

Laboratório de instrumentação eletrônica e controle  
(LIEC)

Relatório de Projeto

**Desenvolvimento de um sistema de acionamento de  
cargas via wifi utilizando microcontrolador**

Pedro Ivo Aragão Guimarães  
pedro.guimaraes@ee.ufcg.edu.br

Campina Grande, Agosto de 2015

## SEMANA I

- Planejamento e elaboração do projeto Embrapii.
- Estudo das camadas de rede.
- Estudo do *protocoloco Modbus*.
- Implementação do protocolo *Modbus*, software\* e hardware, em um *ethernet shield(wiznet w5100)*. (Controle de led: escrita e leitura).

## SEMANA II

- Formatação e configuração de *softwares* do novo computador do laboratório.
- Configuração do novo roteador.
- Estudo do módulo wifi ZG2100mc\*.
- Teste de *plataforma de desenvolvimento* fornecida para o protótipo.

## SEMANA III

- Implementação de *webserver* para acionamento dos relés e monitoriamento de estado de lâmpada com objetivo de estudo das pilhas tcp/ip.
- Estudo do novo *wifi shield(mrf24wboma)\**.

## SEMANA IV

- Estudo do módulo ESP8266\*.
- Construção do novo protocolo(pacote) para o projeto.
- Estudo da biblioteca *strings.h*
- *Inicio do desenvolvimento do software\* para o protótipo.(recepção de dados do servidor).*

\* Todos os software, bibliotecas e manuais estão disponível no google drive. Pasta "TESS".

Zg2100mc -

<https://drive.google.com/a/ee.ufcg.edu.br/folderview?id=0BzgD9WW2milHYIFUTnZtSnVTX1U&usp=sharing>

MRF24WBOMA -

<https://drive.google.com/a/ee.ufcg.edu.br/folderview?id=0BzgD9WW2milHc3hmOFZ5bmpuYW8&usp=sharing>

Esp8266 -

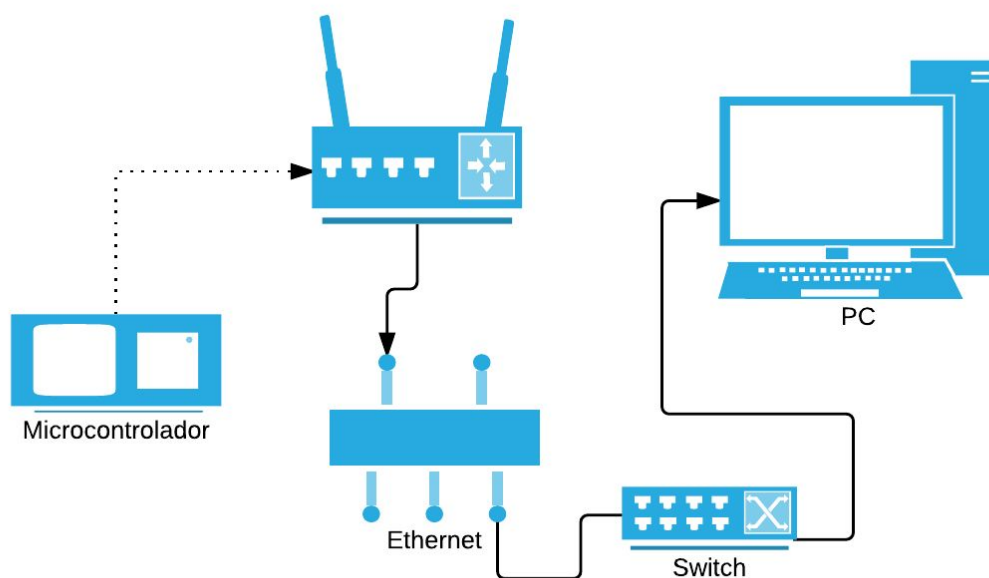
<https://drive.google.com/a/ee.ufcg.edu.br/folderview?id=0BzgD9WW2milHdEJvS2VOTWJ1a0E&usp=sharing>

# Camadas de rede.



“Suas principais funções são o roteamento dos pacotes entre fonte e destino, mesmo que estes tenham que passar por diversos nós intermediários durante o percurso, o controle de congestionamento e a contabilização do número de pacotes ou bytes utilizados pelo usuário, para fins de tarifação.”

Nosso projeto manipulará os dados na fonte (microcontrolador/pc) e enviará via camada Física (spi ou serial) para um periférico (módulo wifi) que fará o enlace wifi do dado e transmitirá o pacote pelas camadas de rede e transporte tcp/ip até o seu destino (microcontrolador/pc).



# Modbus.

Modbus é um protocolo de comunicação de dados amplamente utilizado em sistemas das indústrias, e foi usado como referência para a construção do “frame”.

## FRAME MODBUS.

endereço	comando	end. dos registros		quant. de registros		CRC
02	03	00	00	00	0A	2 caracteres

Implementamos o protocolo modbus tcp em um ethernet shield(wiz5100) como “*Server*” e a comunicação com o “*Client Tcp Modbus*” no computador para escrita/leitura das portas digitais. Um led foi usado para simulação de estados. O objetivo da simulação foi o estudo e compreensão do “pacote” modbus.

O “*Simply Modbus Tcp Client*” foi usado no computador para a comunicação com o arduino e ethernet shield. A biblioteca modbus tcp e os *softwares* estão disponíveis:

<https://drive.google.com/a/ee.ufcg.edu.br/folderview?id=0BzgD9WW2milHay1LSVdydVpWZTA&usp=sharing> (link do sketch (soft arduino) comentado).

<http://www.simplymodbus.ca/TCPclient.htm> (Simply Modbus Tcp Client)

# ZG2100M.

Zg2100m é um modulo wifi já obsoleto fabricado pela ZeroG e posteriormente Microchip. O nosso chip já vem soldado em uma placa com os componentes adicionais para o seu funcionamento, foi levantado e documentado uma lista de I/O dos pinos da placa com o auxilio do multímetro e datasheet.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	MÓDULO (PIN BAR)		CHIP										
2													
3	PINO 20		GND										
4	PINO 19		VDD33 (17)										
5	PINO 18												
6	PINO 17												
7	PINO 16												
8	PINO 15												
9	PINO 14												
10	PINO 13												
11	PINO 12		INT_NX(12)										
12	PINO 11												
13	PINO 10												
14	PINO 09												
15	PINO 08		SCN_N(23)										
16	PINO 07												
17	PINO 06												
18	PINO 05		SDI(35)										
19	PINO 04		SCK(34)										
20	PINO 03		SDO(32)										
21	PINO 02												
22	PINO 01		RST_IN(R8)										
23													
24													
25													
26													
27													

[https://docs.google.com/a/ee.ufcg.edu.br/spreadsheets/d/1tpKs3z91oq2BQ\\_3ueLA-L7Vxdt5fIVvVt-glAPu5sn8/edit?usp=sharing](https://docs.google.com/a/ee.ufcg.edu.br/spreadsheets/d/1tpKs3z91oq2BQ_3ueLA-L7Vxdt5fIVvVt-glAPu5sn8/edit?usp=sharing)

## Desvantagens:

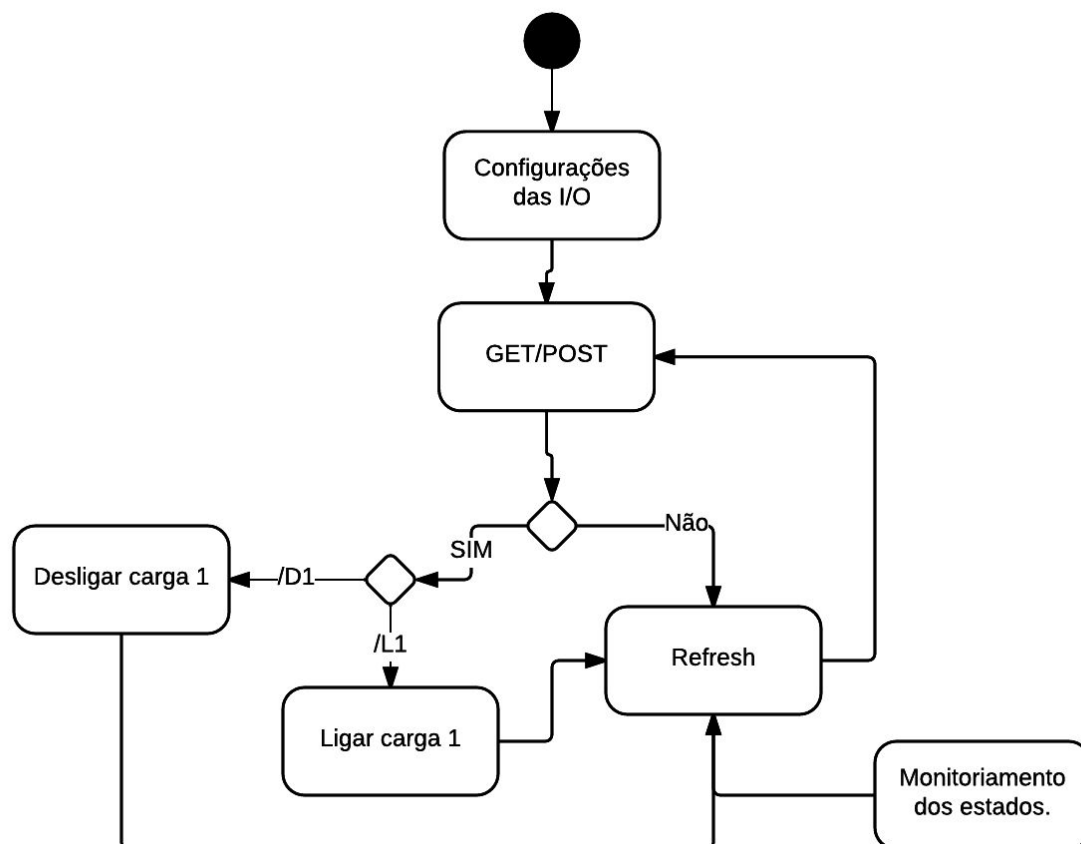
- Obsoleto.
- Não roda a pilha tcp/ip embutida(processamento mais lento e sobrecarrega o microcontrolador).
- Biblioteca compatível apenas com versões antigas do arduino (0023) alpha.

Vantagens:

- Fabricante conhecido.
- Bibliotecas e exemplos disponíveis para o nosso microcontrolador (atmega328p).
- O fato da pilha tcp/ip ser externa e com acesso, traria vantagens caso fosse alguma aplicação mais complexa e robusta.

Criamos um *webserver* com o objetivo de estudar a biblioteca e pilha tcp/ip disponível para o módulo. As funções *GET/POST* são usadas para o controle de três cargas da plataforma de teste e monitoramento dos estados das cargas. O microcontrolador, como client, requisita o GET temporariamente no endereço do servidor <http://10.0.0.13/> e envia o código html armazenado na sua memória flash. Caso haja alguma mudança na url para (<http://10.0.0.13/D1> ou <http://10.0.0.13/L1>) ele identifica o /L1 ou /D1 e associa a alguma lógica pré programada.(ligar ou desligar carga 1).

UML simples:



<https://drive.google.com/a/ee.ufcg.edu.br/folderview?id=0BzgD9WW2milHQTfMN0IMMjlSVzA&usp=sharing>

(link do sketch (software do arduino))

# *Shield MRF24WBOMA.*

*O mrf24wboma é o chip de um shield wifi para arduino produzido pela microchip com as mesmas características do zg2100mc. Uma vantagem adicional é uma memória de 16Mb para armazenagem de páginas html mais complexas.*

## Características Shield:

- Compatível com Arduino Diecimila, Duemilanove e Uno R2/R3.
- Utiliza comunicação SPI (velocidade máxima de 25MHz).
- Todos os pinos do Arduino estão disponíveis na parte superior.
- Botão reset de fácil acesso on-board.
- Antena integrada ao shield.
- Uso de pino de interrupção selecionável entre INT0 e pino digital 8.
- LED no pino digital 9 que pode ser habilitado via jumper.
- Pino CS (memória flash serial) selecionável entre pino digital 10 e pino digital 7.
- Dimensões: 5,7 x 5,7 x 1,5cm

## Características do módulo Wi-Fi:

- Wi-Fi 802.11b certificada.
  - Taxa de transferência de 1Mbps e 2Mbps.
- Suporta infra-estrutura (BSS) e ad hoc (IBSS) redes wireless.
- Capacidade de criar redes com e sem segurança.
  - WEP (64-bit e 128-bit).
  - WPA/WPA2 (TKIP e AES) PSK.
- Baixo consumo de energia.
  - Suspenso: 250µA
  - Transmitindo: 230mA
  - Recebendo: 85mA

## Resultados:

O shield funcionou perfeitamente com os mesmo programas construídos para o zg2100mc, foi usado a mesma biblioteca do zg2100mc, criada para o arduino. A biblioteca é bastante complexa, com mais de 8mil linhas de códigos e estudar a fundo para modificações demandaria um longo tempo, além dele ser recomendado pela própria microchip para desenvolvedores de redes avançados e com um microcontrolador de no mínimo 16 bits. A pilha criada para o arduino também é limitada, criada basicamente para atender projetos com webserver ou webclient. Caso fosse trabalhado com algum microcontrolador da microchip poderíamos ter algumas vantagens por utilizar a pilha tcp/ip do próprio fabricante.

A microchip recomenda em seu site para desenvolvedores que não são especialistas em redes a utilização da linha RN (rn131 ou rn171) que já vem a pilha integrada ao chip.

Resolveu-se aposentar temporariamente o shield e focar na outra opção, esp8266, que além de ser um chip mais moderno e robusto atende as nossas demandas.

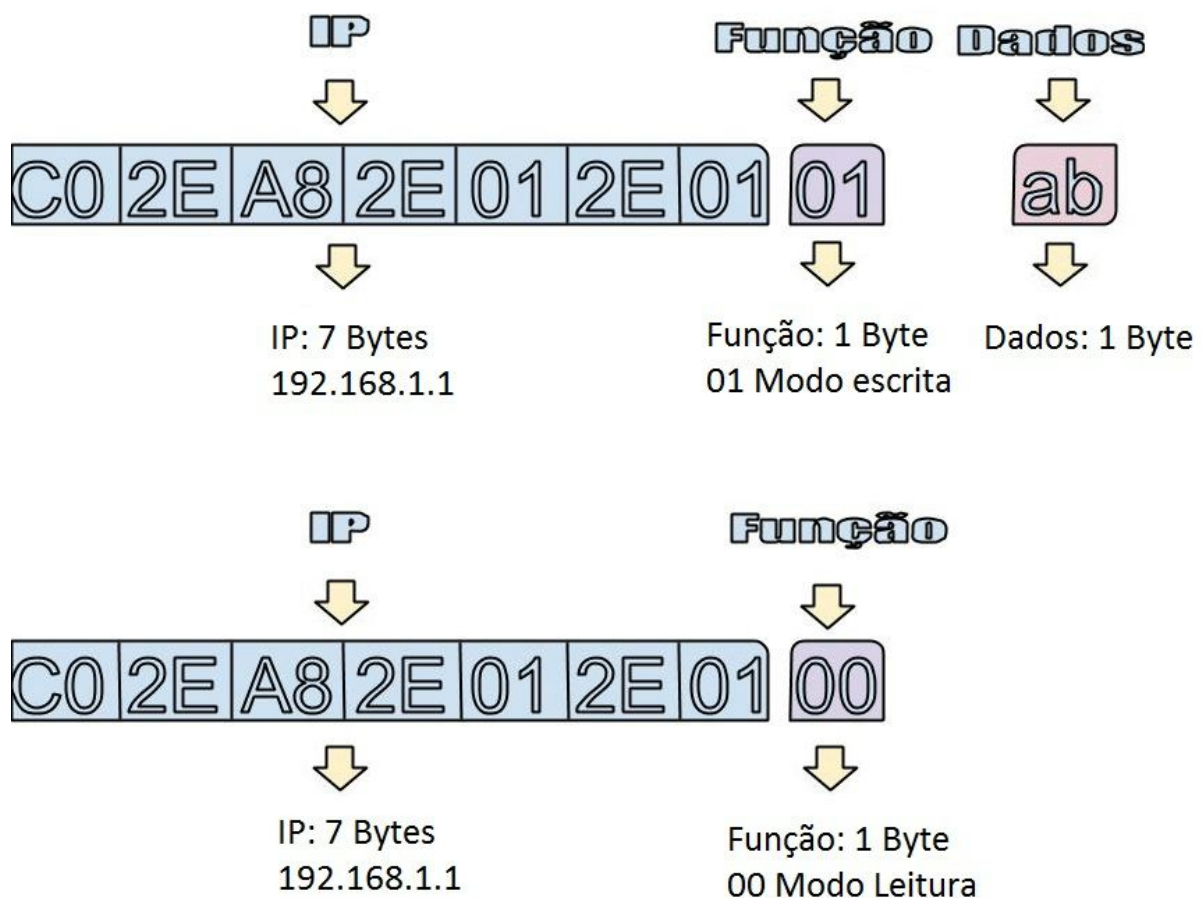
Após a conclusão do protótipo, poderemos retomar o estudo do mrf24wboma, caso seja necessário, por algum motivo técnico do projeto.



# Frame.

O frame (pacote) que será transportado pelo tcp conterá 9 bytes no modo escrita e 8 bytes no modo leitura, adicionados de mais dois bytes \CR\LF. O primeiro campo é o IP que será o ID do pacote, em seguida a função, escrita ou leitura, seguida pelo byte responsável pela escrita(caso o modo seja escrita) dos estados nos registradores de saídas. Futuramente será adicionado um CRC ou um checksum para verificação do pacote pelo destinatário.

O frame é enviado em hexadecimal e pode controlar ou ler o estado de até 8 saídas (um byte).



# ESP8862.

## Características:

- É um System-On-Chip com Wi-Fi embutido;
- Tem conectores GPIO, barramentos I2C, SPI, UART, entrada ADC, saída PWM e sensor interno de temperatura;
- CPU que opera em 80MHz, com possibilidade de operar em 160MHz;
- Arquitetura RISC de 32 bits;
- 32KBytes de RAM para instruções;
- 96KBytes de RAM para dados;
- 64KBytes de ROM para boot;
- Possui uma memória Flash SPI Winbond W25Q40BVNIG de 512KBytes;
- O núcleo é baseado no IP Diamand Standard LX3 da Tensilica;
- Fabricado pela Espressif;
- Chip

	ESP8266	Arduino (Uno)
GPIOs	17 (Fewer typically exposed)	14 (20 including analog)
Analog input	1	6
PWM channels	8	6
Clock speed	80MHz	16MHz
Processor	Tensilica	Atmel
SRAM	45KBytes	2KBytes
Flash	512Kb or more (separate)	32KB (on chip)
Operating Voltage	3.3V	5V
Max current per I/O	12mA	40mA
UART (hardware)	1 ½	1
Networking	Built-in	Separate
Documentation	Poor	Excellent
Maturity	Early	Mature

tabela de comparação com o microcontrolador do arduino uno.

A sua grande vantagem diante os demais trabalhados é a pilha tcp/ip embutida, além de usar um mcu de 32bits com um clock de 80mhz. Foi bem notório a sua velocidade conexão e resposta, e a simplicidade de trabalhar com comandos seriais AT pré-definidos.

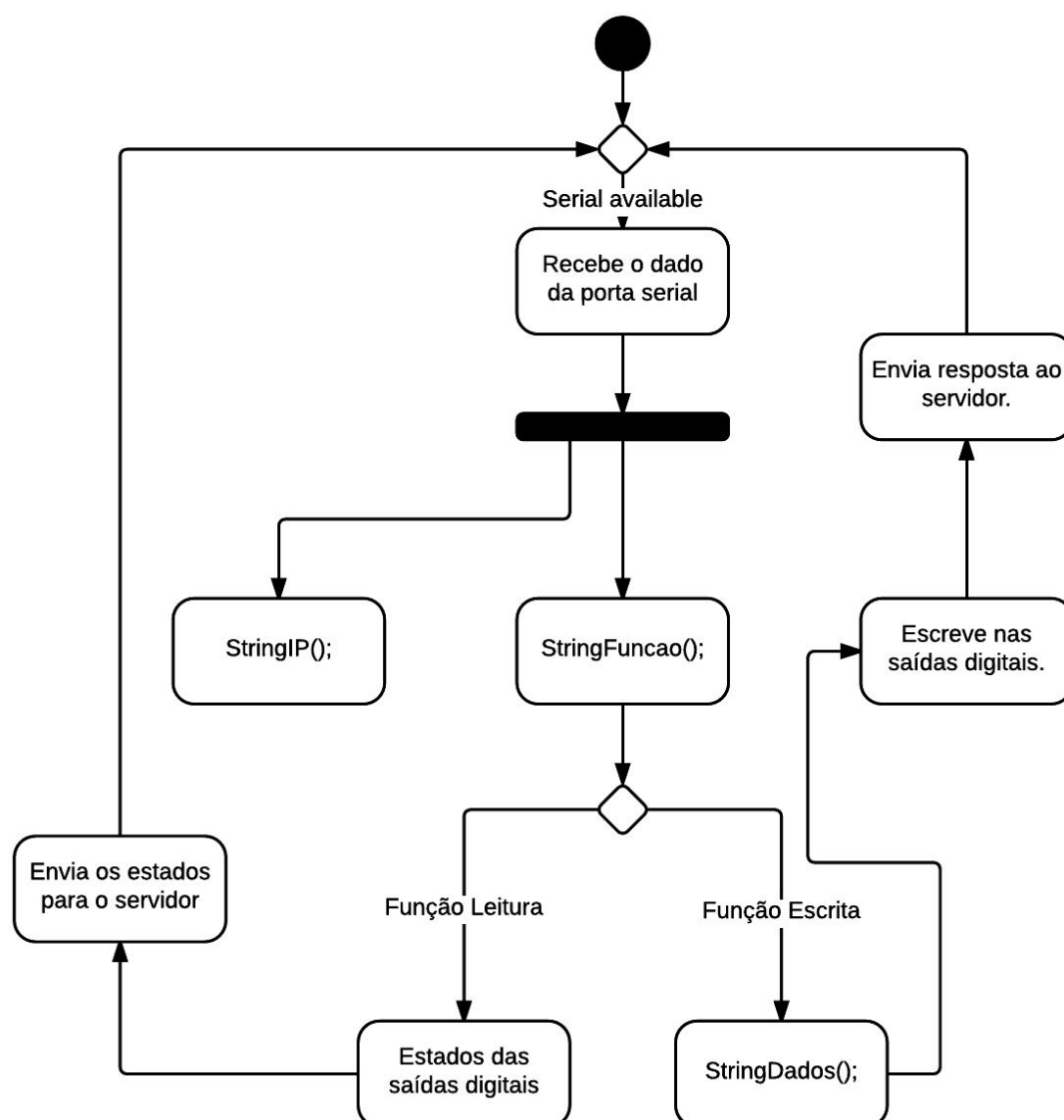
A desvantagem é o fabricante chinês ainda pouco conhecido, porém o fabricante do chip é CADENCE

\*todos os manuais encontrados em inglês e chinês está na pasta do drive do projeto, inclusive um livro que foi lançado esse mês (agosto\2015).

# Recepção de dados.

Parte do código(main) desenvolvido para o esp8266 é a recepção de dados. O esp8266 recebe o frame enviado pelo Servidor em hexadecimal byte a byte e agrupa em um array para o tratamento. A string recebida é dividida em mais três strings: IP, Função e Dados, e posteriormente o tratamento lógico e a resposta do microcontrolador é enviada para o servidor. Os módulos ESP só atendem a pacotes enviados pelo servidor e de tamanhos pré definidos.

<https://drive.google.com/a/ee.ufcg.edu.br/file/d/0BzgD9WW2milHc0Z0UkVfVZrNmM/view?usp=sharing> (sketch (programa arduino)).



# Tarefas Adicionais.

Configuração do roteador wifi:

O roteador foi configurado no modo repetidor de sinal, com as seguintes configurações:

Nome da rede(ssid): LIEC\_WIFI

Senha: LIEC\_123

Gateway: 10.0.0.50

Máscara:

Canal:

DHCP: LIGADO 10.0.0.51 - 10.0.0.199

Configurações WAN do roteador na rede principal:

ip: 150.165.52.132

Máscara: 255.255.255.0

Gateway: 150.165.52.254

DNS primário: 150.165.1.1

DNS secundário: 150.165.61.1

<http://10.0.0.50/>

login: admin

senha: admin