provides ordered, lossless, bi-directional connections can support MQTT. It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited.

MQTT broker

A software running on a computer could be self-built or hosted by a third party. Available as open source and proprietary with extra features added.

The broker acts as a post office, MQTT doesn't use the address of the intended recipient but uses the subject line called "Topic", and anyone who wants a copy of that message will subscribe to that topic. Multiple clients can receive the message from a single broker. Similarly, multiple publishers can publish topics to a single subscriber.

Each client can both produce and receive data by both publishing and subscribing, i.e. the devices can publish sensor data and still be able to receive the configuration information or control commands MQTT is a bi-directional communication protocol. This helps in both sharing data, managing and controlling devices.

With MQTT broker architecture the devices and application become decoupled and more secure. MQTT uses Transport layer security encryption with user name, password protected connections, and optional certifications that requires clients to provide a certificate file that matches with the servers. The clients are unaware of each other IP address.

In case of single source of failure, broker software and clients have an automatic handover to Redundant/automatic backup broker. The backup broker can also be setup to share the load of clients across multiple servers onsite, cloud, or the combination of both.

The broker can support both standard MQTT and MQTT for compliant specifications such as Sparkplug, can be done with same server, same time and with same levels of security.

IV. ADVANTAGES

1. Eliminates vulnerable and insecure client connections
2. Can easily scale from a single device to thousands
3. Manages and tracks all client connection states, including security credentials and certificates
4. Reduced network strain without compromising the security
5. MQTT is designed for devices which run on low power and low bandwidth
6. MQTT is optimized for battery usage
7. Faster response
8. Publish/subscribe mechanism

Related Work

- Implementation of MQTT broker
- MQTT client application
- Arduino IDE
- Performance analysis
- Performance

V. IMPLEMENTATION OF MQTT BROKER

Installation Use the standard "pip" tool for installation, eitherpip install paho-mqttOrpip3 install

paho-mqttdepending on whether you are installing on Python 2 or Python 3Connect This section shows the API usage how to connect with the library to a MQTT broker.There are a number of ways to connect to a broker, a simple but incomplete example is given below:

```
importpaho.mqtt.client as paho
defon_connect(client, userdata, flags, rc):
print("CONNACK received with code %d." % (rc))
client = paho.Client()
client.on_connect = on_connect
client.connect("broker.mqttdashboard.com", 1883)
```

This creates a client instance using the default parameters, assigns a call back to be called once a successful connection has occurred, and starts the connection.

MQTT CLIENT APPLICATION

Use the MQTT Android application to connect to MQTT server, subscribe, and publish to a topic.

- Open the MQTT sample application
- Click the plus sign (+) to open a new MQTT connection
- Enter any unique identifier into the client ID field. Be patient, the keystrokes can be slow.
- Enter the Server field into the IP address of your MQTT server. E.g. mqtt.eclipse.org
- Enter the port of the MQTT connection. The default port number for a normal MQTT connection is 1883.
- Click Connect. If the connection is successful, you see a Connecting messages.

ARDUINO IDE

The Arduino IDE comes with the Ethernet library needed, but the MQTT library needs to be installed. Navigate to Sketch > Include Library > Manager Libraries, and search for MQTT in the search field. The library that we will use is called "PubSubClient" which is a lightweight library for use with MQTT.

Create a new file, call it whatever you want and then include the following libraries at the top of your program. SPI is needed for the Ethernet shield, the Ethernet is the Ethernet library that gets passed to the PubSubClient.h library. This handles the MQTT protocol and messaging.

The first step into using MQTT is to define a number of variables, including the IP address, MAC, server, and some objects. The first line in our program is a function prototype of the function that will handle incoming messages, but this will be looked at more in depth later. The next few lines create our MAC address – which must be unique – and the IP address of our Ethernet. It should be noted that this IP address will be ignored if our router can assign one for us. If it can't, it will fall back and use this IP address.

In our setup function, we need to start by enabling the serial port so we can see the status of our Arduino and begin an Ethernet connection. The begin () function takes two arguments: the MAC address and the IP address (which we simply pass). It's at this point that we also include a small delay to allow the Ethernet Shield to do its thing

The next step is to set the MQTT broker that we will be communicating on. The. Connect () function takes several arguments including usernames and passwords, but since we are using a public testing MQTT broker, we only need to define a user ID. For now, our user ID will be "my ClientID". This function will return a Boolean value depending on the success of the connection to the broker.

If the connection was successful, then it returns true. Otherwise, it returns false. If the connection is true then we create an event handler that will call the function "subscribe Receive" upon receiving a message.

PERFORMANCE ANALYSIS

In this module, the performance of the proposed network method to handle electronics via software-based connections and the end result is vastly deferent from previous existing systems and it's a onetime setup afterwards it doesn't require constant maintenance and monthly internet connections. From the comparison result, final RESULT is concluded

PERFORMANCE

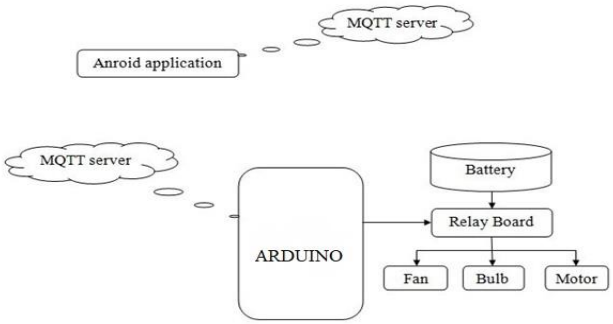
The developed system is robust and flexible in operation. The system was to perform the remote monitoring and control of appliances effectively. Local and remote user interfaces are easy to handle by a novice consumer and are efficient in handling the operation.Depending on the inhabitant usages, appliances connected by smart sensing units are controlled by automation based on the tariff conditions or by the inhabitant locally using GUI and remotely using the website.

VI. METHODOLOGY

System design is the process of defining the architecture, component, modules, and data for system to satisfy specified requirement. One could see it as the application of system theory to product development. There is some overlap with a disciplines of system analysis, systems architecture and systems engineering. If the broader topic of product development blends the perspective of marketing, design, and manufacturing into a single approach to product development, then design is the act of taking the marketing information and creating the design of product to be manufactured. System design is therefore the process of defining and developing system to satisfy specified requirements of the user.

MQTT CLIENT APPLICATION

The Paho project has been created to provide reliable open-source implementations of open and standard messaging protocols aimed at new, existing, and emerging applications for Machine-to-Machine (M2M) and Internet of Things.Paho reflects the inherent physical and cost constraints of device connectivity. Its objectives include effective levels of decoupling between devices and applications, designed to keep markets open and encourage the rapid growth of scalable Web and Enterprise middleware and applications.



Architecture Diagram

ARDUINO IDE

Arduino is an open source, computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world. The project's products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public Licenseor the GNU General Public License, permitting the manufacture of Arduino boards and software distribution by anyone.

Arduino boards are available commercially in preassembled form, or as do-it-yourself kits. Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus on some models.

The microcontrollers are typically programmed using a dialect of features from the programming languages C and C++. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environmentbased on the Processing language project. The Arduino project started in 2005 as a program for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.

HARDWARE DESCRIPTION

POWER SUPPLY CIRCUIT:

Power supply is a reference to a source of electrical power. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a power supply unit or PSU. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

Power supplies for electronic devices can be broadly divided into linear and switching power supplies. The linear supply is a relatively simple design that becomes increasingly bulky and heavy for high current devices; voltage regulation in a linear supply can result in low efficiency. A switched-mode supply of the same rating as a linear supply will be smaller, is usually more efficient, but will be more complex.

Linear Power supply:

An AC powered linear power supply usually uses a transformer to convert the voltage from the wall outlet (mains) to a different, usually a lower voltage. If it is used to produce DC, a rectifier is used. A capacitor is used to smooth the pulsating current from the rectifier. Some small periodic deviations from smooth direct current will remain, which is known as ripple. These pulsations occur at a frequency related to the AC power frequency

Transformer:

1) **Fig Transformer** *Transformers convert AC electricity from one voltage to another with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC.*

2) Step-up transformers increase voltage, step-down transformers reduce voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage to a safer low voltage.The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the core.

3) *Transformers waste very little power so the power out is (almost) equal to the power in. Note that as voltage is stepped down current is stepped up.*

The ratio of the number of turns on each coildetermines the ratio of the voltages. A step-down transformer has a large number of turns on its primary coil which is connected to the high

voltage mains supply, and a small number of turns on its secondary coil to give a low output voltage.

Regulator:Voltage regulator ICs are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current ('overload protection') and overheating ('thermal protection').

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solids stateelectronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustablevoltages and current.

Many of the fixed voltage regulator ICs has 3 leads and look like power transistors, such as the 7805 +5V 1A regulator shown on the right. They include a hole for attaching a heat sink if necessary.

Positive regulator

1. input pin
2. ground pin
3. output pin

It regulates the positive voltage

Negative regulator

4. ground pin
5. input pin
6. output pin

It regulate the negative voltage

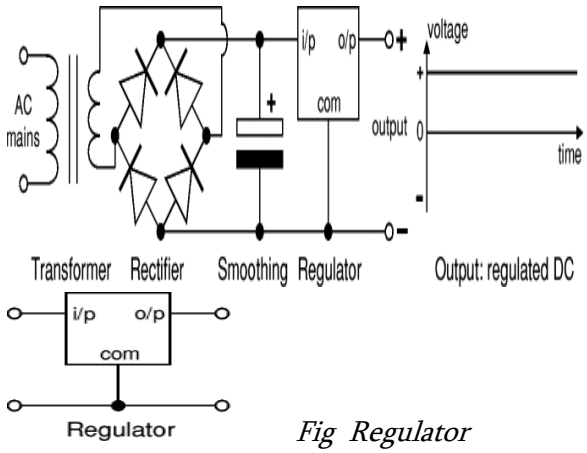


Fig Regulator

The regulated DC output is very smooth with no ripple. It is suitable for all electronic circuits.

ARDUINO UNO

The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software1.0. The Uno board and version 1.0 of Arduino Software were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 programmed as a USB-to-serial converter

The Arduino project started at the Interaction Design Institute Ivrea in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of $100, a considerable expense for many students. In 2003 Hernando Barragán created the

development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language. The project goal was to create simple, low-cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it *Arduino*. Early arduino boards used the FTDI USB-to-serial driver chip and an ATmega168. The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

VII. CONCLUSION

We proposed an electronics controlling system that able to control electronic without internet and high processing power. It uses mqtt protocol to transfer data between devices and it doesn't consume lot of energy when compared to other automation system. The reason we use mqtt protocol is the communication should be speed and when the data strength is low. It is a light weight protocol that occupies low bandwidth and consume less power. The ease of wireless network access through WIFI router. MQTT client application is built on Arduinouno.

A prototype of MQTT based home automation system is implemented on Arduino. The sensors and actuators connected to Arduino are remotely monitored and controlled through a common home gateway. The existing infrastructure can be used to enhance the home appliances and make them smart. This implementation provides an intelligent, comfortable and energy efficient home automation system. It also assists the old and differently abled persons to control the appliances in their home in a better and easier way.

VIII. REFERENCES

[1]. Seung-Chul Son, Nak-Woo Kim, Byung-Tak Lee, Chae Ho Cho, Jo Woon Chong, "GSM-BASED HOME AUTOMATION SYSTEMS", Ieee Transactions On Consumer Electronics, Vol. 62, No. 1, Pp. 10-16, February 2017

[2]. MohdHelmyAbdWahab, Norzilawati Abdullah, AyobJohari, Herdawatie Abdul Kadir, "GSM BASED ELECTRICAL CONTROL SYSTEM FOR SMART HOME APPLICATION"Ieee Computer Society, Pp. 804-809

[3]. Takeshi Yashiro, Shinsuke Kobayashi, Noboru Koshizuka, Ken Sakamura, "AN INTERNET OF THINGS ARCHITECTURE FOR EMBEDDED APPLIANCES", Electrical And Control Engineering 2011 International Conference, Pp. 2578-2581, 2011

[4]. N. Datta, T. Masud, R. Arefm, A. A. Rimon, M. S. Rahman And B. B. Pathik,"DESIGNING AND IMPLEMENTATION OF AN APPLICATION BASED ELECTRICAL CIRCUIT FOR SMART HOME APPLICATION" Automation Quality And Testing Robotics 2014 Ieee International Conference, Pp. 1-5.

[5]. Harshal, Regina, Prince Marys,"ENHANCED HOME AUTOMATION SYSTEM USING INTERNET OF THINGS" Ieee Trans. Auto Tech, Vol. 67, No. 3, Pp. 2543–2556, Mar. 2018

[6]. J.Wang And F.Jingqi,"MOBILE BASED HORNE AUTOMATION USING INTERNET OF THINGS"Berlin, Germany: Springer, 2015, Pp. 128–137

Cite this article as : Arunapriya R, "Software Defined Embedded Systems to Control Electronics Without Using Internet", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 4, pp. 501-508, July-August 2020. Available at doi : https://doi.org/10.32628/CSEIT206490
Journal URL : http://ijsrcseit.com/CSEIT206490