

Universidade Federal de São Carlos

Laboratório de Microcontroladores e Aplicações
Professor Dr. Edilson Kato

Relatório 1 - Turma A

Bruna Zamith (RA: 628093)
Matheus Vrech (RA: 727349)

08/2018
São Carlos - SP, Brasil

Conteúdo

1	Introdução	1
2	Descrição	1
3	Materiais Utilizados	1
3.1	Arduino	2
3.2	Grove Starter Kit	2
4	Desenvolvimento	3
4.1	Versionamento	3
4.2	Página Web	3
4.3	Implementação	5
5	Resultados	9

1 Introdução

O presente relatório visa detalhar as atividades desenvolvidas ao longo das aulas de Laboratório de Microcontroladores e Aplicações ministradas nos dias 24/08 e 31/08 pelo professor Dr. Edilson Kato no Departamento de Computação da Universidade Federal de São Carlos.

O documento está organizado da seguinte forma: A Seção 2 descreve o projeto proposto e seus objetivos principais; a Seção 3 expõe os materiais utilizados para a implementação do projeto; a Seção 4 detalha o seu desenvolvimento; por fim, a Seção 5 expõe e discute os resultados obtidos.

2 Descrição

O projeto a ser detalhado neste relatório objetiva o aprofundamento do conhecimento em Arduino e demais tecnologias, mais especificamente, a interface web e *Liquid Crystal Display* (LCD).

A proposta era utilizar uma página Web para, através de:

1. Um botão “ON” na página, ligar o LED do Arduino;
2. Um botão “OFF” na página, desligar o LED do Arduino;
3. Um botão conectado na placa, exibir no LCD, também conectado à placa, o nome e o RA dos integrantes do grupo;
4. O pressionamento da tecla "B"na página Web, exibir no LCD o nome e o RA da integrante "Bruna Zamith";
5. O pressionamento da tecla "V"na página Web, exibir no LCD o nome e o RA do integrante "Matheus Vrech";
6. Cada *display* no LCD, mudar sua cor de fundo.

3 Materiais Utilizados

Para a implementação do projeto descrito na seção anterior, foram utilizados:

- 1 Arduino Mega;
- 1 *Shield Ethernet*;

- 1 Cabo Ethernet;
- 1 Grove *Base Shield*;
- 1 Grove *Button*;
- 1 Grove LCD;
- Arduino *Integrated Development Environment* (IDE).

3.1 Arduino

O Arduino é uma plataforma *open-source* de *hardware* e *software*, capaz de ler entradas (como sensores e botões) e transformá-las em saídas (como motor e LED). Ele permite a prototipagem eletrônica de *hardware* livre e é projetado com um microcontrolador Atmel AVR¹. É possível programá-lo através de sua IDE e de sua linguagem de programação própria, sendo esta última semelhante à linguagem C. A Figura 1 exibe um Arduino e a Figura 2, a IDE.



Figura 1: Arduino

```

sketch_jan04a | Arduino 1.8.5
File Edit Sketch Tools Help
sketch_jan04a:
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

```

Arduino/Genuino Uno on COM1

Figura 2: IDE do Arduino

3.2 Grove Starter Kit

Grove é um conjunto de ferramentas modulares, projetado para minimizar a dificuldade da engenharia eletrônica fundamental. É formado pelos módulos funcionais TWIG e placa

¹<https://www.arduino.cc/en/Guide/Introduction>

de interface STEM. Cada TWIG unificou a interface de 4 pinos e a forma de quebra-cabeças padronizada para fácil combinação. Eles podem trabalhar com a maior parte das plataformas de desenvolvimento existentes (como Arduino e placas compatíveis, placa beagle, Xbee e etc) via STEM². A Figura 3 mostra o Grove Starter Kit.

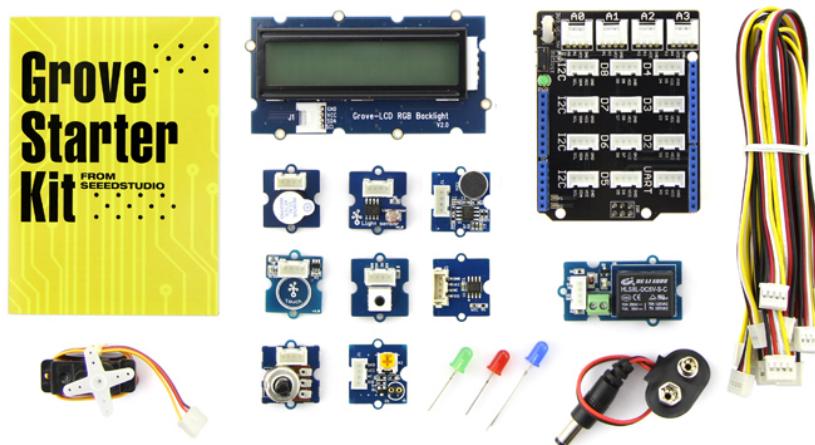


Figura 3: Grove Starter Kit

4 Desenvolvimento

4.1 Versionamento

O controle de versões do nosso código está sendo feito a partir do GitHub, repositório público “Zavech”³. O projeto detalhado neste relatório e todos os outros projetos da disciplina serão incluídos nesse mesmo repositório.

4.2 Página Web

A página Web do projeto foi desenvolvida fazendo uso de HTML e Javascript, e das tecnologias Ajax e Bootstrap. É composta de duas páginas principais: “Home”, contendo infor-

²<https://www.seeedstudio.com/Grove-Starter-Kit-p-709.html>

³<https://github.com/whoismath/zavech>

mações sobre o projeto, lista de tarefas e instruções de como conectar a placa; “Activities”, a qual possuí todas as *features* implementadas.

O site pode ser acessado a partir do *link*: <https://whoismath.github.io/zavech/>.

As Figuras 4 e 5 exibem as interfaces da página Web. Para correto funcionamento, é preciso apenas que o servidor e o arduino estejam conectados na mesma rede.

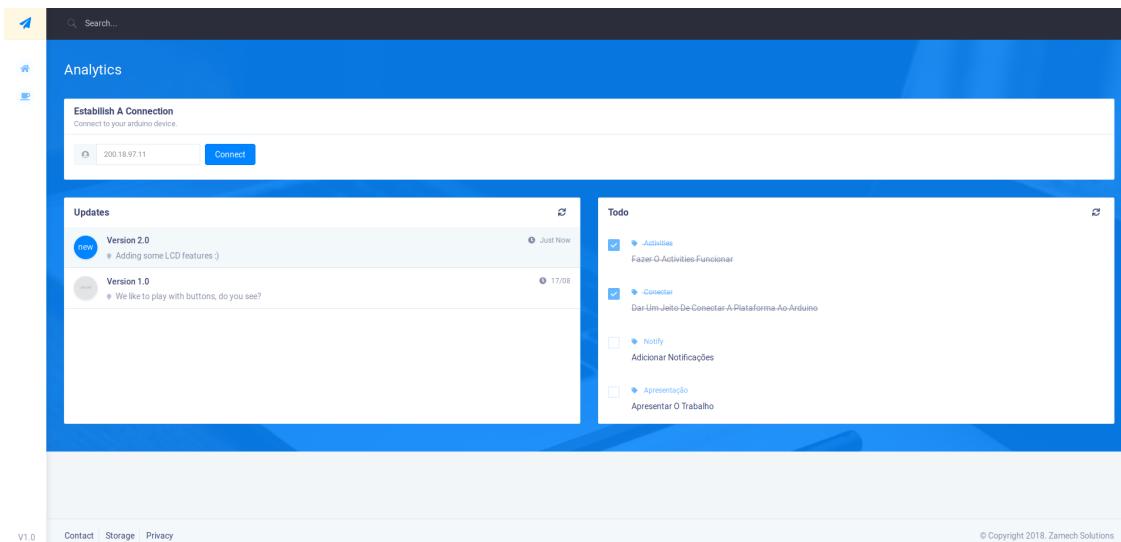


Figura 4: Página “Home”

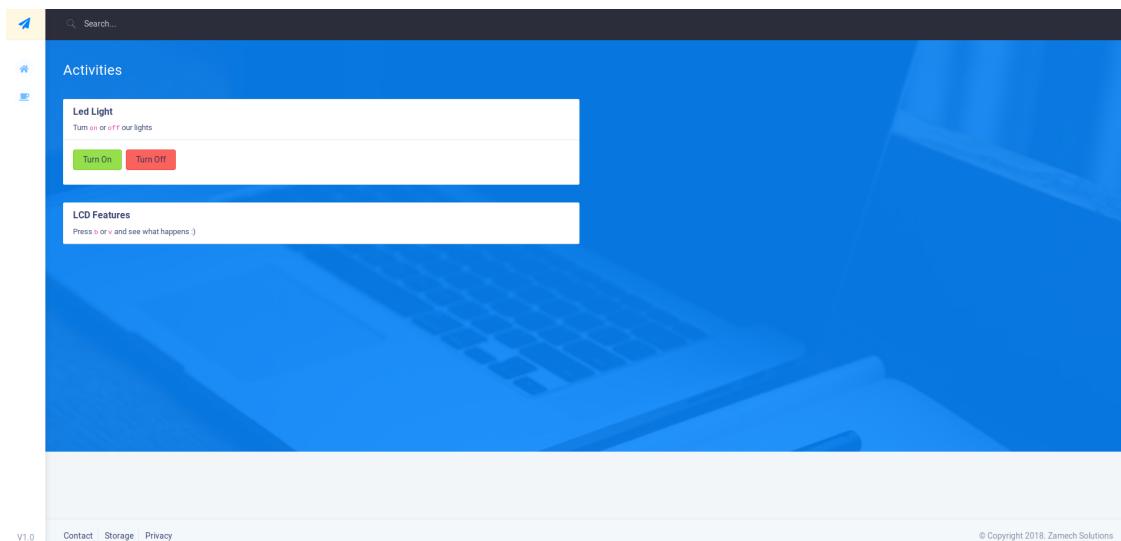


Figura 5: Página “Activities”

4.3 Implementação

As conexões foram feitas da seguinte forma:

- Grove *Button* conectado na porta D4 do Grove *Base Shield*;
- Grove LCD conectado na porta I2C do Grove *Base Shield*;
- Servidor Web e Arduino conectados na mesma rede, endereço IP 200.18.97.11.

Para a programação, foram primeiramente importadas 4 diferentes bibliotecas:

- <**SPI**>: *Serial Peripheral Interface*, permite a conexão com periféricos através do protocolo SPI⁴;
- <**Ethernet**>: Desenvolvida para o *Shield Ethernet*, permitindo que o Arduino conecte-se à internet⁵;
- <**Wire**>: Permite a comunicação com dispositivos I2C/TWI, como é o caso do LCD⁶;
- **rgb_lcd.h**: Permite a mudança de cor de fundo do LCD, dentre outras funcionalidades⁷.

O código final `.ino` pode ser encontrado em <https://github.com/whoismath/zavech/blob/master/class2.ino> e também exposto a seguir. A lógica do programa é a seguinte:

1. Dentro da função `setup()`, são feitas as configurações e atribuições das portas, endereço IP e LCD
2. Na função `loop()`, são feitas, a cada ciclo, algumas verificações:
 - (a) Se o botão foi pressionado. Se sim, um contador é incrementado. Caso o valor do contador seja par, é exibido no LCD as informações da integrante Bruna; senão, as informações do integrante Vrech. Cada um possui uma cor de fundo diferente;
 - (b) Se um cliente foi conectado e um HTTP *packet* foi recebido. Se sim, verifica o valor do GET. Caso seja recebido o valor “lighton”, o LED é acionado. Caso seja recebido o valor “lightoff”, o LED é desligado. Caso seja recebido o valor “bruna”, a informação correspondente é exibida no LCD. Caso seja recebido o

⁴<https://www.arduino.cc/en/Reference/SPI>

⁵<https://www.arduino.cc/en/Reference/Ethernet>

⁶<https://www.arduino.cc/en/Reference/Wire>

⁷http://wiki.seeedstudio.com/Grove-LCD_RGB_Backlight/

valor “vrechson”, a informação correspondente é exibida no LCD. Isso é possível pois, para cada comando enviado através da página Web, é enviado um método HTTP GET diferente.

```
1  /*
2   * Web Server
3   * A simple web server that shows the value of the analog input pins.
4   * using an Arduino Wiznet Ethernet shield.
5   * Circuit:
6   *   * Ethernet shield attached to pins 10, 11, 12, 13
7   *   * Analog inputs attached to pins A0 through A5 (optional)
8   * created 18 Dec 2009
9   * by David A. Mellis
10  * modified 9 Apr 2012
11  * by Tom Igoe
12  * modified 02 Sept 2015
13  * by Arturo Guadalupi
14  * Besides that, this project was modified by Matheus Vrech and Bruna Zamith
15  * for university classes purpose, enjoy it
16  */
17
18 #include <SPI.h>
19 #include <Ethernet.h>
20 #include <Wire.h>
21 #include "rgb_lcd.h"
22
23 rgb_lcd lcd;
24
25 const int pinButton = 4;
26 int counter = 0;
27
28 byte mac[] = {
29   0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
30 };
31
32 IPAddress ip(200, 18, 97, 11);
33 EthernetServer server(80);
34
35 void setup() {
36   Serial.begin(9600);
37   while (!Serial) {}
38   Ethernet.begin(mac, ip);
```

```

    server.begin();
39 Serial.print("server is at ");
Serial.println(Ethernet.localIP());

41 /* set up the LCD's number of columns and rows: */
43 lcd.begin(16, 2);
lcd.setRGB(186, 85, 211);

45 /* Print a message to the LCD. */
47 lcd.print("Press B or V!");

49 /* Define a led to use */
pinMode(LED_BUILTIN, OUTPUT);
51 pinMode(pinButton, INPUT);
}

53

55 void loop() {
56 if (digitalRead(pinButton)){
57   counter++;
58 }
59 if (counter > 0){
60   if (counter % 2 == 0) {
61     display_lcd("Bruna Zamith", "628093");
62   } else {
63     display_lcd("Matheus Vrech", "727349");
64   }
65 }

67
68 EthernetClient client = server.available();
69 /* Get the http packet*/
70 String readString;
71 if (client) {
72   boolean currentLineIsBlank = true;
73   while (client.connected()) {
74     if (client.available()) {
75       char c = client.read();

76       Serial.write(c);

77       /* Store the entire packet */
78     }
79   }
}

```

```

    if (readString.length() < 100) {
      readString += c;
    }

/* Receive GET commands */
if (readString.indexOf("lighton") > 0) {
  digitalWrite(LED_BUILTIN, HIGH);
}
else if (readString.indexOf("lightoff") > 0) {
  digitalWrite(LED_BUILTIN, LOW);
}
else if (readString.indexOf("bruna") > 0) {
  display_lcd("Bruna Zamith", "628093");
}
else if (readString.indexOf("vrechson") > 0) {
  display_lcd("Matheus Vrech", "727349");
}

if (c == '\n' && currentLineIsBlank) {
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");
  client.println("Connection: keep-alive");
  //client.println("Refresh: 5"); // refresh the page
automatically every 5 sec
  client.println();
  client.println("<!DOCTYPE HTML>");
  client.println("<html>");
  client.println("<head></head>");
  /* easter egg for ninjas */
  client.println("<body>i'm a happy arduino, and you found my
easter egg!<br>curiosity will kill you.</body>");
  client.println("</html>");
}
if (c == '\n') {
  currentLineIsBlank = true;
} else if (c != '\r') {
  currentLineIsBlank = false;
}
}

/* wait for something: nothing at all*/

```

```

121     delay(1);
122     // client.stop();
123     // Serial.println("client disconnected");
124 }
125
126 void display_lcd(String nome, String ra){
127     lcd.clear();
128     if(nome.indexOf("Zamith") > 0){
129         lcd.setRGB(255, 127, 80);
130     }
131     else if(nome.indexOf("Vrech") > 0){
132         lcd.setRGB(135, 206, 235);
133     }
134     lcd.setCursor(0, 0);
135     lcd.print(nome);
136     lcd.setCursor(0,1);
137     lcd.print(ra);
138     delay(300);
139 }
```

5 Resultados

Todos os resultados obtidos se mostraram satisfatórios, sendo possível implementar e testar as funcionalidades propostas e descritas na Seção 2. Não foi encontrada nenhuma dificuldade relativa à disciplina, apenas referente ao conhecimento prévio necessário em Javascript e desenvolvimento Web.