**SMART PHONE BASED ACCESS CONTROL SYSTEM**

**ABSTRACT:**

This project is a combination of mobile technology and embedded systems in which the user can control the LED of the photon using android mobile. User can send commands through the application installed on the mobile handset. Wi-Fi technology is used to communicate. It is a wireless controlling technique. A particle photon, IFTTT and IFTTT Do Button is used in the project to turn the LED on and off.

**REQUIREMENTS:**

1. HARDWARE:

* Particle Photon
* Android tablet
* LED
* Bread Board

2. SOFTWARE:

* IFTTT Do Button App
* IFTTT App
* An account over at IFTTT Particle IDE

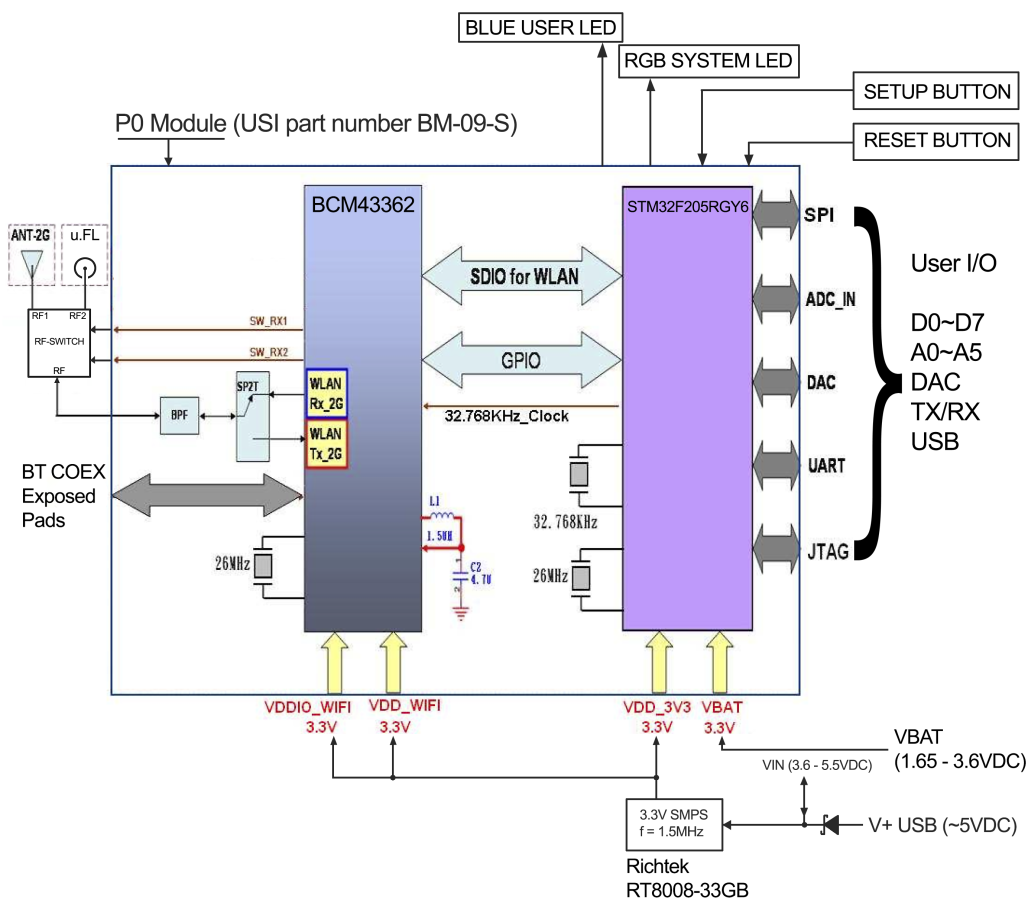
**SYSTEM DESCRIPTION:**

This project is a combination of mobile technology and embedded systems. User can control the LED of photon using android mobile. An application has to be installed on his/her Android mobile handset to toggle the LED of the photon. User can send commands using that application.

This project consists of a particle photon, IFTTT and IFTTT DO BUTTON, a project that will make the LED of a photon to turn on and off when receiving a signal from an IFTTT Do Button.

IFTTT is a free web-based service that allows user to create chains of simple conditional statements, called “recipes”, which are triggered based on changes to other web services. It connects different Internet services together, so you can get one service. IFTTT is an abbreviation of “If This Then That”. An example recipe might consist of sending an email message if the IFTTT user tweets using a certain hashtag. Or , if the user is tagged by someone on Facebook, then the photo will be added to the cloud-based photo archive. Channels are the building blocks of IFTTT , they mainly describe a series of data from a web service . It can also describe some action controlled APIs like SMS. Sometimes, it can represent information in terms of weather and stocks.

The channel which is used in this project is “PARTICLE”. The device used is photon. Triggers are the items that trigger the action. Here we use the LED On/Off trigger . Actions are the output results from the input of the trigger. If LED On trigger is used the LED connected to the D7 pin of the particle photon is written high. If LED Off trigger is used the LED connected to the D7 pin of the particle photon is written low. Recipes are the predicates made from Triggers and Actions. In this project the recipes are created using the IFTTT Do Button . The LED can be controlled with just a tap.

**BLOCK DIAGRAM:** 

Particle cloud server on the internet http://particle.io

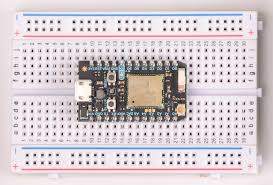
IFTTT Service

http://ifttt.com

Particle photon (Wi-Fi) + sensors

Smartphone

**CIRCUIT DIAGRAM:**



**CODE:**

// Controlling LEDs over the Internet

// First, let's create our "shorthand" for the pins

// led is D7

int led= D7;

// we are also going to register our Spark function

void setup()

{

// Here's the pin configuration

pinMode(led, OUTPUT);

// We are also going to declare a Spark.function so that we can turn the LED on and off from the cloud.

Spark.function("led",ledToggle);

// This is saying that when we ask the cloud for the function "led", it will employ the function ledToggle() from this app.

// For good measure, let's also make sure both LEDs are off when we start:

digitalWrite(led, LOW);

}

// This is the ledToggle function we registered to the "led" Spark.function earlier.

int ledToggle(String command) {

/\* Spark.functions always take a string as an argument and return an integer.

Since we can pass a string, it means that we can give the program commands on how the function should be used.

In this case, telling the function "on" will turn the LED on and telling it "off" will turn the LED off.

Then, the function returns a value to us to let us know what happened.

In this case, it will return 1 for the LEDs turning on, 0 for the LEDs turning off,

and -1 if we received a totally bogus command that didn't do anything to the LEDs.

\*/

if (command=="on") {

digitalWrite(led,HIGH);

return 1;

}

else if (command=="off") {

digitalWrite(led,LOW);

return 0;

}

else if(command=="blink")

//To blink the LED 10 times

{

for(int i=0;i<10;i++)

{

digitalWrite(led,HIGH);

delay(1000);

digitalWrite(led,LOW);

delay(1000);

}

}else{

return -1;

}

}

**CONCLUSION AND CHALLENGES:**

In this project we can control the LED of the particle photon using the IFTTT Do Button. Do Button empowers us to create our own personalised button with just a tap. With Do Button, IFTTT users can trigger actions they want to see in the apps, online services, and smart home devices, saving time and increasing the productivity.

*CHALLENGES:*

* Occasional service hiccups do buttons don’t work flawlessly. No notification of failed commands.
* Requires a certain amount of trails before getting used to the functioning of the Particle Photon.

**TEAM ABD**

Members:

* ANURADHA.H
* BINDU.A
* BINDU.S
* DEEPA.S