

## Arduino Multi-State Code

```
#define pushButton1 2 //Push Button for Accessory port side on D2
#define relay1 3      //High Output to Relay for pushButton1 on 3
#define blueHalo1 35  //Low Output to Blue Halo on pushButton1 on 34
#define redHalo1 37   //Low Output to Red Halo on pushButton1 on 36

#define pushButton2 4 //Push Button, Navigation - Anchor
#define relay2 5      //Output to Relay for Navigation Lights
#define relay2point1 6 //Output to Relay for Anchor Light
#define blueHalo2 39  //Output to Blue Halo on pushButton1 on
#define greenHalo2 41 //Output to Green Halo on pushButton1 on
#define whiteHalo2 43 //Output to White Halo on pushButton1 on

#define pushButton3 //Push Button, Courtesy Lights - Accent Lights
#define relay3      //Output to Relay for pushButton3
#define blueHalo3   //Output to Blue Halo on pushButton3 on
#define greenHalo3  //Output to Green Halo on pushButton3 on
#define whiteHalo3  //Output to White Halo on pushButton3 on

#define toggleSwitch4 //Toggle switch for Port Ignition switched from HIGH
#define blueHalo4      //Output to Blue Halo on toggleSwitch4 on
#define redHalo4       //Output to Red Halo on toggleSwitch4 on
```

```
#define toggleSwitch5 //Toggle Switch for Starboard Ignition switched from HIGH
#define blueHalo5      //Output to Blue Halo on toggleSwitch5 on
#define redHalo5       //Output to Red Halo on toggleSwitch5 on
```

```
#define pushButton6 //Push Button, Blower Motors
#define relay6       //Output to Two 10 Amp Relays
#define blueHalo6    //Output to Blue Halo on pushButton6
#define redHalo6     //Output to Red Halo on pushButton6
```

```
#define pushButton7 //Push Button, Docking Lights, Forward - All ON - OFF
#define relay7       //Output to Relay Forward Docking Lights
#define relay7point1 //Output to Relay for Stern Docking Lights
#define blueHalo7    //Output to Blue Halo on pushButton7 on
#define greenHalo7   //Output to Green Halo on pushButton7 on
#define whiteHalo7   //Output to White Halo on pushButton7 on
```

```
#define pushButton8 //Push Button for Accessory Starboard side on
#define relay8       //Output to Relay for pushButton8 on
#define blueHalo8    //Output to Blue Halo on pushButton8 on
#define redHalo8     //Output to Red Halo on pushButton8 on
```

```
int state = 0;
```

```
int old = 0;
int buttonPoll = 0;
```

```
void setup() {
```

```
    pinMode(pushButton1, INPUT_PULLUP); //Push Button set as Input with internal pullup
    pinMode(relay1, OUTPUT);             //Relay1 Set to output to Relay Module
    pinMode(blueHalo1, OUTPUT);          //blueHalo1 set as output to activate Blue Halo on pushButton1
    pinMode(redHalo1, OUTPUT);           //redHalo1 set as output to activate Red Halo on pushButton1
```

```
    digitalWrite(relay1, LOW);           //set initial state of relay1 to OFF
    digitalWrite(blueHalo1, HIGH);       //set initial state of blueHalo1 to ON
    digitalWrite(redHalo1, LOW);         //set initial state of redHalo1 to OFF
```

```
    pinMode(pushButton2, INPUT_PULLUP); //Push Button set as Input with internal pullup
    pinMode(relay2, OUTPUT);             //Relay2 Set to output to Relay Module
    pinMode(relay2point1, OUTPUT);
    pinMode(blueHalo2, OUTPUT);          //blueHalo2 set as output to activate Blue Halo on pushButton2
    pinMode(greenHalo2, OUTPUT);         //whiteHalo2 set as output to activate Red Halo on pushButton2
    pinMode(whiteHalo2, OUTPUT);
```

```
    digitalWrite(relay2, LOW);           //set initial state of relay2 to OFF
    digitalWrite(relay2point1, LOW);
```

```
digitalWrite(blueHalo2, HIGH); //set initial state of blueHalo2 to ON
digitalWrite(greenHalo2, LOW); //set initial state of greenHalo2 to OFF
digitalWrite(whiteHalo2, LOW); //set initial state of whiteHalo2 to OFF
}

void loop() {
  buttonPoll = digitalRead(pushButton1);
  if (buttonPoll == 1) {
    delay(50);
    buttonPoll = digitalRead(pushButton1);
    if (buttonPoll == 0) {
      state = old + 1;
    }
  } else {
    delay(100);
  }
  switch (state) {
    case 1:
      digitalWrite(redHalo1, HIGH);
      digitalWrite(blueHalo1, LOW);
      digitalWrite(relay1, HIGH);
      old = state;
      break;
    default:
      digitalWrite(redHalo1, LOW);
      digitalWrite(blueHalo1, HIGH);
      digitalWrite(relay1, LOW);
      old = 0;
      break;
  }
}
```

```
}

{
  buttonPoll = digitalRead(pushButton2);
  if (buttonPoll == 1) {
    delay(50);
    buttonPoll = digitalRead(pushButton2);
    if (buttonPoll == 0) {
      state = old + 1;
    }
  } else {
    delay(100);
  }
  switch (state) {
    case 1:
      digitalWrite(blueHalo2, LOW);
      digitalWrite(greenHalo2, HIGH);
      digitalWrite(whiteHalo2, LOW);
      digitalWrite(relay2, HIGH);
      digitalWrite(relay2point1, HIGH);
      old = state;
      break;
    case 2:
      digitalWrite(blueHalo2, LOW);
      digitalWrite(greenHalo2, LOW);
      digitalWrite(whiteHalo2, HIGH);
      digitalWrite(relay2, LOW);
      digitalWrite(relay2point1, HIGH);
  }
}
```

```
    old = state;
    break;
default:
    digitalWrite(blueHalo2, HIGH);
    digitalWrite(greenHalo2, LOW);
    digitalWrite(whiteHalo2, LOW);
    digitalWrite(relay2, LOW);
    digitalWrite(relay2point1, LOW);
    old = 0;
    break;
}
}
}
```



<code>#define button 3</code>	<code>//pushbutton on D3</code>
<code>#define redLED 5</code>	<code>//red LED on D5</code>
<code>#define greenLED 6</code>	<code>//green LED on D6</code>
<code>#define yellowLED 7</code>	<code>//yellow LED on D7</code>
<code>int state = 0;</code>	<code>//integer to hold current state</code>
<code>int old = 0;</code>	<code>//integer to hold last state</code>
<code>int buttonPoll = 0;</code>	<code>//integer to hold button state</code>
<code>void setup() {</code>	
<code>pinMode(button, INPUT_PULLUP);</code>	<code>//button as input</code>
<code>pinMode(redLED, OUTPUT);</code>	<code>//LEDs as output</code>
<code>pinMode(greenLED, OUTPUT);</code>	
<code>pinMode(yellowLED, OUTPUT);</code>	
<code>digitalWrite(redLED, LOW);</code>	<code>//set initial state as off</code>
<code>digitalWrite(greenLED, LOW);</code>	<code>//set initial state as off</code>
<code>digitalWrite(yellowLED, LOW);</code>	<code>//set initial state as off</code>
<code>}</code>	
<code>void loop() {</code>	
<code>buttonPoll = digitalRead(button);</code>	
<code>if(buttonPoll == 1){</code>	
<code>delay(50);</code>	
<code>buttonPoll = digitalRead(button);</code>	
<code>if(buttonPoll == 0){</code>	



```
    state = old + 1;
  }}
  else{
    delay(100);
  }
  switch(state){
    case 1:
      digitalWrite(redLED, HIGH);
      digitalWrite(greenLED, LOW);
      digitalWrite(yellowLED, LOW);
      old = state;
      break;
    case 2:
      digitalWrite(redLED, LOW);
      digitalWrite(greenLED, HIGH);
      digitalWrite(yellowLED, LOW);
      old = state;
      break;
    case 3:
      digitalWrite(redLED, LOW);
      digitalWrite(greenLED, LOW);
      digitalWrite(yellowLED, HIGH);
      old = state;
      break;
    default:
      digitalWrite(redLED, LOW);
      digitalWrite(greenLED, LOW);
      digitalWrite(yellowLED, HIGH);
      old = 0;
      break;
  }
```

```
}  
}
```

```
#define pushButton1 2           //Push Button for Accessory port side on D2  
#define relay1 3               //High Output to Relay for pushButton1 on 0  
#define blueHalo1 35           //Low Output to Blue Halo on pushButton1 on 34  
#define redHalo1 37            //Low Output to Red Halo on pushButton1 on 36
```

```
int state = 0;  
int old = 0;  
int buttonPoll = 0;
```

```
void setup() {
```

```
    pinMode(pushButton1, INPUT_PULLUP);           //Push Button set as Input with internal pullup  
    pinMode(relay1, OUTPUT);                       //Relay1 Set to output to Relay Module  
    pinMode(blueHalo1, OUTPUT);                    //blueHalo1 set as output to activate Blue Halo on  
pushButton1  
    pinMode(redHalo1, OUTPUT);                     //redHalo1 set as output to activate Red Halo on  
pushButton1
```

```

digitalWrite(relay1, LOW);           //set initial state of realy1 to OFF
digitalWrite(blueHalol, HIGH);       //set initial state of blueHalol to OFF
digitalWrite(redHalol, HIGH);        //set initial state of redHalol to OFF

}

void loop() {
    buttonPoll = digitalRead(pushButton1);
    if(buttonPoll == 1){
        delay(50);
        buttonPoll = digitalRead(pushButton1);
        if(buttonPoll == 0){
            state = old + 1;
        }
    }
    else{
        delay(100);
    }
    switch (state) {
        case 1:
            digitalWrite(redHalol, LOW);
            digitalWrite(blueHalol, HIGH);
            digitalWrite(relay1, HIGH);
            old = state;
            break;
        default:
            digitalWrite(redHalol, HIGH);
            digitalWrite(blueHalol, HIGH);
    }
}

```

```
    digitalWrite(relay1, LOW);
    old = 0;
    break;
}
}
```

```
#define pushButton1 2 //Push Button for Accessory port side on D2
#define relay1 3 //High Output to Relay for pushButton1 on 3
#define blueHalo1 35 //Low Output to Blue Halo on pushButton1 on 34
#define redHalo1 37 //Low Output to Red Halo on pushButton1 on 36
```

```
#define pushButton2 4 //Push Button, Navigation - Anchor
#define relay2 5 //Output to Relay for Navigation Lights
#define relay2point1 6 //Output to Relay for Anchor Light
#define blueHalo2 39 //Output to Blue Halo on pushButton1 on
#define greenHalo2 41 //Output to Green Halo on pushButton1 on
#define whiteHalo2 43 //Output to White Halo on pushButton1 on
```