Andhra Pradesh State Skill Development Corporation

Skill AP
APSSDC

eSkill AP
Learn Anytime Anywhere
Andhra Pradesh State Skill Development Corporation

INTERNET OF THINGS

(IoT)

LED TOGGLE WITH BUTTON

LED TOGGLE WITH BUTTON

AIM: Our goal in this project is led toggle with button
Software: Arduino IDE
Components Required:

1. System
2. Arduino Uno board-1
3. Arduino dumping cable-1
4. Button
5. 220 ohm resistor
6. Bread board.
7. LED-1

THEORY:

We already learned how to turn a LED on with a button. The problem arises that if the button is unpressed the LED turns off. If you would like to keep the LED on after the button when pressed we need to remember the state of the button. We can use the boolean function for this. As you can see I have added variables to hold the current and new state of the button.

There is a Boolean variable that holds/remember the state of the object, in our case the button. If the program is run for the first time both states should be LOW. We can now transform the button into a switch. An important part of the sketch is to set the oldSwitchState to the current newSwitchState. We will add the below code to change the state of the button. This is important, since we will use it in our if statements. After we declare our pins and our variables we arrive at the void loop() part of our sketch. In the loop function, the sketch checks whether or not the button is pressed. The sketch will read the signal from the button to give the newSwitchState a LOW or HIGH value. HIGH for pressing the button and LOW if the button is not pressed.

We need to know a couple of things to make sure that the button works as a switch. Firstly, we need to check the condition whether the newSwitchState is different from the old Switch State. For that, we use a comparison operator “!=”. Rember, if the button is pressed it sends a HIGH signal if it is not pressed it sends a LOW signal. After the check is completed we need to check if the button is sending a HIGH signal meaning that the button is pressed.

Furthermore, we need to know whether the LEDstatus is HIGH or LOW. In other words, we need to know whether the LED is on or off. If the LEDstatus is LOW and the button is pressed then the program should recognize that we want to turn the LED on.

The last statement that we will use is an Else statement. If the if statements are false, and the button is pressed, we want the LED to turn off.

An important part of the sketch is to set the oldSwitchState to the current newSwitchState. We will add the below code to change the state of the button. This is is important, since we will use it in our if statements.

Code:

```
// Make a Toggle Switch Button
 // Define the pins being used
 int pinLed = 10;
 int pinSwitch = 2;
 // declaring variables to hold the new and old switch states
 boolean oldSwitchState = LOW;
 boolean newSwitchState = LOW;
 boolean LEDstatus = LOW;
 void setup()
 {
 pinMode(pinLed, OUTPUT);
 digitalWrite(pinLed, LOW);
 pinMode(pinSwitch, INPUT);
 }
 void loop()
 {
 newSwitchState = digitalRead(pinSwitch);
 if ( newSwitchState != oldSwitchState )
 {
 // has the button switch been closed?
 if ( newSwitchState == HIGH )
 {
 if ( LEDstatus == LOW ) {
 digitalWrite(pinLed, HIGH);
 LEDstatus = HIGH;
 }
 else {
 digitalWrite(pinLed, LOW);
 LEDstatus = LOW;
 }
 }
 oldSwitchState = newSwitchState;
 }
 }
```

Result: The LED turns on when button is pushed and turned off when the button is pushed for the second time