

Wifi Power Plug



SUMMARY:

- I. Presentation of Semi.
- II. Objective.
- III. Technical context.
- IV. Implementation.
- V. Conclusion.

SEMI



Electronics and Microelectronics Unit at the University of Mons

Head of Unit : - Professor Carlos Valderrama.

Academic team: - Professor Carlos Valderrama.

- Professor Carlos Dualibe.

Technical team : - Nathalie Durieux, Secretary.

- Daniel Binon, Electronic technician.

- Alexandre Quenon , Teaching and research assistant.

- William Van Hoeck, IT administrator.

3



Objective:



Develop a user interface to control energy output via wifi.

4

User interface

An html page where the user configures internet parameters, dimming values and can read messages from the processor.



5



THE PROCESS :

first

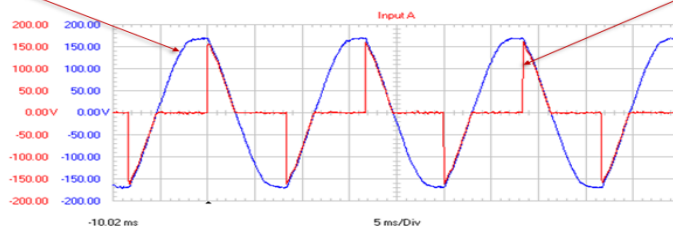
second

last

ZERO DETECTOR

PROCESSOR

POWER OUTPUT



6



STEPS:

- 1) Generate a dimmer code on Arduino.
- 2) Configure the ESP8266 protocol connection.
- 3) Elaborate the code for Multichannel management.
- 4) Bridge the ESP8266 to a web page or an application.

7



IMPLEMENTATION

Plan :

- ❖ Presentation of the tools.
- ❖ Developement of user interface.
- ❖ Presentation of the tutorial .

8

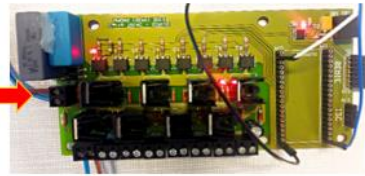
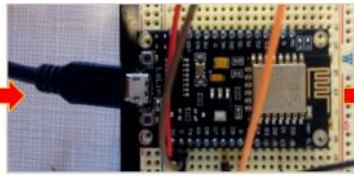


PRESENTATION OF THE TOOLS :

ZERO DETECTOR

PROCESSOR

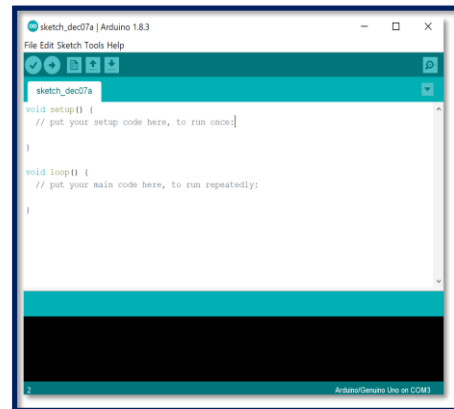
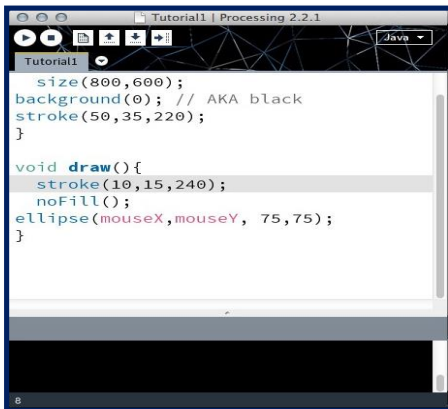
POWER OUTPUT



9

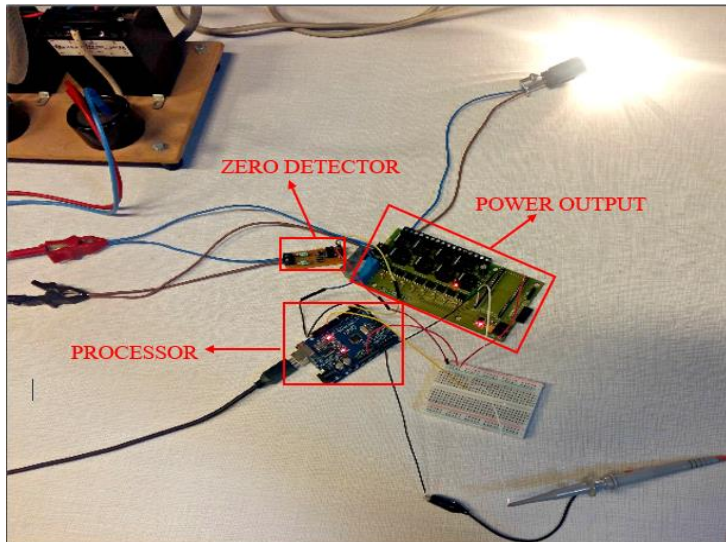


PRESENTATION OF TOOLS: Arduino IDE & Processing



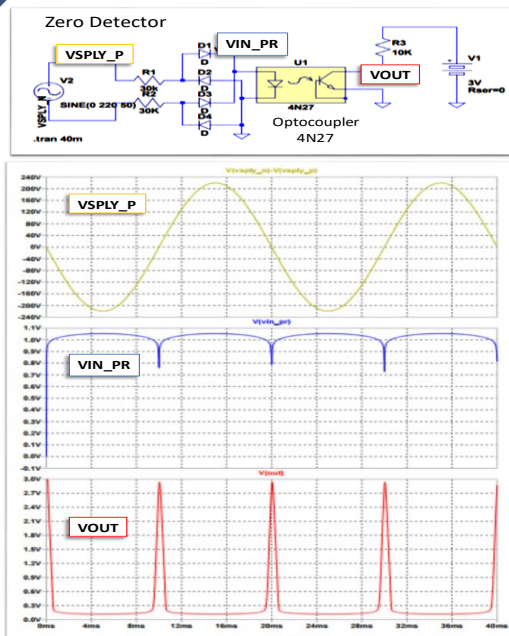
10

OVERVIEW:



11

ZERO DETECTOR:



Simulation of the circuit on LTspice that draws power and optocoupler input as well as general output .

12

PROCESSOR:

Connection to the server and communicate with websockets

```
Serial.begin(115200);
for(int i=0; i < N_Channel; i++) {
  pinMode(AC_pin[i], OUTPUT);    // Set the Triac pin as output
}
WiFi.begin(ssid, password);

while(WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
  delay(200);
}

Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

attachInterrupt(12, zero_cross_detect, RISING); // Attach an Interrupt to Pin 12 (interrupt 0) for Zero Cross Detection

timer1_attachInterrupt(dim_check);
timer1_enable(TIM_DIV16, TIM_EDGE, TIM_SINGLE);
timer1_write(freqStep);

delay(300);

Serial.println("Start Websocket Server");
websocket.begin();
websocket.onEvent(websocketEvent);
}
```

13

PROCESSOR:

```
void zero_cross_detect() {          // Network zero crossing management routine (interrupt)
  zero_cross = true;                // Set the boolean to true to tell our dimming function that a zero cross has occurred
  for(int i=0; i < N_Channel; i++) {
    digitalWrite(AC_pin[i], HIGH); // Turn on TRIAC (and AC)
    counter[i]=0;                   // Re initialize the counter for each channel
  }
}
```

Interruption that alarms the processor that a zero cross has occurred .

14

PROCESSOR

```
void Triac_Dim(int idout, int level) {

    dim[idout-1] = map(level, 0, 100, 0, 128);
    //dim = level*1.28;
    Serial.print("Sortie: ");
    Serial.print(idout);
    Serial.print(" - Niveau dimmer: ");
    Serial.print(level);
    Serial.print(" - var. dim: ");
    Serial.println(dim[idout-1]);
}
```

- Function to converge the user's dimming value into processor's language .

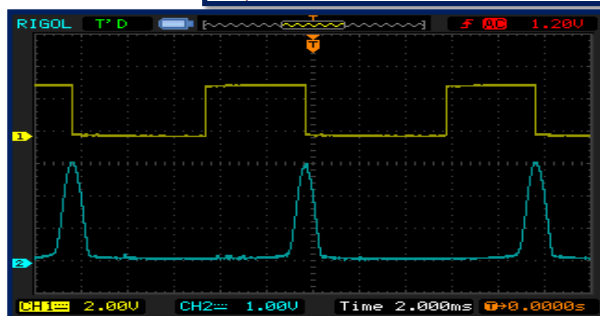
15

PROCESSOR :

- Interruption to turn the triac on at appropriate dimming level after a zero cross detect .

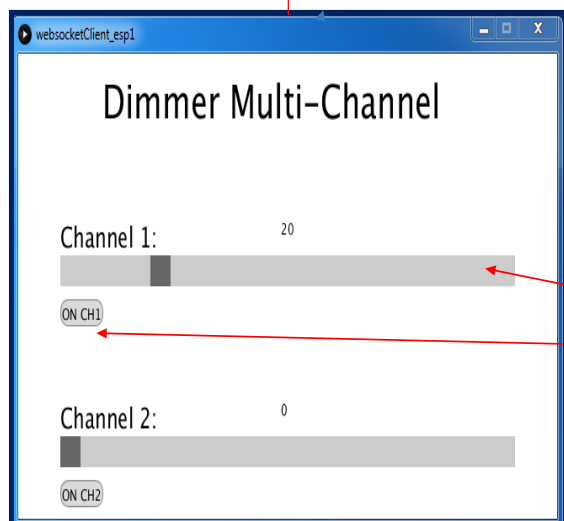
```
void ICACHE_RAM_ATTR dim_check() {
    static unsigned int zccount=N_Channel;

    if(zero_cross == true) {
        for(int i=0; i < 2; i++) {
            if(counter[i]>=dim[i]) {
                digitalWrite(AC_pin[i], LOW); // turn on light
                counter[i]=0; // reset time step counter
                //zero_cross = false; //reset zero cross detection
                zccount--;
            }
            else {
                counter[i]++; // increment time step counter
            }
        }
        if (zccount==0) {
            zero_cross = false; //reset zero cross detection
            zccount = N_Channel;
        }
    }
}
```



16

GRAPHICS GENERATOR



```

websocketClient_esp12
1 import websockets.*;
2
3 WebSocketClient wsc;
4 int now;
5 int oldValue1=0, oldValue2=0;
6
7 HScrollbar hs1, hs2; // Two scrollbars
8 Button bt1, bt2;
9
10 void setup(){
11   size(640,360);
12   noStroke();
13
14   wsc= new WebSocketClient(this, "ws://192.168.10.100:81/");
15   now=millis();
16
17   hs1 = new HScrollbar(50, height/2-12, width-100, 24, 16);
18   hs2 = new HScrollbar(50, height/2+128, width-100, 24, 16);
19   bt1 = new Button("ON CH1",50, 190, 50, 20);
20   bt2 = new Button("ON CH2",50, 330, 50, 20);
21
22 }
23

```

17

```

float value1 = hs1.getPos();
float value2 = hs2.getPos();
int value_1 = (int)map(value1,53,591,0,100);
int value_2 = (int)map(value2,53,591,0,100);
text(value_1,312,140);
text(value_2,312,280);

if (oldvalue1 != value_1){
  oldValue1 = value_1;
  String message_1 = "led"+str(1)+" "+str(value_1);
  println(message_1);
  wsc.sendMessage(message_1);
}

if (oldvalue2 != value_2){
  oldValue2 = value_2;
  String message_2 = "led"+str(2)+" "+str(value_2);
  println(message_2);
  wsc.sendMessage(message_2);
}

```

```

Serial.begin(115200);
for(int i=0; i < N_Channel; i++) {
  pinMode(AC_pin[i], OUTPUT); // Set the Triac pin as output
}
WiFi.begin(ssid, password);

while(WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
  delay(200);
}

Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

attachInterrupt(12, zero_cross_detect, RISING); // Attach an Interrupt to Pin 2 (Interrupt 0) for Zero Cross Detection

timer1_attachInterrupt(dim_check);
timer1_enable(TIM_DIV16, TIM_EDGE, TIM_SINGLE);
timer1_write(freqStep);

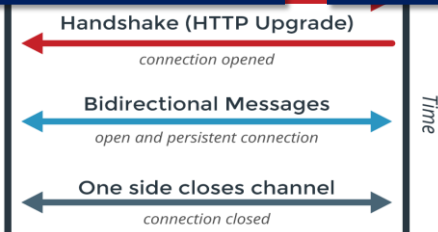
delay(300);

Serial.println("Start Websocket Server");
WebSocket.begin();
WebSocket.onEvent(WebSocketEvent);

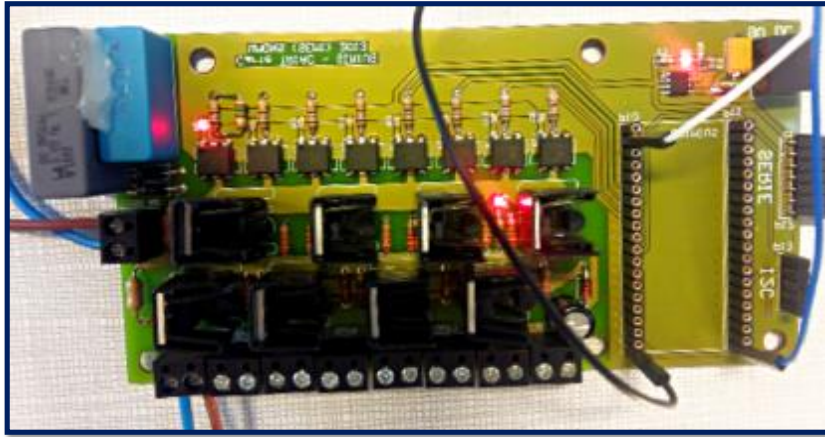
```

Client

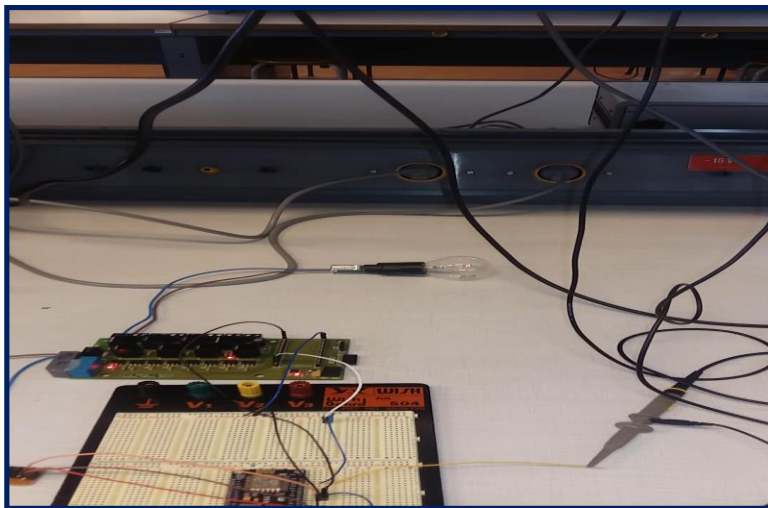
Server



18

OUTPUT GENERATOR

19

TUTORIAL:

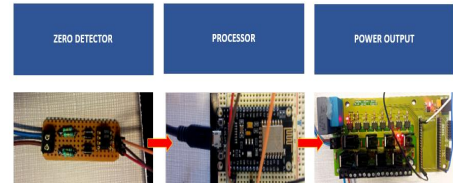
20



CONCLUSION :

In this project , we have been able to :

- Develop an interface user to switch on/off the PowerPlug or dim the energy level .
- Develop a microprocessor program to manage the zero detection , dimming level and multichannels .
- Connect the program to a web server via websockets .
- Use a wifi-chip that will allow the user to connect to the web page .



21

THANK YOU FOR YOUR ATTENTION

22