ISTANBUL AYDIN UNIVERSITY

FACULTY OF ENGINEERING
ELECTRICAL AND ELECTRONIC ENGINEERING

GRADUATION PROJECT

IOT smart home

OGR.GOR. ABDULKADIR ALWER

Abdalrhman AL-Afandi B1705.020042
Omar Shirwa B1805.020056
Hamzah Saleh B1805.020058

Contents

SUMMARY

The Internet of Things (IoT) has a rich technological legacy and a bright future. ubiquitous connectivity has created a new paradigm, and the closed, static, and bounded systems of the past will soon be obsolete. With the connection of low-cost sensors to cloud plat- forms, it's now possible to track, analyze, and respond to operational data at scale. The promise of the IoT is indeed wonderful.

intelligent systems made up of smart machines that talk with each other and with people in real time, and data analytics driving optimization and transformation in industries as varied and far-reaching as aeronautics and agriculture, transportation and municipal services, manufacturing, and healthcare, and even within our homes.

ABTRACT

Smart home systems have grown in popularity in recent decades since it improves people's comfort and quality of life. Smartphones and microcontrollers are used to control most smart home systems devices. Using wireless connection technology, a smartphone application is utilized to control and monitor home functions.
We utilize a microcontroller from the AVR family for this system. To receive user commands over the internet, this microcontroller is connected to a WIFI modem. We also have an LCD display that shows the status of the system. To switch loads, relays are utilized. A 12 transformer provides electricity to the entire system. After receiving user commands over the internet, the microcontroller interprets them to regulate the loads and display the system status on an LCD display. As a result, this method enables effective home automation over the internet.

ACKNOWLEDGEMENT

Firstly, we would like to refer our gratitude to Allah who gives us the faith and power to continue in this road in a good health and we hope everything we did in the university be in the way to satisfy his grace.

On that record we would like to acknowledge our university Istanbul Aydin University and its academicians that supported us during our lectures in the university.

We are grateful to Dr. ABDULKADIR ALWER for believing in us and teaching us all the way up to finish this book, this work can't be done without his support and kindness.

We are grateful to the head of department Prof. Öğr.EYLEM GÜLCE ÇOKER for her support and being by our side the whole time.

We are grateful to Prof. Dr. NECİP GÖKHAN KASAPOĞLU, lecturer, in the department of Electric and Electronic Engineering. We are extremely thankful to him for sharing experience and guidance and encouragement extended to us.

We are grateful to Dr. Mohammed F.M Alkrunz, lecturer, in the department of Electric and Electronic Engineering. He was kind and we are so thankful for his kindness and for his experience that he shared with us.

We are grateful to all of those with whom we have had the pleasure to work with during these years, our thanks to our friends and colleagues too.

Finally, our deep and sincere gratitude to our families for their continuous and unparalleled love, help and support.

Introduction

When electrical equipment is plugged in but not in use, there is usually still a flow of power. That means we'll waste roughly 5% to 10% of the electrical energy we use while using it, resulting in waste of money and resources. Furthermore, this could result in several accidents, such as a conflagration caused by an electrical short circuit. As a result, many people who frequently forget to unplug electrical devices must remind themselves each time they leave the house. On the other side, if they fail to unplug, they will have to return home to pull the plug out to avoid harmful scenarios, wasting valuable time. Smart home technologies will be necessary to solve these issues. With the advancement of technology, several smart home research projects have been established to help people live better lives. A smart home is one in which the technology used to make all electronic equipment around the house act "smart" or "intelligent" or more automated is used to make all electronic equipment around the house act "smart" or "intelligent" or smarter. Lighting, climate control, security, and a variety of other services are all automated in a smart house. A smart gadget is a regular appliance that has a powerful computer installed to provide it additional functionality and the ability to monitor a wide range of daily activities. A smart house is beneficial to everyone and can be utilized to improve daily life at home.

As a result, a smart home is made up of three components: a network, controlling devices, and home automation. The network, which can be wired or wireless, is utilized to connect the automation to the controlling devices.

The systems are managed using controlling devices. Furthermore, home automation enables easy control of the physical surroundings.

Figure 1 IOT Smarthome

IOT

Introduction of IOT

The Internet of Things (IoT) is the next step in the growth of the Internet, and it establishes a global infrastructure that connects machines and humans. When the Internet first became widely available in the early 1990s, the first wave of its use and deployment was primarily focused on the impact on everyday services and applications, which disrupted previously established models for financial transactions, shopping, news consumption, and information sharing. From banking and retail shopping to face-to-face contact and government services, it was a revolution that digitized a wide variety of services as we knew them. The first two decades of the Internet revolution were dominated by consumer services and corporations, but it was a human-centric revolution. For customers, new business models for banking, online shopping, video communication, and other services have emerged. Business to business models and the cloud have had a tremendous impact on businesses, wiping out entire industries that failed to adapt to the revolution's rapid pace. The economic consequences have been enormous. Because of the Internet's reach into our homes and workplaces, we now witness and experience a new way of life, more than two decades later. The advancements in communication technology that allowed for the adoption and success of the Internet at home and at work had a side consequence.

In contrast to the operational technology (OT) environment, which controls physical equipment, and the information technology (IT) environment, where humans use computers for work, the development of complex interconnections among machines in the operational environment. The existing automated industrial environment welcomed developing technologies with open arms, adopting the most appropriate and establishing a network architecture, primarily private, that enables highly productive industrial operations. It was only a natural progression for the Internet to incorporate these processes. Furthermore, industrial control models have been enhanced and employed in a wide range of application domains, taking advantage of smart devices – that is, devices that have processing, memory, and networking resources – that are deployed in various locations.

Transportation, aeronautics, energy production and distribution, manufacturing, and health care all use comparable control methods, relying on smart sensors, actuators, and devices to automate control for sophisticated applications. Today, these technologies are used to run countries' critical infrastructure. The Internet-of-Things (IoT) is the next logical step in the Internet revolution, which began roughly three decades ago. Importantly, the Internet of Things is erecting a global infrastructure that will impact every aspect of our lives, from agriculture to mining, health care to manufacturing and transportation. Clearly, it will offer the infrastructure on which the new AI revolution will be built, and it will allow us to experience a new way of life as the Internet expands its reach into our homes and workplaces.

The advancements in communication technologies that allow for the adoption and success of the Internet at home and at business has a side consequence. In contrast to the operational technology (OT) environment, which regulates physical equipment, the information technology (IT) environment, where humans use computers for work, the development of sophisticated interconnections among machines in the operational environment. The existing automated industrial environment accepted developing technologies well, embraced the appropriate ones, and built a network architecture, primarily private, that enables highly productive industrial processes. It was only a natural progression for the Internet to incorporate these processes.

Additionally, The control models for the industrial environment have been extended and employed in a wide range of application domains, taking advantage of smart devices – that is, devices that include processing, memory, and networking resources – that are deployed in various environments. Transportation, aeronautics, energy production and distribution, manufacturing, and health care all use comparable control methods, relying on smart sensors, actuators, and devices to automate control for sophisticated applications. Today, these technologies are used to run countries' critical infrastructure. The Internet-of-Things (IoT) is the next logical step in the Internet revolution, which began roughly three decades ago. Importantly, the Internet of Things is erecting a global infrastructure that will impact every aspect of our lives, from agriculture to mining, health care to manufacturing and transportation. It will, without a doubt, provide the architecture upon which the new AI revolution will be built.

What Is IoT?

The Internet of Things (IoT) has become a popular topic of discussion and marketing strategy. IoT has evolved as a significant technology with applications in a variety of industries, despite the hype. Pervasive information systems, sensor networks, and embedded computers are all precursors of the Internet of Things. The phrase IoT system, rather than Internet of Things, more correctly represents how this technology is used. The majority of IoT devices are connected to build purpose-specific systems; they are employed as general-access devices on a global network less commonly. IoT expands on pervasive computing and information systems, which were previously focused on data. Pervasive computing devices include smart refrigerators, for example. Built-in PCs were added in several items, allowing consumers to enter information about the contents of their refrigerator for menu planning. To handle data entry, conceptual gadgets would scan the contents of the refrigerator automatically. Menu planning applications for stand-alone personal computers are like the use cases envisioned for these refrigerators. Sensor network research took on a variety of forms. Many of these were created with the goal of collecting data at very low rates. After then, the data would be sent to servers to be processed. In-network processing was not emphasized in traditional sensor network research. Embedded computing was mostly focused on stand-alone devices or tightly connected networks like those found in automobiles. Both consumer electronics and hyperphysical systems were key embedded computer application categories, emphasizing designed systems with well-defined goals. No one, unambiguous definition of IoT technology has arisen, owing to the vast range of supporters. We can think of a few possibilities: • Soft real-time sensor networks • Internet-enabled physical devices, albeit many do not use the Internet Protocol • Networks of embedded computer devices that are dynamic and evolving This phrase is used to describe two features. First, rather than being a collection of Internet-connected devices, the system is tailored to one or a few applications. Second, the IoT system takes physical system dynamics into account. An IoT system may consist mostly of sensors, but it may also incorporate many actuators in some circumstances. The goal is to process signals and time-series data in both circumstances.

The availability of microelectromechanical (MEMS) sensors has sparked interest in the Internet of Things. Sensors with built-in accelerometers, gyroscopes, chemical sensors, and other features are now commonly available. These sensors' low cost and low power consumption open up new possibilities much beyond those of typical laboratory or industrial measurement equipment.

IoT systems are being pushed toward signal processing by these sensor applications. The low cost of VLSI digital and analog circuitry also helps to allow the Internet of Things. As we'll see, IoT nodes don't rely on cutting-edge VLSI fabrication techniques.

In fact, they are inexpensive because they can make use of older manufacturing lines; the lower device counts available in these older technologies are more than sufficient for many IoT systems. IoT systems must consume very little power. Power consumption is a key factor in total cost of ownership for IoT systems. Achieving the necessary power levels requires careful attention to hardware design, software design, and application algorithms. Security and safety are key design and operational requirements for IoT systems. As we have argued elsewhere, safety and security are no longer separable problems. The merger of computational and physical systems requires us to merge the previously separate tasks of safe physical system design and secure computer system design.

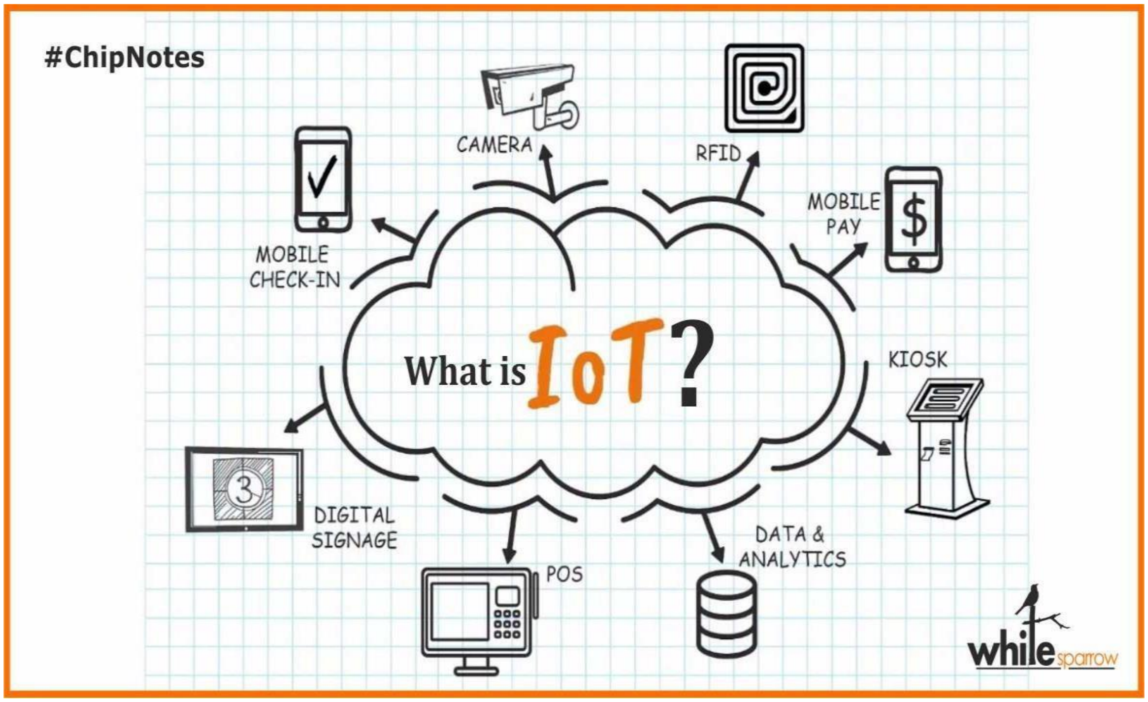


Figure 2 What is IOT ?

Applications of IOT

- IoT systems can be used for a variety of purposes, including:
- Sensors are used in industrial systems to monitor both the industrial processes and the quality of the product as well as the condition of the equipment. Sensors that collect data to forecast approaching motor failures are becoming more common in electric motors, for example.
- Sensors in smart buildings determine people's locations as well as the state of the building. To cut operational expenses, this data can be utilized to manage heating, ventilation, and air conditioning systems, as well as lighting systems. Sensors are also used in smart buildings and structures to monitor structural health.
- Sensors are used in smart cities to monitor pedestrian and vehicular traffic, and data from smart buildings may be integrated. 1 3rd Generation Internet of Things (IoT)

• Vehicles employ networked sensors to monitor the vehicle's condition, resulting in improved dynamics, lower fuel consumption, and lower emissions.

• Medical systems link a variety of patient monitoring sensors, which could be found at home, in emergency vehicles, in the doctor's office, or at the hospital.

Use cases assist us in comprehending the needs of an IoT system.

Network of sensors The system could be used solely to collect data from a set of sensors.

Data from sensors can be gathered and evaluated by the alert system. When certain criteria are met, alerts are sent out.

Data from sensors is gathered and evaluated by the analysis system, however in this case, the analysis is ongoing. Reports on analytic results can be generated on a regular basis – hourly, daily, etc. – or they can be updated on a continuous basis. System that reacts Actuators may be triggered as a result of sensor data analysis. We use the term reactive to describe systems that do not follow standard control laws.

System of control Sensor data is input into control algorithms, which produce actuator outputs.

A class of nonfunctional needs can be identified that apply to many IoT systems. The system's nonfunctional needs impose nonfunctional requirements on the components as well.

While event latency from capture to destination may not be critical for batch-oriented applications, it is critical for online analysis.

Throughput of events the throughput of the nodes, network bandwidth, and cloud throughput all influence the rate at which events may be recorded, transferred, and processed. Rate of event loss and buffer capacity the environment may produce more events in an interval than the system can produce if there are no strict upper bounds on event production rates. The desired capability is captured by the event loss rate, whereas buffer capacity is a more practical need that may be directly linked to component capabilities.

Service latency and throughput Ultimately, events will be processed by services. We can also specify the latency and throughput for services.

Reliability and availability Since IoT systems are distributed, reliability is more likely to be specified over parts of the network rather than reliability of the complete system. Availability is commonly used to describe distributed systems.

Service lifetime IoT systems are often expected to have longer lifetimes than we expect for PC systems. The lifetime of the system or a subset of the system may be considerably longer than that of a component, particularly if the system uses redundant sensors and other components.

A Technologist's Definition of the IoT

Kevin Ashton of the Massachusetts Institute of Technology (MIT) coined the term Internet of Things. At the time, industrial automation technologies were starting to move from the factory into new environments like hospitals, banks, and offices. This early form of intercommunication often involved machines of the same type such as a one ATM machine talking to another in the same general location hence the term, Machine-to-Machine, or M2M. As early M2M implementations grew increasingly more sophisticated, machines were connected to other kinds of devices like servers. Those servers ultimately moved from on-premise locations into data centers and eventually "the cloud." We can appreciate the prescience of Kevin Ashton's term. Yet while the "IoT" is a catchy phrase, it doesn't help us understand the full implications of this new paradigm. While the Internet is, of course a critical, enabling element, it is only a part of the essential concept the idea that we can connect our reality, part and parcel, to the virtual world of information systems that is so truly transformational for smart connected products and operations alike. Today, the Internet of Things can include industrial and commercial products, everyday products like dishwashers and thermostats, and local networks of sensors to monitor farms and cities. In an IoT solution, objects can be sensed and controlled through the Internet, whether these objects are remote devices, smart products, or sensors that represent the status of a physical location. And information can be made available to applications, data warehouses, and business systems.

How the IoT works

An IoT ecosystem is made up of web-enabled smart devices that gather, send, and act on data from their surroundings using embedded CPUs, sensors, and communication hardware. Connecting to an IoT gateway or other edge device allows IoT devices to share sensor data, which is then forwarded to the cloud for analysis or analyzed locally. These gadgets occasionally connect with other linked devices and act on the information they receive. People can engage with devices to set them, give them instructions, or retrieve data, but they can accomplish most of the job without human participation. The connectivity, networking, and communication protocols utilized with these web-enabled devices are heavily influenced by the IoT applications deployed.

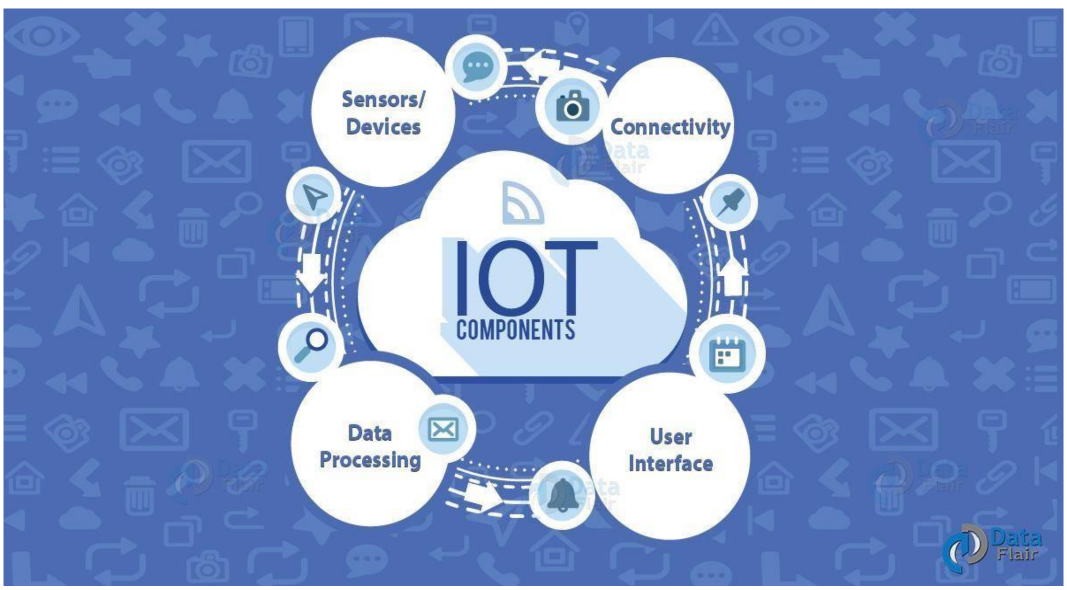


Figure 3 How IOT works ?

INTERNET OF THINGS DEFINITION EVOLUTION

IoT EMERGENCE

Kevin Ashton is accredited for using the term "Internet of Things" for the first time during a presentation in 1999 on supply-chain management . He believes the "things" aspect of the way we interact and live within the physical world that surrounds us needs serious reconsideration, due to advances in computing, Internet, and data-generation rate by smart devices. At the time, he was an executive director at MIT's Auto-ID Center, where he contributed to the extension of RFID applications into broader domains, which built the foundation for the current IoT vision.

INTERNET OF EVERYTHING

Since then, many definitions for IoT have been presented, including the definition that focuses mostly on connectivity and sensory requirements for entities involved in typical IoT environments. Whereas those definitions reflect IoT's basic requirements, new IoT definitions give more value to the need for ubiquitous and autonomous networks of objects where identification and service integration have an important and inevitable role. For example, Internet of Everything (IoE) is used by Cisco to refer to people, things, and places that can expose their services to other entities.

INDUSTRIAL IoT

Also referred to as Industrial Internet , Industrial IoT (IIoT) is another form of IoT applications favored by big high-tech companies. The fact that machines can perform specific tasks such as data acquisition and communication more accurately than humans has boosted IIoT's adoption. Machine to machine (M2M) communication, Big Data analysis, and machine learning techniques are major building blocks when it comes to the definition of IIoT. These data enable companies to detect and resolve problems faster, thus resulting in overall money and time savings. For instance, in a manufacturing company, IIoT can be used to efficiently track and manage the supply chain, perform quality control and assurance, and lower the total energy consumption

SMARTNESS IN IoT

- Another feature of IoT that has recently been highlighted in definitions is "smartness." This distinguishes the Internet of Things from comparable notions like sensor networks, and it can be further divided into "object smartness" and "network smartness." A smart network is a type of communication infrastructure that includes the following features:
- The communication protocols employed, from the layers that interface with the physical world (such as tags and sensors) through the communication layers between nodes and with the Internet, should be standardized and open.
- object addressability (direct IP address) and multifunctionality (the ability for a network designed for one application (for example, road traffic monitoring) to be used for other ones) (eg, environmental-pollution monitoring or traffic safety)

HUMAN IN THE LOOP

Machine-to-machine, human-to-machine, and human-with-environment interactions are all aided by the Internet of Things. The IoT trend is projected to transition toward the merging of smart and autonomous networks of Internet-capable items equipped with the ubiquitous computing paradigm as the number of smart devices grows and new protocols such as IPv6 are adopted.

Including humans in the IoT loop has several benefits for a variety of applications, including emergency management, healthcare, and so on. As a result, another critical role of IoT is to create a collaborative system capable of successfully responding to an event collected via sensors, as well as effective crowd discovery and information exchange between found crowds of various domains.

IMPROVING THE QUALITY OF LIFE

IoT is also recognized by the impact on quality of life and businesses , which can revolutionize the way our medical systems and businesses operate by: (1) expanding the communication channel between objects by providing a more integrated communication environment in which different sensor data such as location, heartbeat, etc. can be measured and shared more easily. (2) Facilitating the automation and control process, whereby administrators can manage each object's status via remote consoles; and (3) saving in the overall cost of implementation, deployment, and maintenance, by providing detailed measurements and the ability to check the status of devices remotely. According to Google Trends, the word "IoT" is used more often than "Internet of Things" since 2004, followed by "Web of Things" and "Internet of Everything" as the most frequently used words. Quoting the same reference, Singapore and India are the countries with the most regional interest in IoT. This is aligned with the fact that India is estimated to be the world's largest consumer of IoT devices by 2020.

Figure 4 Industry IOT

Automation

What is Automation ?

Human decision-making and manual command-response operations are replaced by automation, which uses logical programming commands and mechanized equipment. Mechanization, such as the employment of a timing mechanism to trigger a lever or a ratchet and pawl, has historically benefited humans in executing the physical requirements of a task. Automation, on the other hand, goes a step further than mechanization, drastically lowering the need for human sensory and cerebral requirements while also increasing productivity. The term "automation" is thought to have been coined in the 1940s by a Ford Motor Company engineer to describe various systems in which automatic actions and controls replaced human effort and intelligence. Control devices at the time were electromechanical in nature. Logic was performed by means of relays and timers interlocked with human feedback at decision points. By wiring relays, timers, push buttons, and mechanical position sensors together, simple logical motion sequences could be performed by turning on and off motors and actuators. With the advent of computers and solid-state devices, these control systems became smaller, more flexible, and less expensive to implement and modify. The first programmable logic controllers were developed in the 1970s and 1980s by Modicon in response to a challenge by GM to develop a substitute for hardwired relay logic. As technology improved and more automation companies entered the market, new control products were developed.

Advantages of automation

A few advantages of automation are:

- Human operators performing tasks that involve hard physical or monotonous work can be replaced.
- Human operators performing tasks in dangerous environments, such as those with temperature extremes or radioactive and toxic atmospheres, can be replaced.

Tasks that are beyond human capabilities are made easier. Handling heavy or large loads, manipulating tiny objects, or the requirement to make products very quickly or slowly are examples of this.

- Production is often faster and labor costs less on a per product basis than the equivalent manual operations.
- Automation systems can easily incorporate quality checks and verifications to reduce the number of out-of-tolerance parts being produced while allowing for statistical process control that will allow for a more consistent and uniform product.
- Automation can serve as the catalyst for improvement in the economies of enterprises or society. For example, the gross national income and standard of living in Germany and Japan improved drastically in the 20th century, due in large part to embracing automation for the production of weapons, automobiles, textiles, and other goods for export.
- Automation systems do not call in sick.

Disadvantages of automation

Some disadvantages of automation are:

• Current technology is unable to automate all desired tasks. Some tasks cannot be easily automated, such as the production or assembly of products with inconsistent component sizes or in tasks where manual dexterity is Automation and Manufacturing required. There are some things that are best left to human assembly and manipulation.

• Certain tasks would cost more to automate than to perform manually. Automation is typically best suited to processes that are repeatable, consistent, and high volume.

• The research and development cost of automating a process is difficult to predict accurately beforehand. Since this cost can have a large impact on profitability, it is possible to finish automating a process only to discover that there is no economic advantage in doing so. With the advent and continued growth of different types of production lines, however, more accurate estimates based on previous projects can be made.

Initial costs are relatively high. The automation of a new process or the construction of a new plant requires a huge initial investment compared with the unit cost of the product. Even machinery for which the development cost has already been recovered is expensive in terms of hardware and labor. The cost can be prohibitive for custom production lines where product handling and tooling must be developed.

• A skilled maintenance department is often required to service and maintain the automation system in proper working order. Failure to maintain the automation system will ultimately result in lost production and/or bad parts being produced. Overall, the advantages would seem to outweigh the disadvantages. It can be safely said that countries that have embraced automation enjoy a higher standard of living than those that have not. At the same time, a concern is often aired that automating tasks takes jobs from people that used to build things by hand.

Figure 5 Automation

Smart home

Introduction

Classic smart home, internet of things, cloud computing and rule-based event processing, are the building blocks of our proposed advanced smart home integrated compound. Each component contributes its core attributes and technologies to the proposed composition. IoT contributes the internet connection and remote management of mobile appliances, incorporated with a variety of sensors. Sensors may be attached to home related appliances, such as air-conditioning, lights and other environmental devices. And so, it embeds computer intelligence into home devices to provide ways to measure home conditions and monitor home appliances' functionality. Cloud computing provides scalable computing power, storage space and applications, for developing, maintaining, running home services, and accessing home devices anywhere at anytime. The rule-based event processing system provides the control and orchestration of the entire advanced smart home composition. Combining technologies in order to generate a best-of-breed product, already appear in recent literature in various ways. Christos Stergioua merged cloud computing and IoT to show how the cloud computing technology improves the functionality of the IoT. Majid Al-Kuwari focused on embedded IoT for using analyzed data to remotely execute commands of home appliances in a smart home. Trisha Datta proposed a privacy-preserving library to embed traffic shaping in home appliances. Jian Mao enhanced machine learning algorithms to play a role in the security in a smart home ecosystem. Faisal Saeed proposed using sensors to sense and provide in real-time, fire detection with high accuracy.

What is smart home ?

Smart home is defined as a home made up of electronic equipment and appliances remotely controlled by computers, cell phones, tablets or any device that is connected to the Internet, either through an application or a specific program. For example, a fridge that is monitored from the cell phone to increase or decrease its energy consumption, the lights of a house that are turned on or off without switches, a washing machine that controls a distance or a television with voice assistant that also allows transmit what is being viewed in the fridge.

In other words, a smart home is a space that a person can monitor and change a disposition to improve the quality of life. Today, thanks to home automation, there are no limits. This set of systems that allow to control a house aims to improve the quality of life of users in controls as varied as comfort, security, energy saving and entertainment.

However, a decisive issue compared to smart homes is also connectivity. Both the internet network and Bluetooth connections must be of high capacity. In addition, for these procedures it is necessary to have an optimal security system (or firewall). Apps like Samsung's SmartThings allow users to control all or at least most of the features of compatible appliances from the palm of their hand. For this brand, Samsung's Connected Living line provides a wide range of devices including cell phones, smart watches, state-of-the-art refrigerators, washers / dryers, air conditioners and televisions with artificial intelligence that can be connected to each other. So the whole house ‘talks’ from the application or websit. And this is not a distant dream, but a reality. The nightmare of being outside and remembering that, although you put the clothes in the washing machine, you never turned them on or you went on vacation with the concern of leaving something on. If you left home and forgot to turn off the TV, you can trun it off. Or turn off your child's TV and make sure it stays that way. You can also set your home temperature on a sunny day or just check that everything is in order just by checking your app or the website. A couple of years ago Kevin Ashton, technology pioneer responsible for the Massachusetts Institute of Technology (MIT) Auto ID Network Center and creator of the term "Internet of Things," spoke to Semana about technological evolution. During the conversation, Ashton stated that "technology is an essential part of humanity and what makes human beings unique is the constant development of new technological capabilities." Following that logic and seeing the advances that companies like Samsung, with their Connected Living line, have introduced to the Smart Home market, we can see that technology is serving the most basic human needs, thus facilitating their lives and democratizing the technology.

How Smart Homes Work ?

A smart home's devices are connected with each other and can be accessed through one central point—a smartphone, tablet, laptop, or game console. Door locks, televisions, thermostats, home monitors, cameras, lights, and even appliances such as the refrigerator can be controlled through one home automation system. The system is installed on a mobile or other networked device, and the user can create time schedules for certain changes to take effect. Smart home appliances come with self-learning skills so they can learn the homeowner's schedules and make adjustments as needed. Smart homes enabled with lighting control allow homeowners to reduce electricity use and benefit from energy-related cost savings. Some home automation systems alert the homeowner if any motion is detected in the home when they're away, while others can call the authorities—police or the fire department—in case of imminent situations. Once connected, services such as a smart doorbell, smart security system, and smart appliances are all part of the internet of things (IoT) technology, a network of physical objects that can gather and share electronic information. Security and efficiency are the main reasons behind the increase in smart home technology use. Smart homes can feature either wireless or hardwired systems—or both. Wireless systems are easier to install. Putting in a wireless home automation system with features such as smart lighting, climate control, and security can cost several thousand dollars, making it very cost-friendly. Hardwired systems, on the other hand, are considered more reliable and are typically more difficult to hack. A hardwired system can increase the resale value of a home. But there is a drawback—it's fairly expensive. Installing a luxury and hardwired smart system can cost homeowners tens of thousands of dollars.

Smart home services

Measuring home conditions A typical smart home is equipped with a set of sensors for measuring home conditions, such as: temperature, humidity, light and proximity. Each sensor is dedicated to capture one or more measurement. Temperature and humidity may be measured by one sensor, other sensors calculate the light ratio for a given area and the distance from it to each object exposed to it. All sensors allow storing the data and visualizing it so that the user can view it anywhere and anytime. To do so, it includes a signal processer, a communication interface and a host on a cloud infrastructure.

Managing home appliances

Creates the cloud service for managing home appliances which will be hosted on a cloud infrastructure. The managing service allows the user, controlling the outputs of smart actuators associated with home appliances, such as such as lamps and fans. Smart actuators are devices, such as valves and switches, which perform actions such as turning things on or off or adjusting an operational system. Actuators provides a variety of functionalities, such as on/off valve service, positioning to percentage open, modulating to control changes on flow conditions, emergency shutdown (ESD). To activate an actuator, a digital write command is issued to the actuator.

Controlling home access

Home access technologies are commonly used for public access doors. A common system uses a database with the identification attributes of authorized people. When a person is approaching the access control system, the person's identification attributes are collected instantly and compared to the database. If it matches the database data, the access is allowed, otherwise, the access is denied. For a wide distributed institute, we may employ cloud services for centrally collecting persons' data and processing it. Some use magnetic or proximity identification cards, other use face recognition systems, finger print and RFID. In an example implementation, an RFID card and an RFID reader have been used. Every authorized person has an RFID card. The person scanned the card via RFID reader located near the door. The scanned ID has been sent via the internet to the cloud system. The system posted the ID to the controlling service which compares the scanned ID against the authorized IDs in the database.

Smart home and IoT examples

1- **Discovery of water leaks and its prevention**

First step is deploying water sensors under every reasonable potential leak source and an automated master water valve sensor for the whole house, which now means the house is considered as an IoT. In case the water sensor detects a leak of water , it sends an event to the hub, which triggers the “turn valve off” application. The home control application then sends a “turn off” command to all IoT appliances defined as sensitive to water stopping and then sends the “turn off” command to the main water valve . An update message is sent via the messaging system to these appearing in the notification list . This setup helps defending against scenarios where the source of the water is from the house plumbing. The underlying configuration assumes an integration via messages and commands between the smart home and the IoT control system. It demonstrates the dependency and the resulting benefits of combining smart home and IoT.

2- Smoke detectors

Most houses already have the typical collection of smoke detectors , but there is no bridge to send data from the sensor to a smart home hub. Connecting these sensors to a smart home app , enables a comprehensive smoke detection system. It is further expanded to notify the elevator sensor to block the use of it due to fire condition , and so, it is even further expanded to any IoT sensor, who may be activated due to the detected smoke alert. In figure they designed a wireless sensor network for early detection of house fires. They simulated a fire in a smart home using the fire dynamics simulator and a language program. The simulation results showed that the system detects fire early.

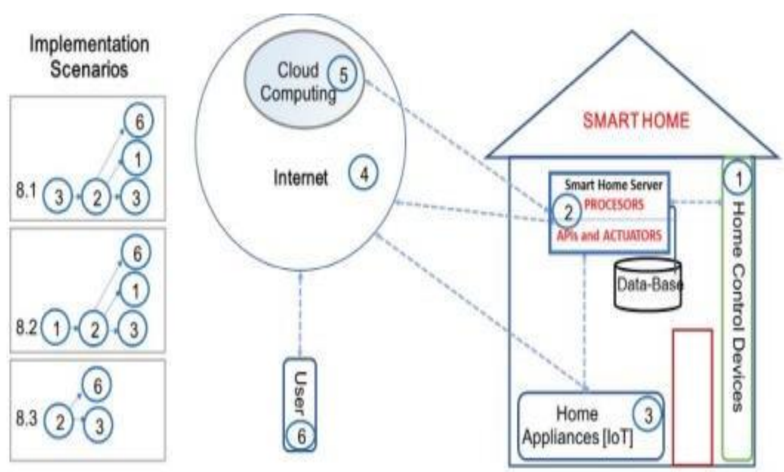


Figure 6 Advanced smart home implementations chart

Incident management to control home appliances

Consider the scenario where you leave home while some of the appliances are still on. In case your absence is long enough, some of the appliances may over heat and are about to blowout. To avoid such situations, we connect all IoT appliances' sensors to the home application , so that when all leave home it will automatically adjust all the appliances' sensors accordingly , to avoid damages. Note that the indication of an empty home is generated by the Smart Home application, while the "on" indication of the appliance, is generated by IoT. Hence, this scenario is possible due to the integration between smart home and IoT system.

Smart home challenges

1) Challenge one: Security Smart home also comes with some security concerns. For instance, hackers can access the network system. They have the ability to control all smart devices especially the security appliances.

2) Challenge two: Adaption to New Environment Owning a smart home means having to learn how to use your home that requires you to adapt to many innovations around you such as security systems and many sensors that always detect your movement. Accordingly, it will take reading manuals and learning about how-to of your home.

3) Challenge three: High Cost of Intelligence Although smart homes have many properties that makes human's lives convenient, these smart properties are in a higher price tag. The cost of an intelligent home is high because some of the technology is relatively new.

Components and their function in our system

Atmega328P Microcontroller

The ATmega328 is a single-chip microcontroller created by Atmel in the megaAVR family (later Microchip Technology acquired Atmel in 2016). The high-performance Microchip 8-bit AVR RISC-based microcontroller unites 32KB ISP flash memory with read-while-write capabilities, 1KB EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three adjustable timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire sequential interface, SPI serial port, 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes.

Figure 7 Atmega328p

Software Requirements

The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and provides simple one-click mechanism to compile and load programs to an Arduino board. A program written with the IDE for Arduino is called a "sketch". The Arduino IDE supports the languages C and C++ using special rules to organize code. The Arduino IDE supplies a software library called Wiring from the Wiring project, which provides many common input and output procedures. A typical Arduino C/C++ sketch consist of two functions that are compiled and linked with a program stub main() into an executable cyclic executive program: After compiling and linking with the GNU toolchain, also included with the IDE distribution, the Arduino IDE employs the programavrdude to convert the executable code into a text file in hexadecimal coding that is loaded into the Arduino board by a loader program in the board's firmware. Arduino programs may be written in any programming language with compiler that produces binary machine code. Atmel provides a development environment for their microcontrollers, AVR Studio and the newer Atmel Studio, which can be used for programming Arduino.

Why we used ATmega microcontroller?

- 23 general purpose I/O lines
- A 6-channel 10-bit ADC (analog input)
- 6 output PWM channels (analog output)
- A serial programmable USART
- This allows very fast start-up combined with low power consumption
- Easily available in Market
- Low Cost and easy to use
- Because the esp2688 Can be programmed using Arduino IDE

Atmega328P Features

Program Memory Type	Flash
Program Memory Size (KB)	32
CPU Speed (MIPS/DMIPS)	20
SRAM Bytes	2,048
Data EEPROM/HEF (bytes)	1024
Digital Communication Peripherals	1-UART, 2-SPI, 1-I2C
Capture/Compare/PWM Peripherals	1 Input Capture, 1 CCP, 6PWM
Timers	2 x 8-bit, 1 x 16-bit

Figure 8 Atmega328P Features table

Wifi module (ESP8266)

The ESP8266 is the name of a micro controller designed by Espressif Systems. Espressif is a Chinese company based out of Shanghai. The ESP8266 advertises itself as a self-contained WiFi networking solution offering itself as a bridge from existing micro controller to WiFi and is also capable of running self contained applications. Volume production of the ESP8266 didn't start until the beginning of 2014 which means that, in the scheme of things, this is a brand new entry in the line-up of processors. And… in our technology hungry world, new commonly equates to interesting. A couple of years after IC production, 3rd party OEMs are taking these chips and building "breakout boards" for them. If I were to hand you a raw ESP8266 straight from the factory, it is unlikely we would know what to do with one. They are very tiny and virtually impossible for hobbyists to attach wires to allow them to be plugged into breadboards. Thankfully, these OEMs bulk purchase the ICs, design basic circuits, design printed circuit boards and construct pre-made boards with the ICs pre-attached immediately ready for our use. It is these boards that capture our interest and that we can buy for a few dollars on ebay. There are a variety of board styles available. The two that I am going to focus on have been given the names ESP-1 and ESP-12. It is important to note that there is only one ESP8266 processor and it is this processor that is found on ALL breakout boards. What distinguishes one board from another is the number of GPIO pins exposed, the amount of flash memory provided, the style of connector pins and various other considerations related to construction. From a programming perspective, they are all the same.

The ESP8266 specification

When we approach a new electronics device, we like to know about its specification. Here are some of the salient points:

Voltage	3.3V
Current consumption	10uA – 170mA
Flash memory attachable	16MB max (512K normal)
Processor	Tensilica L106 32 bit
Processor speed	80-160MHz
RAM	32K + 80K
GPIOs	17 (multiplexed with other functions)
Analog to Digital	1 input with 1024 step (10 bit) resolution
802.11 support	b/g/n/d/e/i/k/r
Maximum concurrent TCP connections	5

Figure 9 ESP8266 specification table

The question of determining how long an ESP8266 can run on batteries is an interesting one. The current consumption is far from constant. When transmitting at full power, it can consume 170mA but when in a deep sleep, it only need 10uA. That is quite a difference. This means that the runtime of an ESP8266 on a fixed current reservoir is not just a function of time but also of what it is doing during that time and that is a function of the program deployed upon it. The ESP8266 is designed to be used with a partner memory module and this is most commonly flash memory. Most of the modules come with some flash associated with them. Realize that flash has a finite number of erases per page before something fails. They are rated at about 10,000 erases. This is not normally an issue for configuration change writes or daily log writes, but if your application is continually writing new data extremely fast, then this may be an issue and your flash memory will fail.

Transport Control Protocol (TCP):

The set of rules that the Internet depends on. ESP8266 has the ability to set up WIFI connections. A high-level Wi-Fi network is the ability to participate in TCP/ IP connections over a wireless connection. You can make your ESP run on TCP / IP or UDP.

Access point (AP) and station (STA):

Once you start working with the ESP, you will encounter these terms frequently. Let's say we want to connect two devices to the internet but one of them is to be connected via the other device ,so the one which sharing the internet is activate the access point . Here is the first device which is an internet connection source is the access point (AP) and the other device that uses the internet is called a station (STA). The ESP8266 can be used in three modes, AP mode, STA mode or in both STA and AP modes (combined).

SSID:

This is a fairly simple term. Almost all of us have used WIFI. Its Wi-Fi network name is called SSID. When we have multiple access points for a station to connect to, the station must know the access point that it must be connected to, and then an identity is given to each access point (AP) known as the SSID.

API:

ESP8266 API is used to talk to the internet world. For example, if we want to know the time, climate or anything we should ask in the form of an API for the corresponding website. The website will receive this request and return the desired result back to our ESP. Supports serial communication hence compatible with many development platform like Arduino .Can be programmed using Arduino IDE or AT-commands or Lua Script.

How we use the wifi module in our project ?

We will use our wifi module to read some data from specific website witch we had designed so the wifi module will work station mood (STA) and the also the module will check any change happened in the website while the module is working

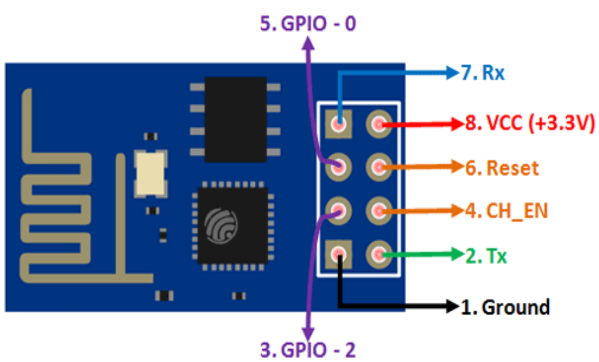


Figure 10 ESP8266

LCD Display

What is the LCD ?

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc. LCD modules are very commonly used in most embedded projects, the reason being its cheap price, availability and programmer friendly. Most of us would have come across these displays in our day to day life, either at PCO's or calculators. The appearance and the pinouts have already been visualized above ,now let us get a bit technical. 16×2 LCD is named so because; it has 16 Columns and 2 Rows. There are a lot of combinations available like, 8×1, 8×2, 10×2, 16×1, etc. but the most used one is the 16×2 LCD. So, it will have (16×2=32) 32 characters in total and each character will be made of 5×8 Pixel Dots. A Single character with all its Pixels is shown in the below picture. Now, we know that each character has (5×8=40) 40 Pixels and for 32 Characters we will have (32×40) 1280 Pixels. Further, the LCD should also be instructed about the Position of the Pixels. Since it will be a hectic task to handle everything with the help of MCU, thus an Interface IC like HD44780is used, which is mounted on the backside of the LCD Module itself. The function of this IC is to get the Commands and Data from the MCU and process them to display meaningful information onto our LCD Screen.

Features of LCD16x2

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5×8 pixel box
- The alphanumeric LCDs alphabets & numbers

- Is display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight
- It displays a few custom generated characters

Register of LCD

- In this module there are 2 main types of register first one is data register and the second one is command register. The RS pinout is used for the change the register.
- If we set zero then the register is command and at one data register will work.
- Now we discuss these two registers with the detailed.

Command Register

- The main function of this register is to save instructions shown on display.
- That help to a clearing of data changes the location of the cursor and display control.

Data Register

- This register saves the date to display on the liquid crystal screen. When we send data to liquid crystal display it moves to the data register, processing of that data will initiate.
- If we set the value of register at one then the data register will start operation.

Relay (MOC3021)

relays are switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. The MOC3021 is a Zero-Crossing TRIAC driven Optocoupler or Optoisolator. As we know the term Optocoupler/apostolater means the same that is we use light to indirectly couple to sets of circuits. The specialty of MOC3021 is that it has Zero-Crossing ability and is driven by a Triac. Since the output is driven by a TRIAC we can drive loads up to 400V and the triac can conduct in both directions hence controlling AC loads will not be a problem. Also, since it has zero-crossing ability, when the AC load is switched on for the first time the TRIAC will start conducting only after the AC wave reaches 0V this way we can avoid direct peak voltages to the Load and thus preventing it from getting damaged. It also has a decent rise and fall time and hence can be used to control the output voltage. This features of MOC3021 makes it an ideal choice for controlling high voltage AC loads through digital controllers like MPU/MCU. Since the output is controlled, we can control the intensity of the light or the speed of a AC motor. The MOC3021 can be used to switch loads by just turning the LED on or off, or we can also use PWM signals to switch the LED and thus the TRIAC. When we switch the TRIAC using PWM signals then the output voltage across the load can be controlled thus controlling the speed/brightness of the load.

When trying to switch AC loads it is important to understand the switching speed of the Opto-coupler. This switching speed depends in the amplitude of voltage that is being controlled by the TRIAC and the operating ambient temperature.

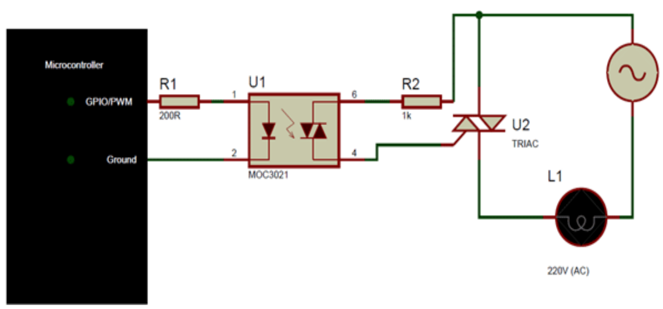


Figure 11 MOC*3021*

BT136 (TRIAC)

The BT136 is TRIAC with 4A maximum terminal current. The gate threshold voltage of the BT136 is also very less so can be driven by digital circuits. Since TRIACs are bi-directional switching devices they are commonly used for switching AC applications. So if we are looking to switch of control (dim, speed control) an AC load which consumes less than 6A with a digital device like microcontroller or microprocessor then BT136 might be good.

Applications

- AC Light dimmers
- Strode lights
- AC motor speed control
- Noise coupling circuits
- Controlling AC loads using MCU/MPU
- Ac/DC Power control

Since the TRIAC and SCR share most of the same characteristics, just like SCR the TRIAC will also not turn off when the gate voltage is removed. We need special type of circuit called commutation circuit to turn of the SCR again. This commutation is normally done by reducing the load current (forced commutation) less than the holding current. To put it simple the TRIAC will remain turned on only till the load current is greater than the holding current of the TRIAC.

BT136 can also be controlled through a microcontroller or a microprocessor. To do this we need an Opto-isolator like MOC3021 to isolate the AC circuit form Digital electronics. This way the Load can not only be switched but also the output voltage can be controlled by using PWM signals for fast switching

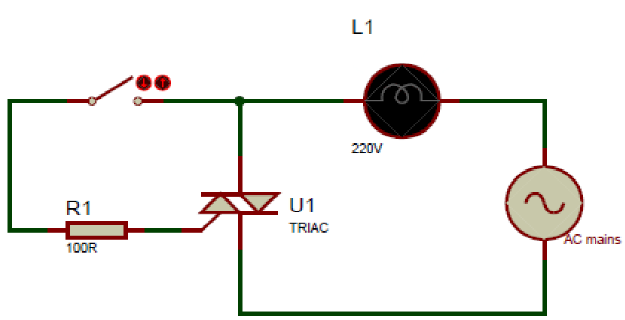


Figure 12 BT136

Resistors

The resistor is a passive electrical component to create resistance in the flow of electric current. In almost all electrical networks and electronic circuits they can be found. The resistance is measured in ohms. An ohm is the resistance that occurs when a current of one ampere passes through a resistor with a one volt drop across its terminals. The current is proportional to the voltage across the terminal ends. This ratio is represented by Ohm's law: formula with ohm's law: R=V/I. Resistors are used for many purposes. A few examples include delimit electric current, voltage division, heat generation, matching and loading circuits, control gain, and fix time constants. They are commercially available with resistance values over a range of more than nine orders of magnitude. They can be used to as electric brakes to dissipate kinetic energy from trains, or be smaller than a square millimeter for electronics

Resistors Code Calculation

Color	Value
Black (2nd and 3rd bands only)	0
Brown	1
Red	2
Orange	3
Yellow	4
Green	5
Blue	6
Violet	7
Grey	8
White	9

Figure 13 Resistors Code Calculation

Capacitors

- The capacitor is a component which has the ability or "capacity" to store energy in the form of an electrical charge producing a potential difference (Static Voltage) across its plates, much like a small rechargeable battery.
- There are many different kinds of capacitors available from very small capacitor beads used in resonance circuits to large power factor correction capacitors, but they all do the same thing, they store charge and filter the signals in some circiuts .

Fuse

- In electronics and electrical engineering, a fuse is an electrical safety device that operates to provide overcurrent protection of an electrical circuit. Its essential component is a metal wire or strip that melts when too much current flows through it, thereby stopping or interrupting the current. It is a sacrificial device; once a fuse has operated it is an open circuit, it must be replaced or rewired, depending on type.

Diodes

A diode is a two-terminal electronic component that conducts current primarily in one direction (asymmetric conductance); it has low (ideally zero) resistance in one direction, and high (ideally infinite) resistance in the other. A diode vacuum tube or thermionic diode is a vacuum tube with two electrodes, a heated cathode and a plate, in which electrons can flow in only one direction, from cathode to plate. The most common function of a diode is to allow an electric current to pass in one direction (called the diode's forward direction), while blocking it in the opposite direction (the reverse direction).

We used it in the project for this purpose:

- The diode is to provide a safe path for the inductive kickback of the motor. If you try to switch off the current in an inductor suddenly, it will make whatever voltage is necessary to keep the current flowing in the short term. Put another way, the current through an inductor can never change instantaneously. There will always be some finite slope. The motor is partially an inductor. If the transistor shuts off quickly, then the current that must still flow through the inductor for a little while will flow through the diode and cause no harm. Without the diode, the voltage across the motor would get as large as necessary to keep the current flowing, which would probably mean frying the transistor.

- We used it also as led .

Transistor(tip122)

- We used the tip122 in our circuit for switching the dc load, its connection would be as shown in the figure
- The emitter to ground ,base to the micro controller or the relay used , and collector should be connected to the load . when the tip122 switch on the load will be connected to the ground so the current flows through the load, when tip 122 is switch off the load will have no ground so the current will not flow

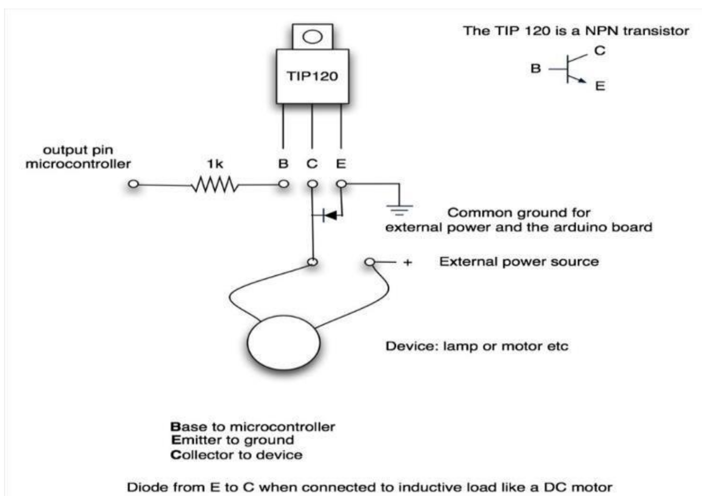


Figure 14 *TIP122*

Phototransistor (MCT2E)

- **MCT2E** is a phototransistor Optocoupler, as the name "phototransistor" suggests it has a transistor which is controlled based on light (photon). So this IC basically has an IR LED and a photo-transistor inside it. When the IR led is powered, the light from it falls on the transistor and it conducts.

- The **MCT2E** can be used in two modes, the Phototransistor mode and the photodiode mode. Out of which the Phototransistor mode is mostly used. In the Photo-transistor mode we will not be using the base pin (pin 6) of the transistor; we just have to connect the anode pin of the IR LED (pin 1) to the logic input which has to be isolated and the cathode (pin 2) of the IR led to the ground. Then Pull high the collector pin of the transistor using a resistor (here I have used 1K) and connect the collector pin to the output of your desired logic circuit. The Emitter (pin 4) is grounded.

Why we use it ?

- This IC is used to provide electrical isolation between two circuits, one part of the circuit is connected to the IR LED and the other to Photo-transistor. The digital signal given to the IR LED will be reflected on the transistor but there will be no hard electrical connection between the two. This comes in very handy when you are trying to isolate a noisy signal from your digital electronics, so we are looking for an IC to provide optical isolation in our circuit design , so that's why we used it .

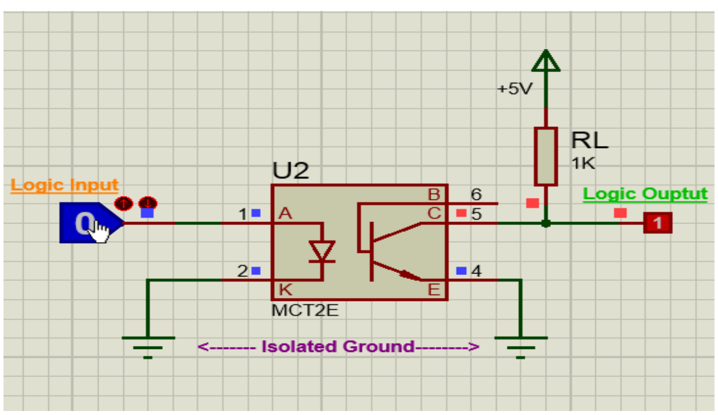


Figure 15 *MCT2E*

crystal oscillator

A crystal oscillator is an electronic oscillator circuit that uses the mechanical resonance of a vibrating crystal of piezoelectric material to create an electrical signal with a precise frequency. This frequency is often used to keep track of time, as in quartz wristwatches, to provide a stable clock signal for digital integrated circuits, and to stabilize frequencies for radio transmitters and receivers. The most common type of piezoelectric resonator used is the quartz crystal, so oscillator circuits incorporating them became known as crystal oscillators,but other piezoelectric materials including polycrystalline ceramics are used in similar circuits. A crystal oscillator, particularly one using a quartz crystal, works by distorting the crystal with an electric field, when voltage is applied to an electrode near or on the crystal; a property known as electrostriction or inverse piezoelectricity. When the electric field is removed, the quartz which oscillates at a precise frequency generates an electric field as it returns to its previous shape, and this can generate a voltage. The result is that a quartz crystal behaves like an RLC circuit, but with a much higher Q. Quartz crystals are manufactured for frequencies from a few tens of kilohertz to hundreds of megahertz. More than two billion crystals are manufactured annually. Most are used for consumer devices such as wristwatches, clocks, radios, computers, and cellphones. Quartz crystals are also found inside test and measurement equipment, such as counters, signal generators, and oscilloscopes.

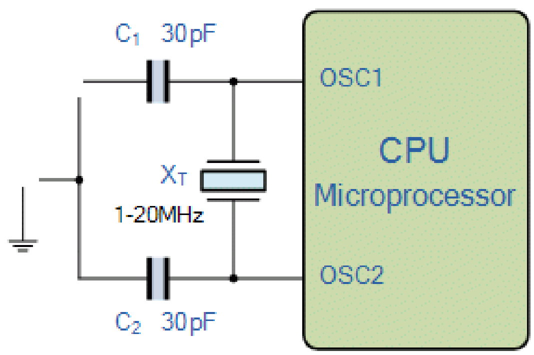


Figure 16 crystal oscillator

printed circuit board (PCB)

printed circuit board (PCB) A printed circuit board (PCB) mechanically supports and electrically connects electrical or electronic components using conductive tracks, pads and other features etched from one or more sheet layers of copper laminated onto and/or between sheet layers of a non-conductive substrate. Components are generally soldered onto the PCB to both electrically connect and mechanically fasten them to it. Printed circuit boards are used in all but the simplest electronic products. They are also used in some electrical products, such as passive switch boxes. Alternatives to PCBs include wire wrap and point-to-point construction, both were once popular but now rarely used. PCBs require additional design effort to lay out the circuit, but manufacturing and assembly can be automated. Electronic computer-aided design software is available to do much of the work of layout. Mass-producing circuits with PCBs is cheaper and faster than with other wiring methods, as components are mounted and wired in one operation. Large numbers of PCBs can be fabricated at the same time, and the layout only has to be done once. PCBs can also be made manually in small quantities, with reduced benefits. PCBs can be single-sided (one copper layer), double-sided (two copper layers on both sides of one substrate layer), or multi-layer (outer and inner layers of copper, alternating with layers of substrate). Multi-layer PCBs allow for much higher component density, because circuit traces on the inner layers would otherwise take up surface space between components. The rise in popularity of multilayer PCBs with more than two, and especially with more than four, copper planes was concurrent with the adoption of surface mount technology. However, multilayer PCBs make repair, analysis, and field modification of circuits much more difficult and usually impractical.

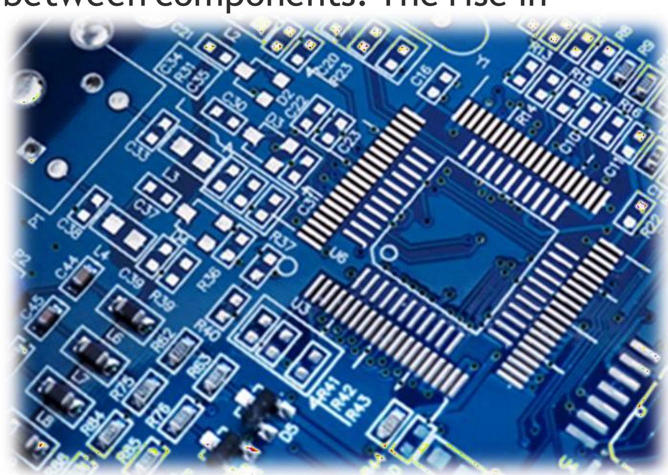


Figure 17 PCB

Transformer or adapter

A **transformer** is a passive electrical device that transfers electrical energy from one electrical circuit to another, or multiple circuits. A varying current in any one coil of the transformer produces a varying magnetic flux in the transformer's core, which induces a varying electromotive force across any other coils wound around the same core. Electrical energy can be transferred between separate coils without a metallic (conductive) connection between the two circuits. Faraday's law of induction, discovered in 1831, describes the induced voltage effect in any coil due to a changing magnetic flux encircled by the coil.

Transformers are most commonly used for increasing low AC voltages at high current (a step-up transformer) or decreasing high AC voltages at low current (a step-down transformer) in electric power applications, and for coupling the stages of signal processing circuits. Transformers can also be used for isolation, where the voltage in equals the voltage out, with separate coils not electrically bonded to one another. A wide range of transformer designs is encountered in electronic and electric power applications. Transformers range in size from RF transformers less than a cubic centimeter in volume, to units weighing hundreds of tons used to interconnect the power grid.

Adapter :

An **adapter** or **adaptor** is a device that converts attributes of one device or system to those of an otherwise incompatible device or system. Some modify power or signal attributes, while others merely adapt the physical form of one connector to another.

An AC adapter, AC/DC adapter, or AC/DC converter is a type of external power supply, often enclosed in a case similar to an AC plug. Other common names include plug pack, plug-in adapter, adapter block, domestic mains adapter, line power adapter, wall wart, power brick, and power adapter.

bridge rectifier

A bridge rectifier circuit is a common part of the electronic power supplies. Many electronic circuits require rectified DC power supply for powering the various electronic basic components from available AC mains supply. We can find this rectifier in a wide variety of electronic AC power devices like home appliances, motor controllers, modulation process, welding applications,

What is a Bridge Rectifier?

A Bridge rectifier is an Alternating Current (AC) to Direct Current (DC) converter that rectifies mains AC input to DC output. Bridge Rectifiers are widely used in power supplies that provide necessary DC voltage for the electronic components or devices. They can be constructed with four or more diodes or any other controlled solid state switches. Depending on the load current requirements, a proper bridge rectifier is selected. Components' ratings and specifications, breakdown voltage, temperature ranges, transient current rating, forward current rating, mounting requirements and other considerations are taken In to account while selecting a rectifier power supply for an appropriate electronic circuit's application.

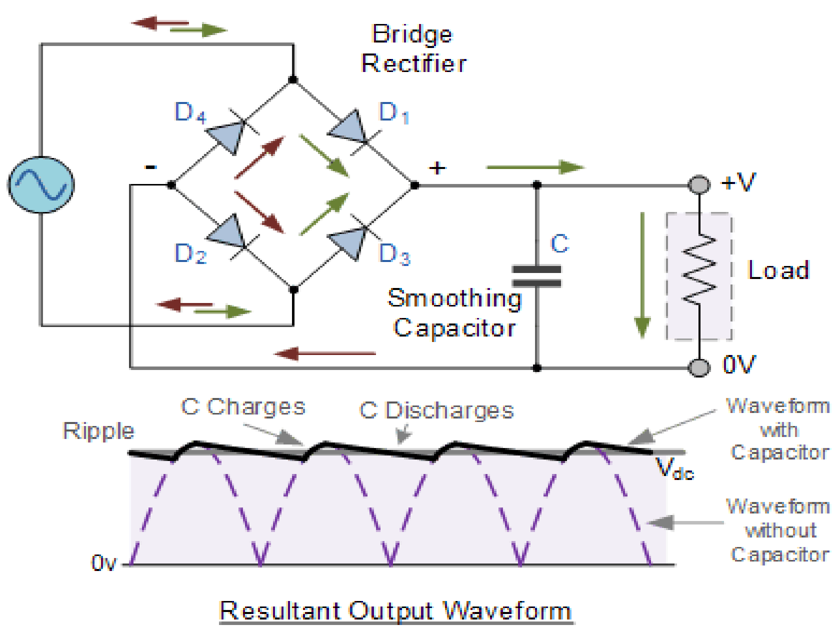


Figure 18 Bridge rectifier

push button

we use push button to rest the microcontroller. A Push Button is a type of switch that works on a simple mechanism called "Push-to-make". Initially, it remains in off state or normally open state but when it is pressed, it allows the current to pass through it or we can say it makes the circuit short when pressed. Normally their body is made up of plastic or metal in some types.

voltage regulator

Its an integral circuit which gives a fixed or constant dc output voltage. the input voltage should be greater than the output voltage. It always should be with two capacitors the first one (c1) for noise emitting and the second one works as smoothing filter ,If the load is big motor in other word if the load needs a large current we should ad transistor above the voltage regulator to make a path for the current to reach the load , but if the load does not need large current maybe the voltage regulator would be enough , so the user should read the datasheet of the regulator . In our project we are using small dc motor so the output current of the voltage regulator would be enough to run the small motor.

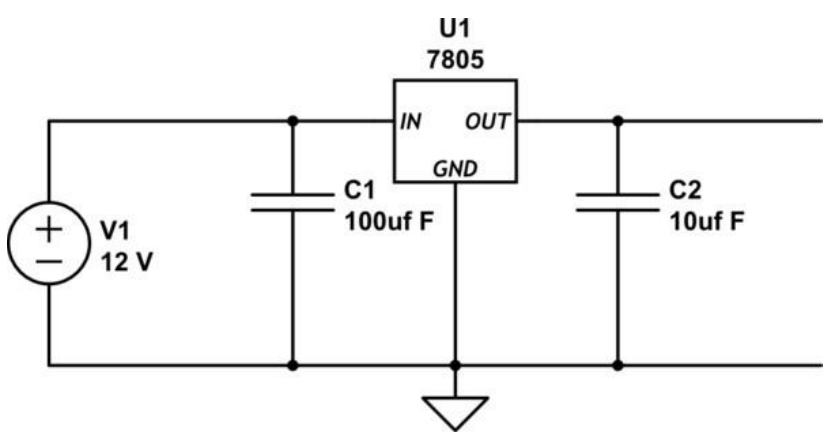


Figure 19 voltage regulator

system diagram (simulation diagram)

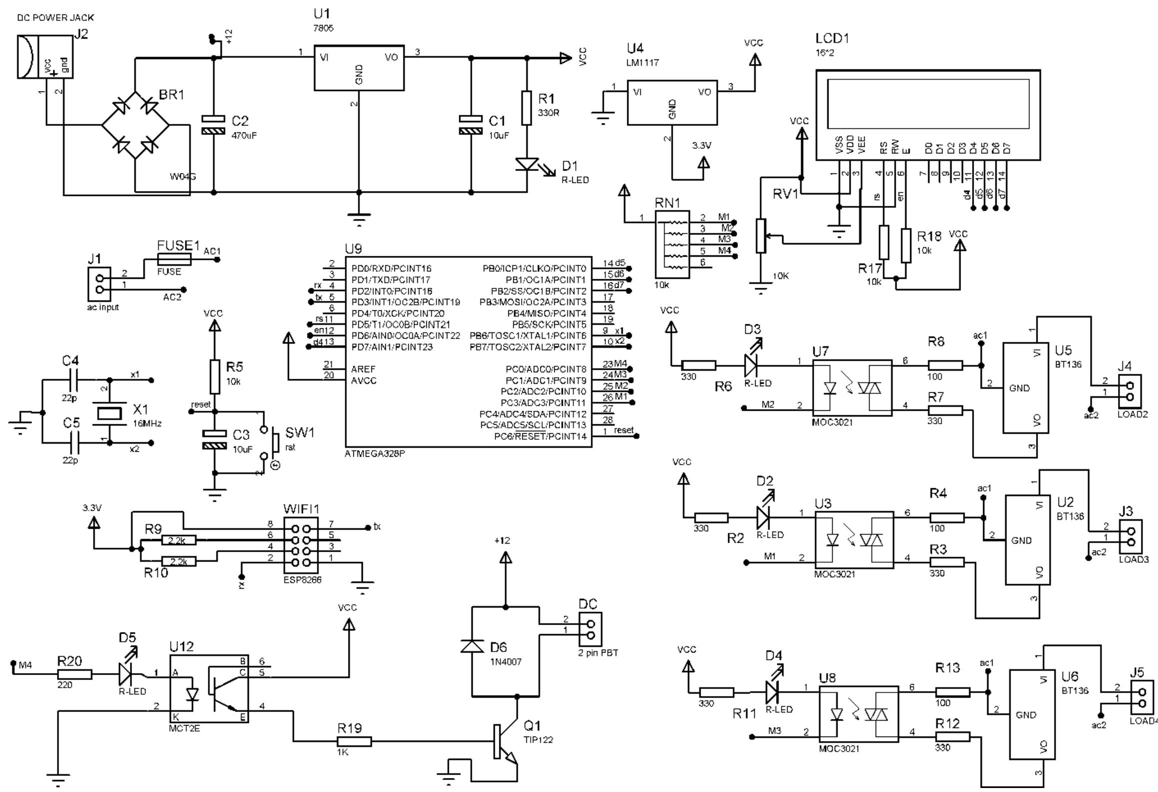


Figure 20 simulation

As the figure shows, we designed our system according to all the information we've mentioned previously in each components' explanation.

The system can control AC loads, and DC load wirelessly by using website page. We used a website that was designed for us (the API page) which we asked them to do. Shown in the next page figure

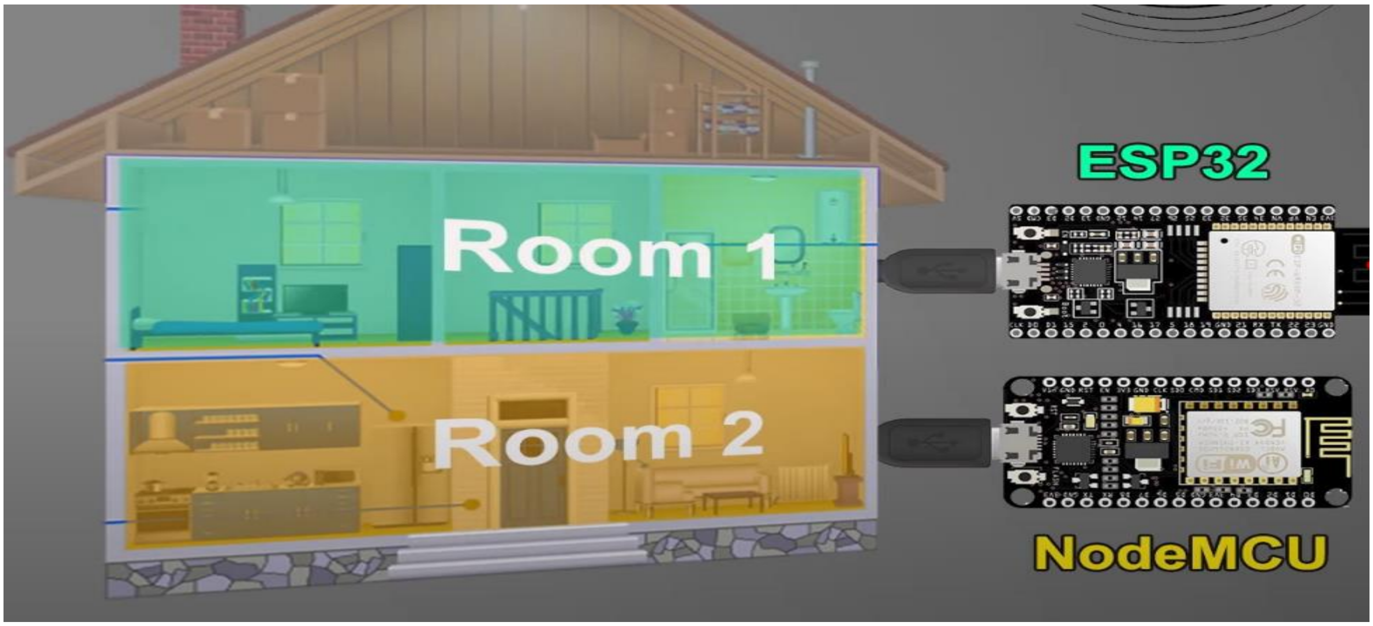
Website & circuit design

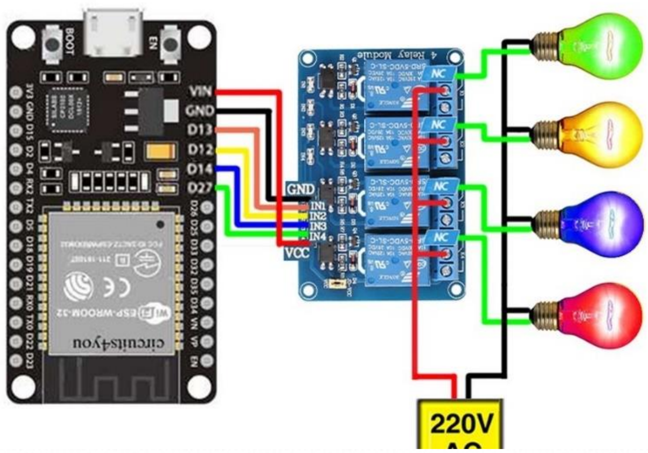
Website design

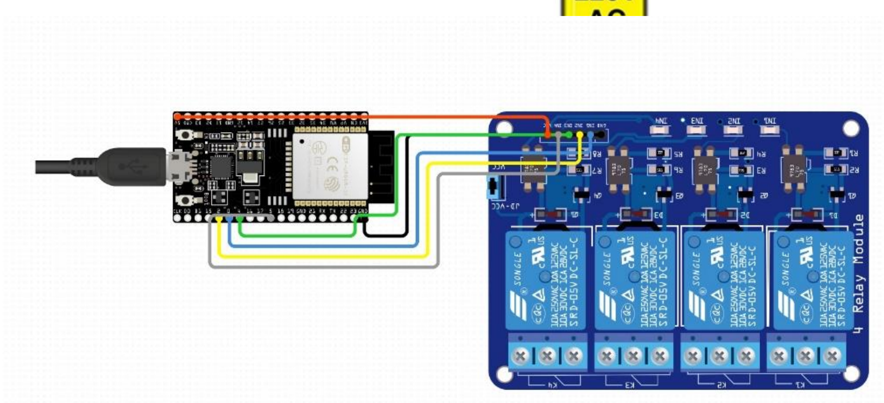
Figure 21 Website design

We used iot gecko website, to design our API by useing their server . IOTgecko website is a private website (need an ID & Password) So , to let the Wi-Fi module directly access to the private website and take the response directly from the API of the private website

Expected circuits:

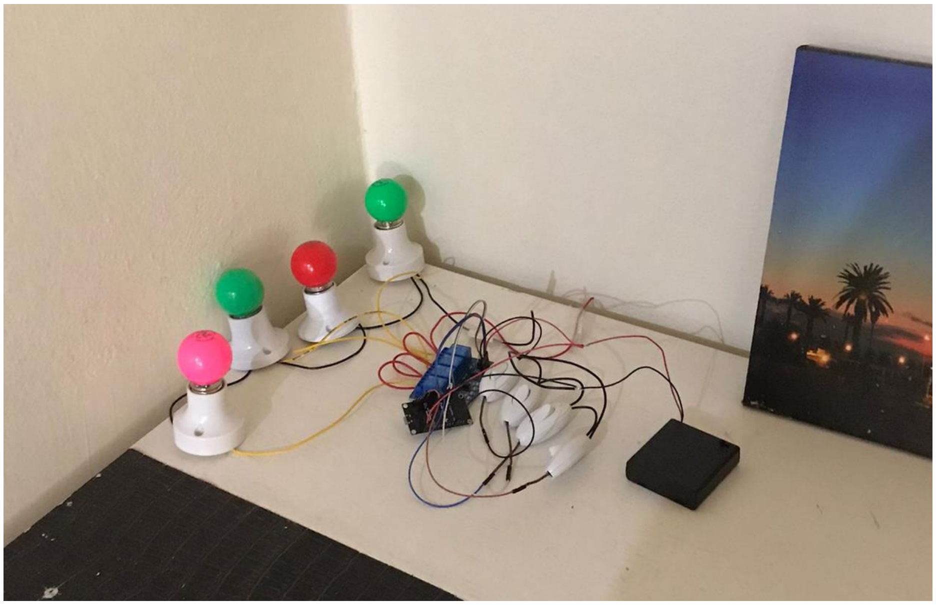






project Prototype:

1- NodeMCU (esp8266) circuit



2- Both circuits

Our Code
Esp32 code:

```
#if defined(ESP32)
#include <WiFi.h>
#include <FirebaseESP32.h>
#elif defined(ESP8266)
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#endif

Initiating Firebase Database;
#define out1 17
#define out2 5
#define out3 18
#define out4 19
#define out5 1

//1. Firebase authentication token.
#define FIREBASE_HOST "espevren2-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "MfXEuyR4XvGNRjwUEIaxFGJDIIlQv0p38hwlOgSa"

//SSID and Password.
#define WIFI_SSID "WIFI NAME"
#define WIFI_PASSWORD "PASSWARD"

void setup() {
// put your setup code here, to run once:
Serial.begin(9600);

// searching to connect to wifi.
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("connecting");
while (WiFi.status() != WL_CONNECTED) {
Serial.print(".");
delay(500);
}
Serial.println();
Serial.print("connected: ");
Serial.println(WiFi.localIP());
// Firebase connenction
// Reconnecting if the network is down
Firebase.reconnectWiFi(true);
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);

pinMode(out1,OUTPUT);
```

```
pinMode(out2,OUTPUT);
pinMode(out3,OUTPUT);
pinMode(out4,OUTPUT);
pinMode(out5,OUTPUT);
}

void loop() {

if(Firebase.getInt(veritabanim, "c1")) //We can use getInt, getBool, getFloat, getDouble or getString according to the data type to be retrieved.
{

//if the connection is successful and data is being sent
Serial.print("String type data retrieval successful, data = ");
Serial.println(veritabanim.stringData());

if (veritabanim.intData()==1){
digitalWrite(out1,LOW);
Serial.print("OUT-ON");
delay(50);
}

else{

digitalWrite(out1,HIGH);
Serial.print("OUT-OFF");
delay(50);
}
}

else{
//prints error message and reason if there is an error

Serial.print("Str data could not be retrieved, ");
Serial.println(veritabanim.errorReason());
}

if(Firebase.getInt(veritabanim, "c2")) //We can use getInt, getBool, getFloat, getDouble or getString according to the data type to be retrieved.
{

//if the connection is successful and data is being sent.
Serial.print("Data retrieval of type int successful, data = ");
Serial.println(veritabanim.intData());

if (veritabanim.intData()==1){
digitalWrite(out2,LOW);
Serial.print("OUT-ON ");
delay(50);
```

```
}

else {

digitalWrite(out2,HIGH);
Serial.print("OUT-OFF  ");
delay(50);
}
}

else{
//prints error message and reason if there is an error

Serial.print("Failed to retrieve int data, ");
Serial.println(veritabanim.errorReason());
}

if(Firebase.getInt(veritabanim, "c3")) //We can use getInt, getBool, getFloat, getDouble or getString according to the data type to be retrieved.
{

//if the connection is successful and data is being sent.
Serial.print("Data retrieval of type int successful, data = ");
Serial.println(veritabanim.intData());

if (veritabanim.intData()==1){
digitalWrite(out3,LOW);
Serial.print("OUT-ON-3");
delay(50);
}

else{

digitalWrite(out3,HIGH);
Serial.print("OUT-OFF-3");
delay(50);
}
}

else{
//prints error message and reason if there is an error

Serial.print("int verisi çekilemedi, ");
Serial.println(veritabanim.errorReason());
}

if(Firebase.getInt(veritabanim, "c4")) //We can use getInt, getBool, getFloat, getDouble or getString according to the data type to be retrieved.
{
```

```
//if the connection is successful and data is being sent.
Serial.print("Data retrieval of type int successful, data = ");
Serial.println(veritabanim.intData());

if (veritabanim.intData()==1){
digitalWrite(out4,LOW);
Serial.print("OUT-ON ");
delay(50);
}

else{

digitalWrite(out4,HIGH);
Serial.print("OUT-OFF  ");
delay(50);
}
}

else{
//prints error message and reason if there is an error.

Serial.print("Failed to retrieve int data, ");
Serial.println(veritabanim.errorReason());
}

if(Firebase.getInt(veritabanim, "c5")) //We can use getInt, getBool, getFloat, getDouble or getString according to the data type to be retrieved.
{

//if the connection is successful and data is being sent.
Serial.print("Data retrieval of type int successful, data = ");
Serial.println(veritabanim.intData());

if (veritabanim.intData()==1){
digitalWrite(out5,LOW);
Serial.print("OUT-ON ");
delay(50);
}

else{

digitalWrite(out5,HIGH);
Serial.print("OUT-OFF  ");
delay(50);
}

}else{
//prints error message and reason if there is an error.
```

```
Serial.print("Str data could not be retrieved, ");
Serial.println(veritabanim.errorReason());
}
}
```

NodeMCU code (ESP8266):

```
#define BLYNK_PRINT Serial
#include <BlynkSimpleEsp8266.h>

// defining the GPIO connected with Relays and switches
#define RelayPin1 5  //D1
#define RelayPin2 4  //D2
#define RelayPin3 14 //D5
#define RelayPin4 12 //D6

#define SwitchPin1 10  //SD3
#define SwitchPin2 0   //D3
#define SwitchPin3 13  //D7
#define SwitchPin4 3   //RX

#define wifiLed   16   //D0

#define VPIN_BUTTON_1    V1
#define VPIN_BUTTON_2    V2
#define VPIN_BUTTON_3    V3
#define VPIN_BUTTON_4    V4

int toggleState_1 = 1; //Define integer to remember the toggle state for relay 1
int toggleState_2 = 1; //Define integer to remember the toggle state for relay 2
int toggleState_3 = 1; //Define integer to remember the toggle state for relay 3
int toggleState_4 = 1; //Define integer to remember the toggle state for relay 4

int wifiFlag = 0;
```

Figure 32 code part1

```
#define AUTH "bwBlqQk7B-_Ci9tHXamjLmLY9JgxipJ-"                 // You should get Auth Token in the Blynk App.
#define WIFI_SSID "Omar Shirwa's iphone"             //Enter Wifi Name
#define WIFI_PASS "aaaaaaaa"         //Enter wifi Password

BlynkTimer timer;

void relayOnOff(int relay){

    switch(relay){
      case 1:
             if(toggleState_1 == 1){
              digitalWrite(RelayPin1, LOW); // turn on relay 1
              toggleState_1 = 0;
              Serial.println("Device1 ON");
              }
             else{
              digitalWrite(RelayPin1, HIGH); // turn off relay 1
              toggleState_1 = 1;
              Serial.println("Device1 OFF");
              }
             delay(100);
      break;
      case 2:
             if(toggleState_2 == 1){
              digitalWrite(RelayPin2, LOW); // turn on relay 2
              toggleState_2 = 0;
              Serial.println("Device2 ON");
              }
             else{
              digitalWrite(RelayPin2, HIGH); // turn off relay 2
              toggleState_2 = 1;
              Serial.println("Device2 OFF");
              }
             delay(100);
      break;
      case 3:
             if(toggleState_3 == 1){
              digitalWrite(RelayPin3, LOW); // turn on relay 3
              toggleState_3 = 0;
              Serial.println("Device3 ON");
              }
             else{
              digitalWrite(RelayPin3, HIGH); // turn off relay 3
              toggleState_3 = 1;
              Serial.println("Device3 OFF");
              }
             delay(100);
      break;
```

Figure 33 code part2

```
      case 4:
             if(toggleState_4 == 1){
              digitalWrite(RelayPin4, LOW); // turn on relay 4
              toggleState_4 = 0;
              Serial.println("Device4 ON");
              }
             else{
              digitalWrite(RelayPin4, HIGH); // turn off relay 4
              toggleState_4 = 1;
              Serial.println("Device4 OFF");
              }
             delay(100);
      break;
      default : break;
      }

}

void with_internet(){
    //Manual Switch Control
    if (digitalRead(SwitchPin1) == LOW){
      delay(200);
      relayOnOff(1);
      Blynk.virtualWrite(VPIN_BUTTON_1, toggleState_1);   // Update Button Widget
    }
    else if (digitalRead(SwitchPin2) == LOW){
      delay(200);
      relayOnOff(2);
      Blynk.virtualWrite(VPIN_BUTTON_2, toggleState_2);   // Update Button Widget
    }
    else if (digitalRead(SwitchPin3) == LOW){
      delay(200);
      relayOnOff(3);
      Blynk.virtualWrite(VPIN_BUTTON_3, toggleState_3);   // Update Button Widget
    }
    else if (digitalRead(SwitchPin4) == LOW){
      delay(200);
      relayOnOff(4);
      Blynk.virtualWrite(VPIN_BUTTON_4, toggleState_4);   // Update Button Widget
    }
}
```

Figure 34 code part3

```
void without_internet(){
    //Manual Switch Control
    if (digitalRead(SwitchPin1) == LOW){
      delay(200);
      relayOnOff(1);
    }
    else if (digitalRead(SwitchPin2) == LOW){
      delay(200);
      relayOnOff(2);
    }
    else if (digitalRead(SwitchPin3) == LOW){
      delay(200);
      relayOnOff(3);
    }
    else if (digitalRead(SwitchPin4) == LOW){
      delay(200);
      relayOnOff(4);
    }
}

BLYNK_CONNECTED() {
  // Request the latest state from the server
  Blynk.syncVirtual(VPIN_BUTTON_1);
  Blynk.syncVirtual(VPIN_BUTTON_2);
  Blynk.syncVirtual(VPIN_BUTTON_3);
  Blynk.syncVirtual(VPIN_BUTTON_4);
}

// When App button is pushed - switch the state

BLYNK_WRITE(VPIN_BUTTON_1) {
  toggleState_1 = param.asInt();
  digitalWrite(RelayPin1, toggleState_1);
}

BLYNK_WRITE(VPIN_BUTTON_2) {
  toggleState_2 = param.asInt();
  digitalWrite(RelayPin2, toggleState_2);
}

BLYNK_WRITE(VPIN_BUTTON_3) {
  toggleState_3 = param.asInt();
  digitalWrite(RelayPin3, toggleState_3);
}
```

Figure 35 code part4

```
BLYNK_WRITE(VPIN_BUTTON_4) {
  toggleState_4 = param.asInt();
  digitalWrite(RelayPin4, toggleState_4);
}


void checkBlynkStatus() { // called every 3 seconds by SimpleTimer

  bool isconnected = Blynk.connected();
  if (isconnected == false) {
    wifiFlag = 1;
    digitalWrite(wifiLed, HIGH); //Turn off WiFi LED
  }
  if (isconnected == true) {
    wifiFlag = 0;
    digitalWrite(wifiLed, LOW); //Turn on WiFi LED
  }
}
```

Figure 36 code part5

```
void setup()
{
  Serial.begin(9600);

  pinMode(RelayPin1, OUTPUT);
  pinMode(RelayPin2, OUTPUT);
  pinMode(RelayPin3, OUTPUT);
  pinMode(RelayPin4, OUTPUT);

  pinMode(wifiLed, OUTPUT);

  pinMode(SwitchPin1, INPUT_PULLUP);
  pinMode(SwitchPin2, INPUT_PULLUP);
  pinMode(SwitchPin3, INPUT_PULLUP);
  pinMode(SwitchPin4, INPUT_PULLUP);

  //During Starting all Relays should TURN OFF
  digitalWrite(RelayPin1, toggleState_1);
  digitalWrite(RelayPin2, toggleState_2);
  digitalWrite(RelayPin3, toggleState_3);
  digitalWrite(RelayPin4, toggleState_4);

  WiFi.begin(WIFI_SSID, WIFI_PASS);
  timer.setInterval(3000L, checkBlynkStatus); // check if Blynk server is connected every 3 seconds
  Blynk.config(AUTH);
}

void loop()
{
  if (WiFi.status() != WL_CONNECTED)
  {
    Serial.println("WiFi Not Connected");
  }
  else
  {
    Serial.println("WiFi Connected");
    Blynk.run();
  }

  timer.run(); // Initiates SimpleTimer
  if (wifiFlag == 0)
    with_internet();
  else
    without_internet();
}
```

Figure 37 code last part

future design Using nodeMCU Wi-Fi module

Blynk Authentication

When your app needs to know the identity of the user in order to provide distinctive data to that user, a form of secure sign-in is necessary. Building and maintaining sign-in infrastructure is a difficult and expensive proposition. From a user perspective, giving credentials and personal data to an app can also be a user experience bump - potentially causing them to reject using it. Thus, apps that allow users to sign in with known credentials, such as those provided by Google, Facebook, and Twitter, are becoming more popular. With these issues and trends in mind, Blynk Authentication has been built to give you an easy API that will allow you to use sign-in from federated providers, a simple email/password scheme, or integrate with any existing authentication back-ends you own. It integrates with Blynk services such as the Realtime database. ,in the node mcu we could directly use Blynk library to read all the values from the database and as we will show in the video the response of it is very fast .

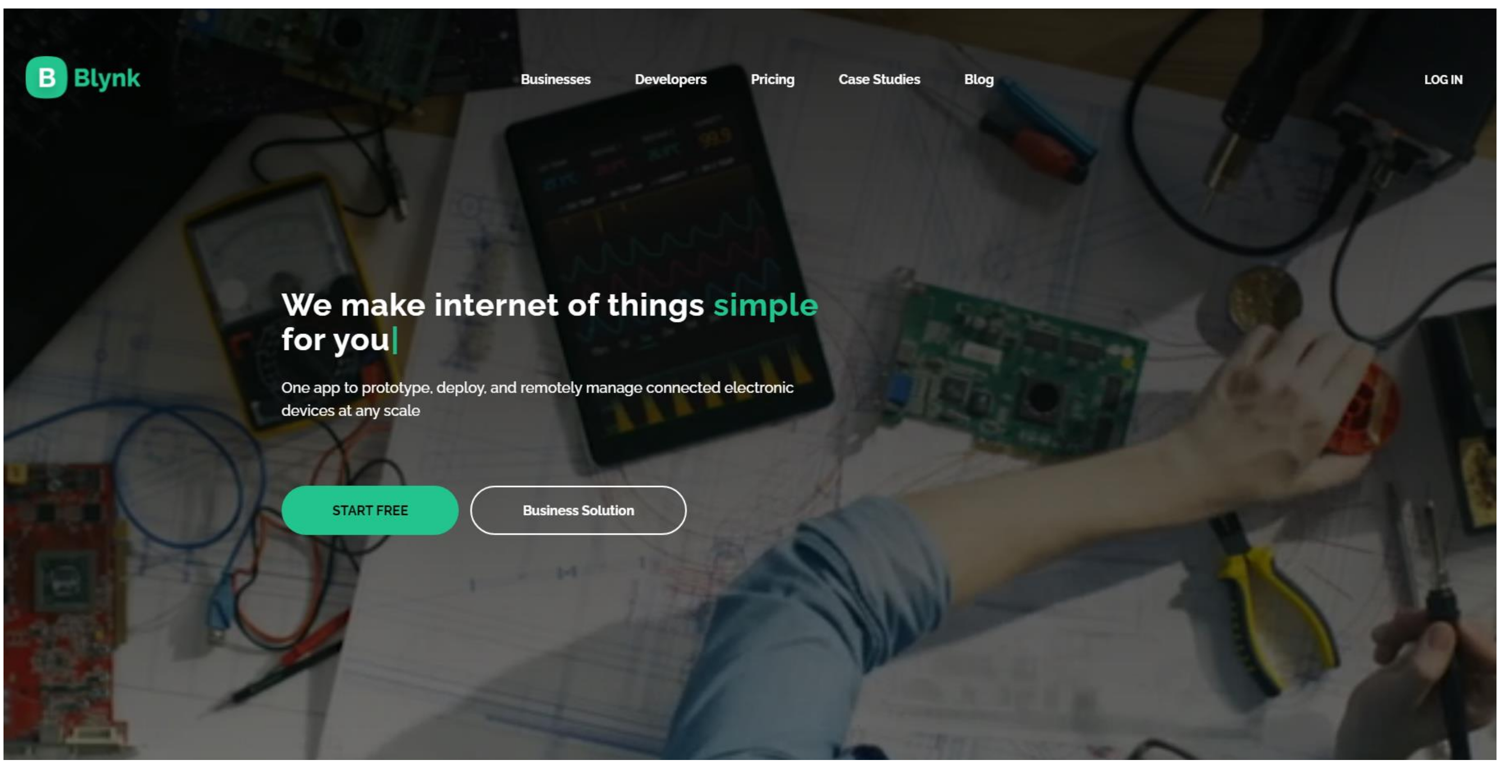


Figure 38 Blynk

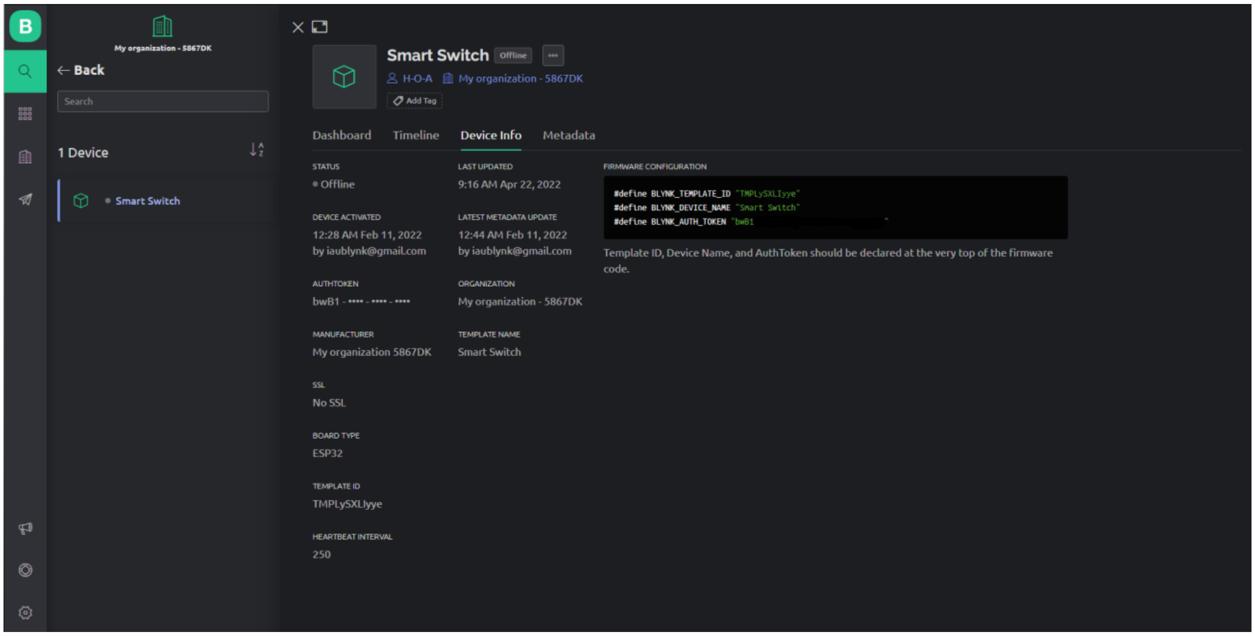


Figure 39 Blynk Authentication and device configurations

Blynk also provides an authentication tokenshown in the figure. to use it, connect your app to Firebase and it allows you to upload information in the data base

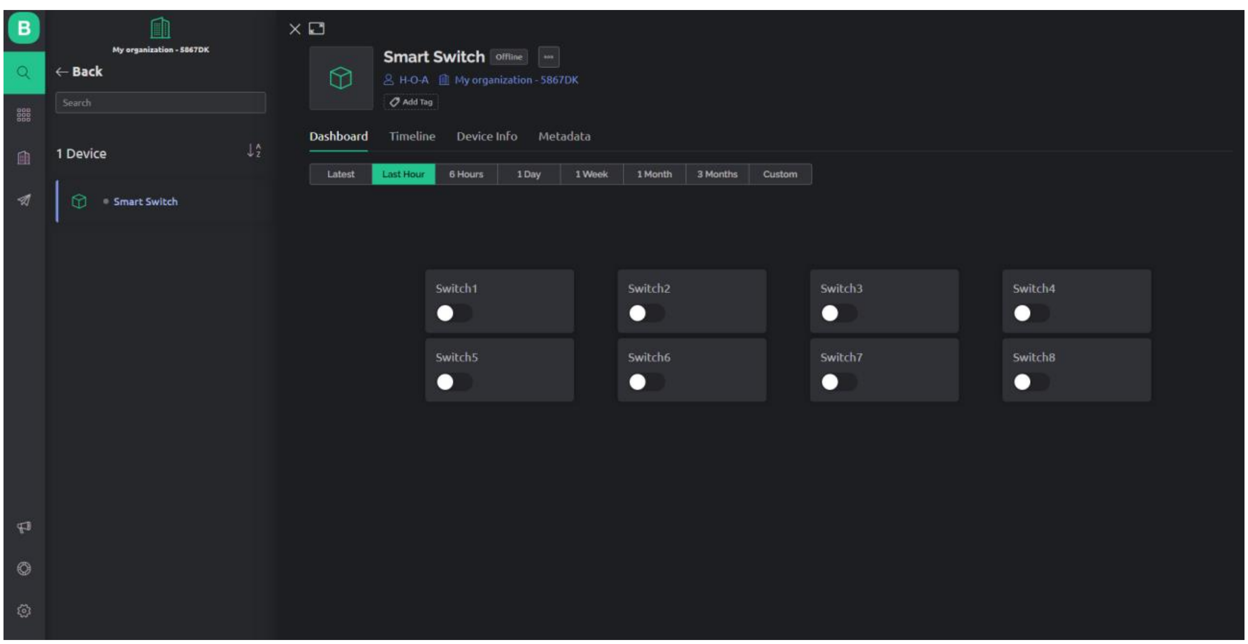
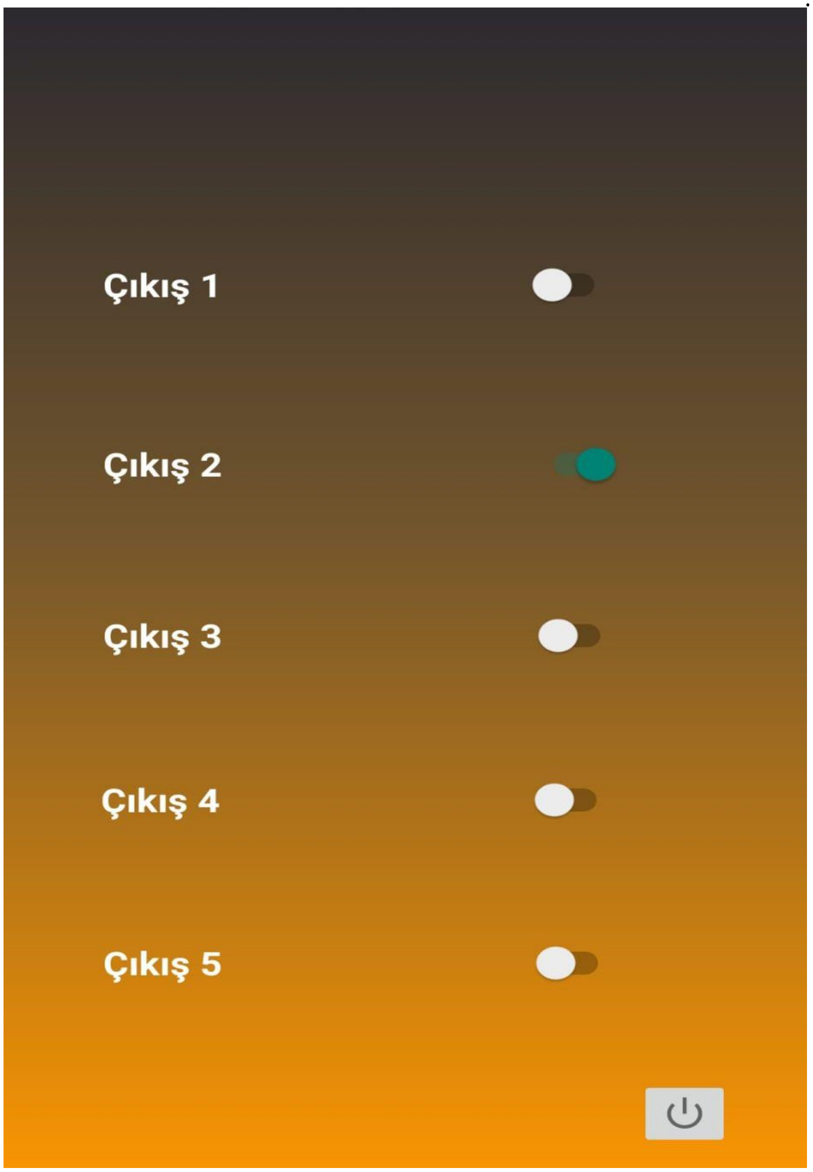
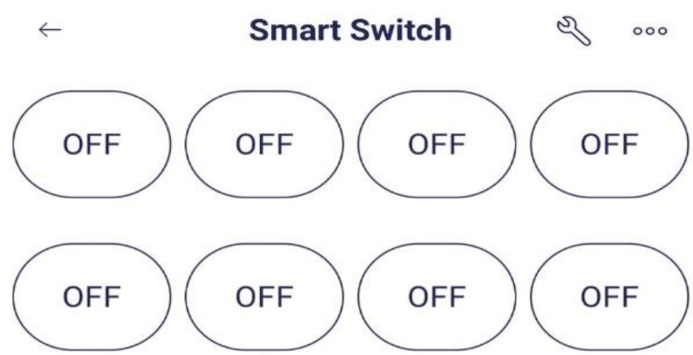


Figure 40 Blynk Project Dashboard

Applications





References

Internet of Things Principles and Paradigms by rajkumar buyya

(buyya, 2016)

the definitive guide to firebase: build android apps on google's mobile platform

(Moroney, 2017)

Smart Home Systems Based on Internet of Things

(Domb, 2019)

industrial automation hands on

(lamb, 2013)

internet-of-things (iot) systems architectures algorithms methodologies

(Serpanos, Dimitrios, Wolf, Marilyn Claire,2017)

Designing the Internet of Things (Adrian McEwen, 2013)

Kolban's Book on ESP8266 (Kolban, 2016)