

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA – CEFET/RJ

Trabalho de Conclusão de Curso

Projeto final apresentado em cumprimento as
normas do Departamento de Educação Superior.
Curso: Engenharia Industrial Elétrica com ênfase em telecomunicações.

Aluno: Manoel Felipe Costa Furtado
Matricula: 0620200190

Orientador: Vinicius Coutinho de Oliveira

**Rio de Janeiro
Dezembro de 2015**

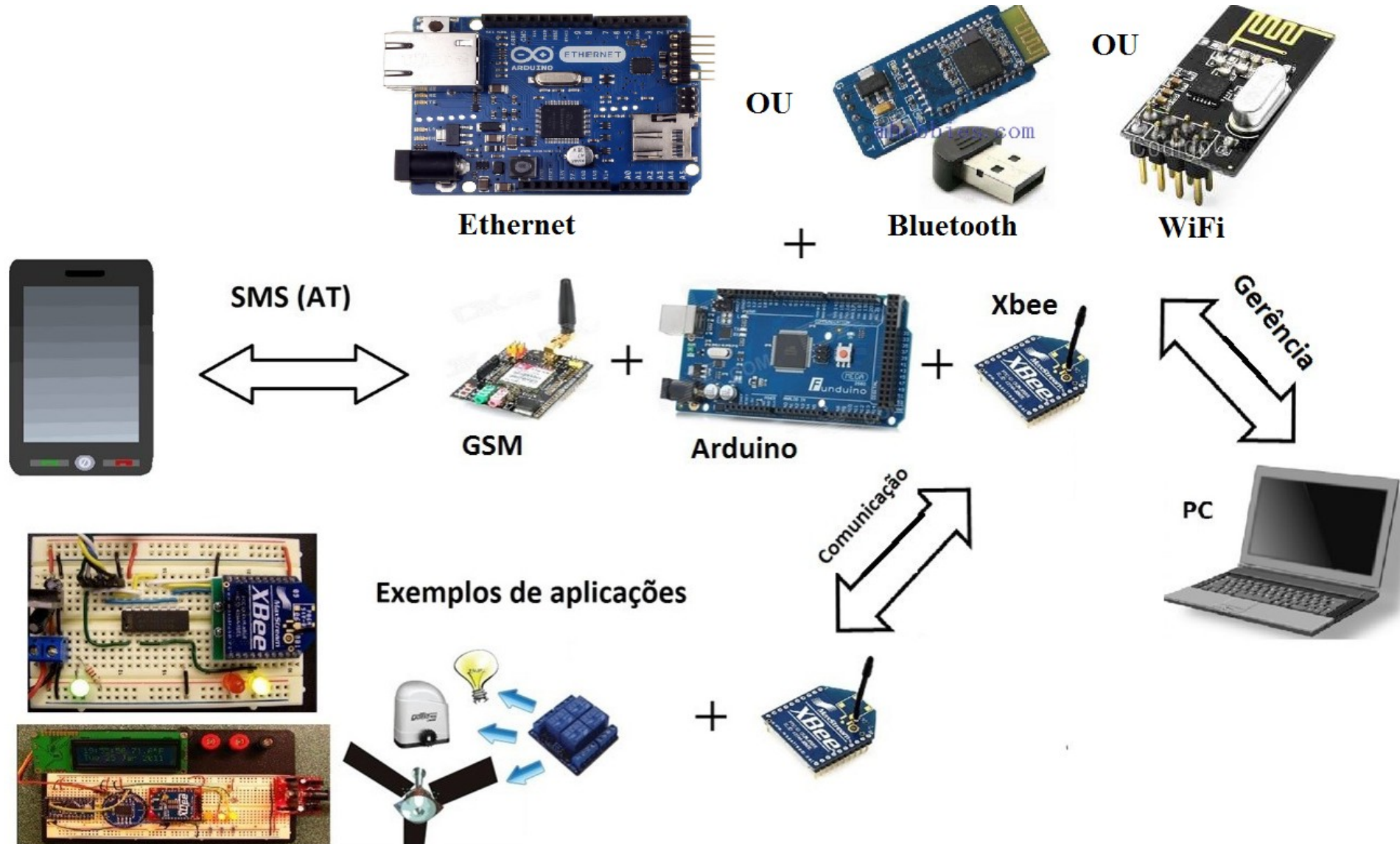
Tema do Trabalho de Conclusão de Curso

Projeto de Automação Residencial utilizando
Comunicação entre Mobile Station(GSM),
Arduino(GSM Shield), XBee e PC.

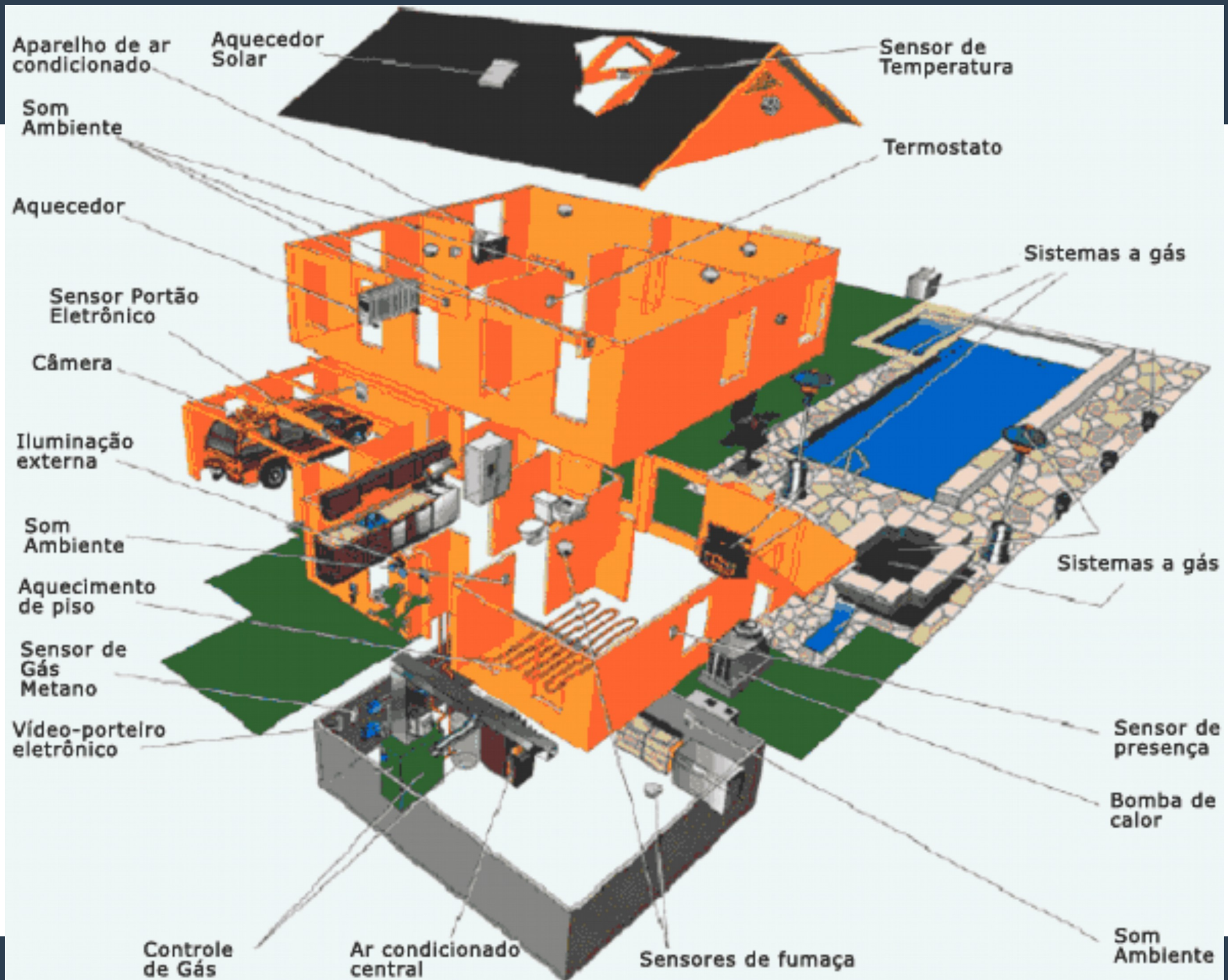
Motivação

- O avanço de novas tecnologias de comunicação sem fio e na microeletrônica, está facilitando a realização da automação residencial e comercial no dia a dia na sociedade globalizada em que vivemos, proporcionando até as classes menos favorecidas poder implementar esse serviço.
- Controlando o Xbee pelo GSM-SMS, garantindo uma eficiência energética, segurança e facilidade.

Arquitetura da Rede Proposta



Possibilidades



Temas Para a Implementação

- Protocolo Zig-Bee - IEEE 802.15.4
- Tecnologia GSM - Comandos AT
- Microcontrolador – Atmel
Hardware Arduino e suas bibliotecas
- Linguagem de Programação C/C++,
Shell Script, SQL, PHP, HTML, CSS,
JavaScript etc.

Interdisciplinariedade

- Na prática a interdisciplinaridade é um esforço de superar a fragmentação do conhecimento, tornar este relacionado com a realidade e os problemas da vida moderna. Na ciência, por sua vez, os esforços estão na busca de respostas, impossíveis com os conhecimentos fragmentados de uma única área especializada.
- Para viabilizar a rede proposta pode ser observado a necessidade da interdisciplinariedade dos cursos de Engenharia Industrial Elétrica com ênfase em Eletrônica, Telecomunicações, de Controle e Automação Ciência da Computação e Tecnologia em Sistemas para Internet no CEFET/RJ.

Exemplos Reais:

- 1) Na UNIVERSIDADE FEDERAL DO CEARÁ - UFC. Oferta o Bacharelado em Engenharia de Teleinformática reconhecido pelo MEC.
- 2) criação em 1986, o Grupo de Teleinformática e Automação (GTA) vinculado a (COPPE) da UFRJ.

Arduino e suas Bibliotecas



- Linus Torvald: “Estou fazendo um sistema operacional gratuito (apenas um hobby, não será grande e profissional como GNU) para 386/486 AT.”
 - Massimo Banzi: “Tornar acessíveis os microcontroladores com o código aberto para projetos interativos, a partir de instalações artísticas.
 - ”Relação comum?? Colaboração das pessoas pela internet. Globalização.
- “Você não precisa mais da permissão de alguém para fazer algo incrível.” Ter acesso ao código fonte e modificar à sua vontade é ser livre para progredir.

Objetivo do Protótipo Desenvolvido

- Nesse trabalho, será apresentada uma aplicação no protótipo eletrônico de hardware livre chamado Arduino que interligarão hardware proprietário garantindo a interoperabilidade de forma a garantir o funcionamento de um sistema de comunicação seguro (através do SMS), rápido (através da rede Móvel 2G) e com eficiência energética (através do ZigBee). O Arduino proporcionará inúmeras possibilidades para gerenciar, coletar informações interagindo com outras linguagens como Shell Script, MySQL, HTML, PHP, AJAX, JAVA.

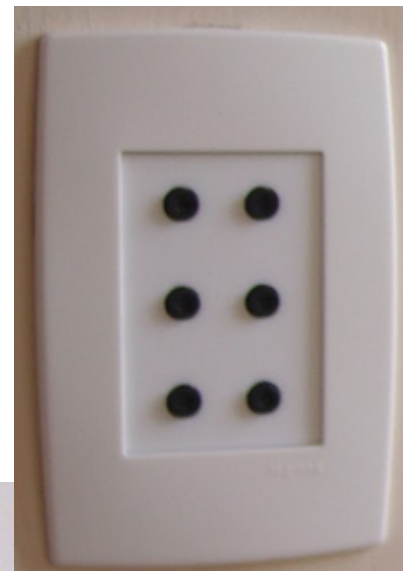
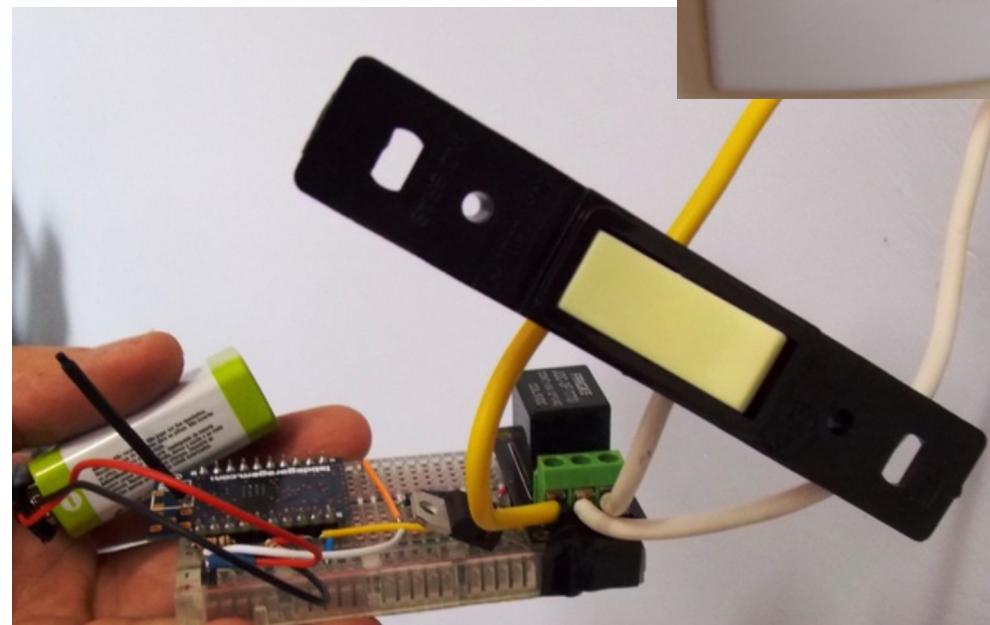
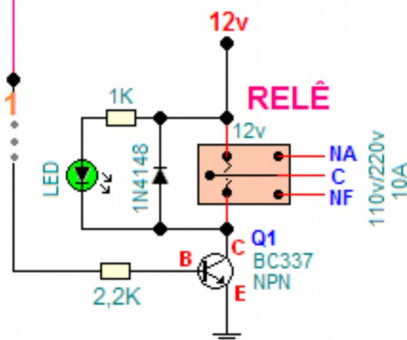
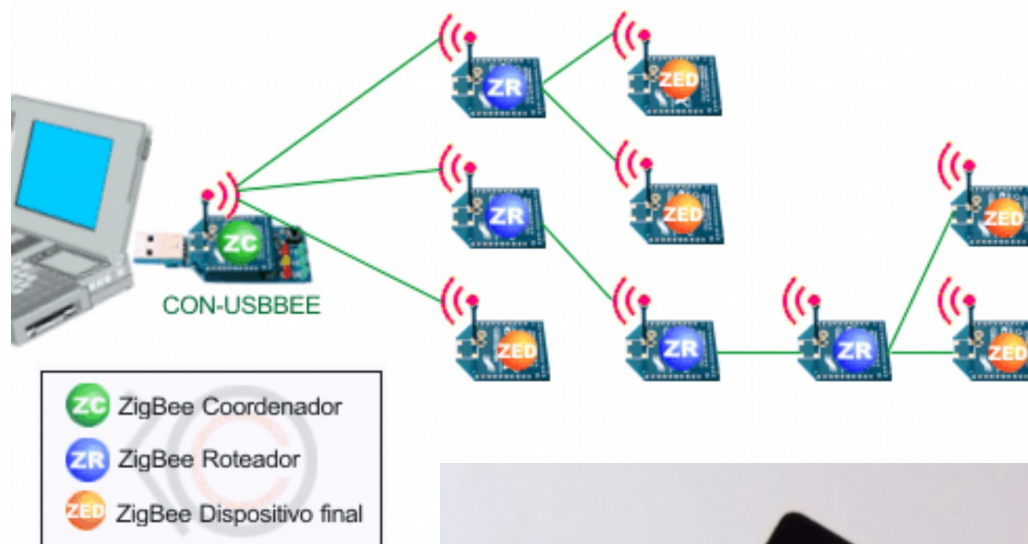
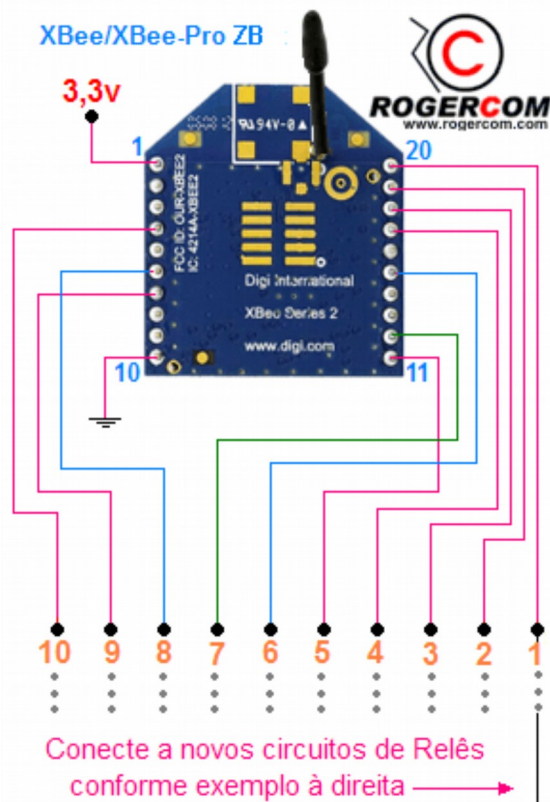
Objetivo do Protótipo Desenvolvido

- Será empregado um aparelho celular (GSM) para enviar os comandos e receber as informações via SMS ao Arduino integrado com o PC. Para essa comunicação, se estabelecer, será necessário o módulo GSM com um módulo Ethernet, Wifi ou bluetooth com o Arduino. Depois de estabelecida a comunicação, o módulo XBee, ligado no mesmo Arduino, poderá comunicar com outro XBee de modo que possa ser controlada por telefone móvel usando mensagem SMS fornecido pela tecnologia GSM. Essa comunicação entre XBee TX/RX será provida pelo Arduino, que recebeu as instruções pelo SMS, acionando uma porta digital qualquer utilizando a sua linguagem, no módulo XBee. Após o acionamento do módulo XBee receptor, mudará o estado lógico de uma das portas do XBee acionando a exemplo um relé. O PC gerenciará todo os eventos da comunicação e apresentar informações úteis.

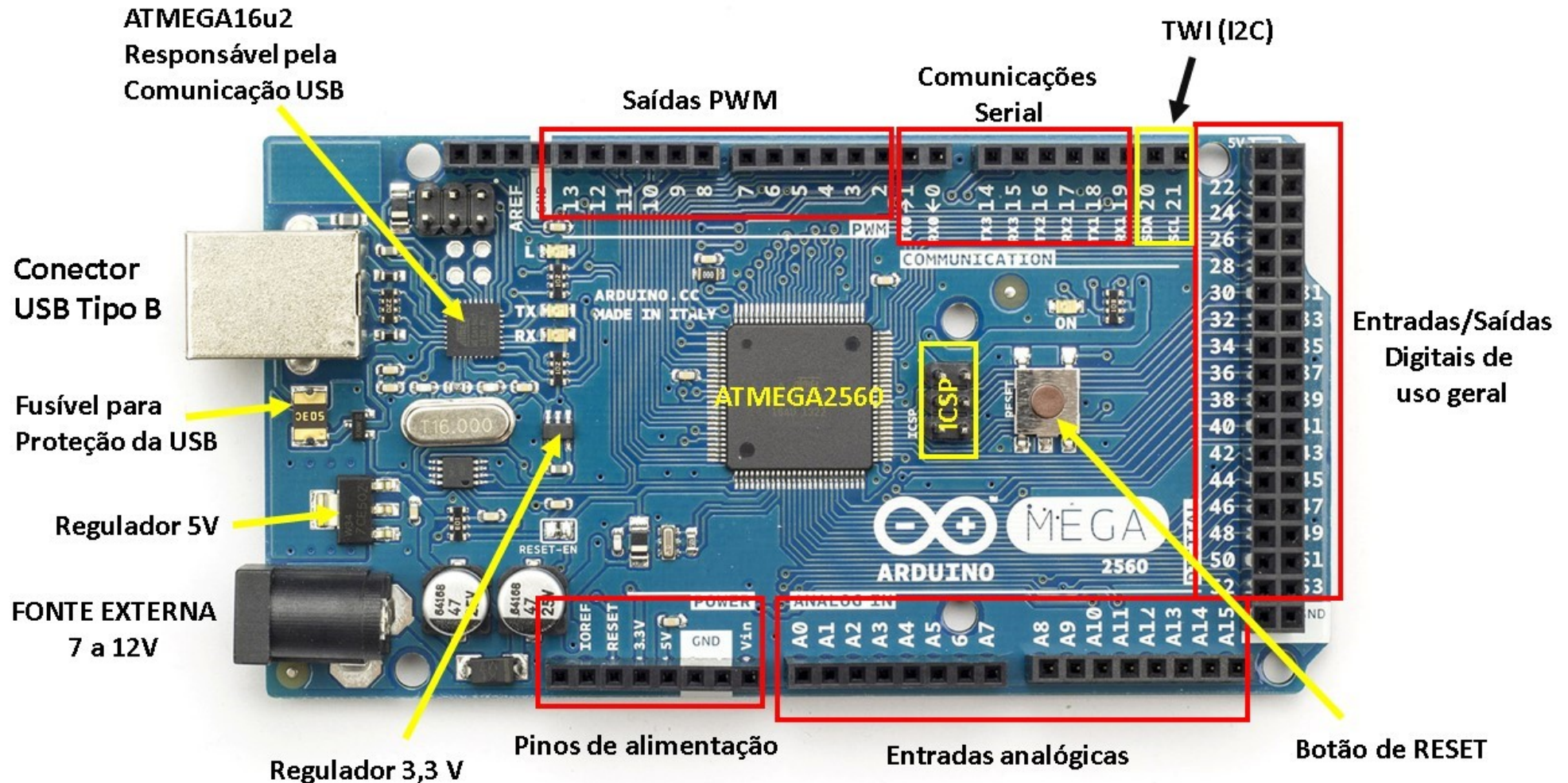
Programação necessária

- Programação da comunicação entre celular(GSM-SMS) e modulo (Shield-GSM) acoplado no Arduino.
- Programação da comunicação entre o Arduino e Xbee acoplado no Arduino. E entre o Xbee acoplado no Arduino e o outro Xbee receptor.
- Programação da comunicação das mensagens SMS através de comandos AT integrando todo o projeto.
- Programação da comunicação entre o PC e o Arduino.
- Desenvolvimento de interface de usuário e o registro dos eventos através de banco de dados e HTML, utilizando serviços GNU/Linux.

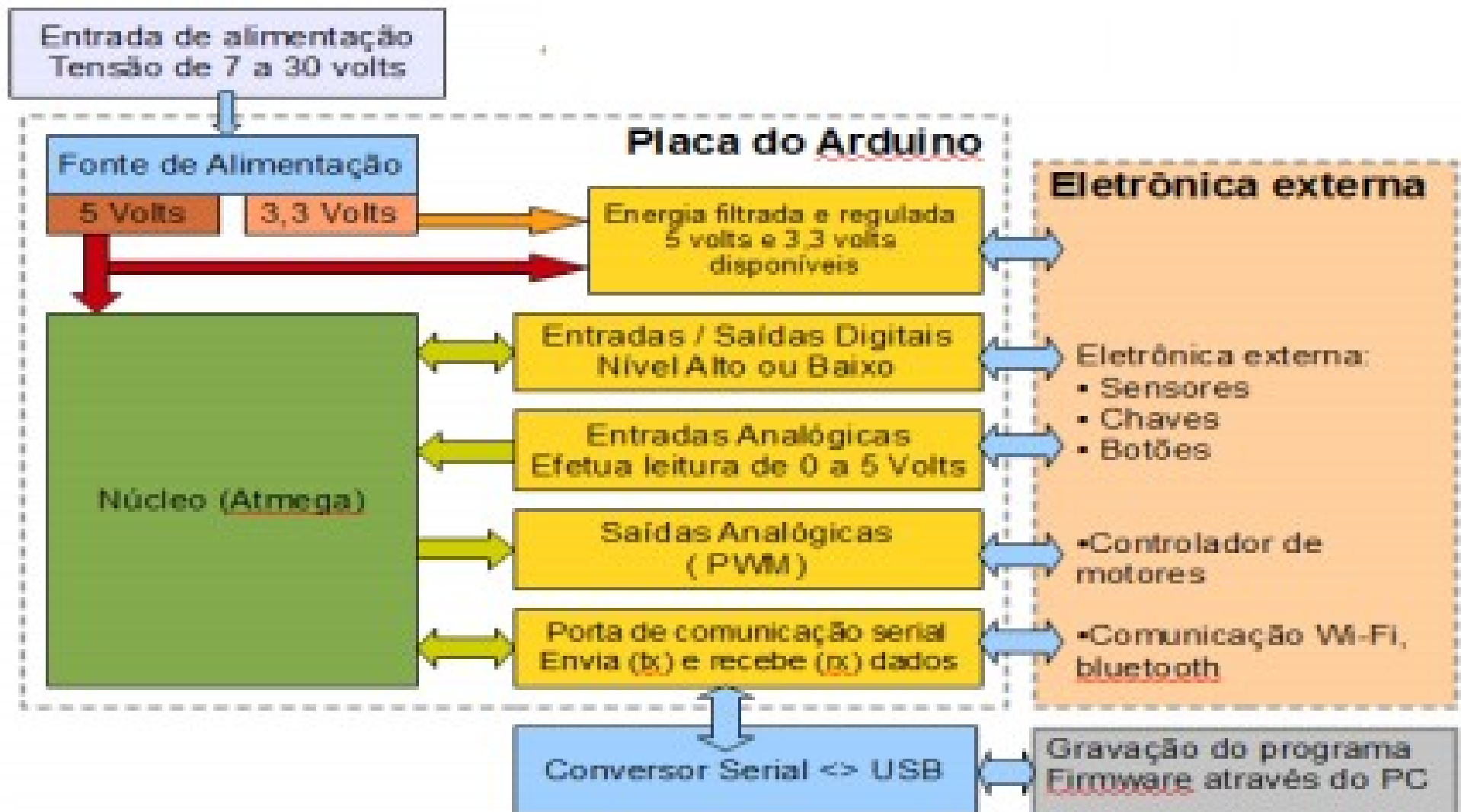
Motivação do Protótipo



Arduino Mega 2560



Arduino Mega 2560



Módulo Xbee e o IEEE 802.15.4

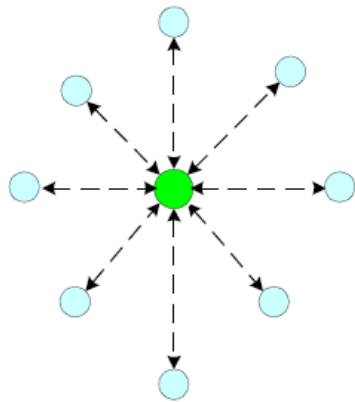
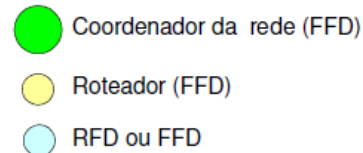
- Tecnologia baseada no padrão IEEE 802.15.4/ZigBee Alliance.
- XBee é a marca da Digi International® Inc. (Digi®) International para uma família de módulos de rádio. Os primeiros rádios XBee foram introduzidos sob a marca MaxStream em 2005 e foram baseados no padrão 802.15.4/2003 projetado para ponto-a-ponto e comunicação estrela para taxas de transmissão “over-the-air” de 250 kb/s.
- ANATEL - Certificado de Homologação do XBee - N°: 0991-10-1209 Emissão: 14/05/2010.

Características técnicas básicas:					
Faixa de Frequências Tx (MHz)	Potência Máxima de Saída (W)	Designação de Emissões	Tecnologia	Tipo de Modulação	Taxa de Transmissão
2400,0 a 2483,5	0,00125	1M56X9D	Sequência Direta	O-QPSK	Até 230,4k bit/s

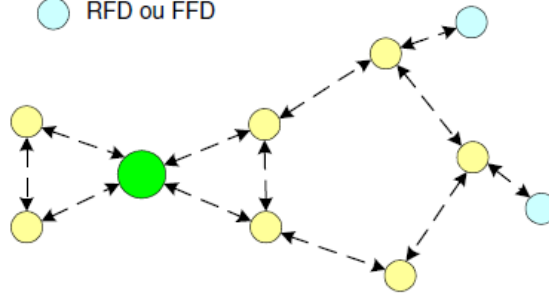
Frequency (MHz)	Modulation Scheme	Bit Rate (kb/s)	Symbol Rate (ksymbols/s)	Channels
868	BPSK	20	20	1
902 to 928	BPSK	40	40	10
2400 to 2483.5	OQPSK	250	62.5	16

Módulo Xbee e o IEEE 802.15.4

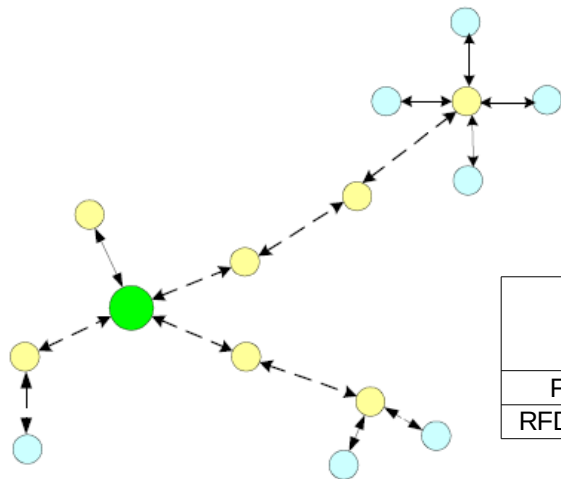
Legenda



(a) topologia estrela



(b) topologia ponto a ponto ou malha



(c) topologia árvore (Cluster-tree)

Coordenador ZigBee	Roteador ZigBee	Dispositivo Final ZigBee	Função na Camada de Rede
X			Estabelecer uma nova rede ZigBee
X	X		Conceder endereço lógico de rede
X	X		Permitir que dispositivos entrem ou saiam da rede
X	X		Manter lista de vizinhos e rotas
X	X		Rotear pacotes da camada de rede
X	X	X	Transferir pacotes da camada de rede

Tipos de dispositivos	Funcionalidade disponíveis no protocolo	Fonte de alimentação típica	Configuração típica do receptor
FFD – Full Function Device	A maioria ou todas	Principal	Ligado quando em espera
RFD – Reduced Function Device	Limitada	Bateria	Desligado quando em espera

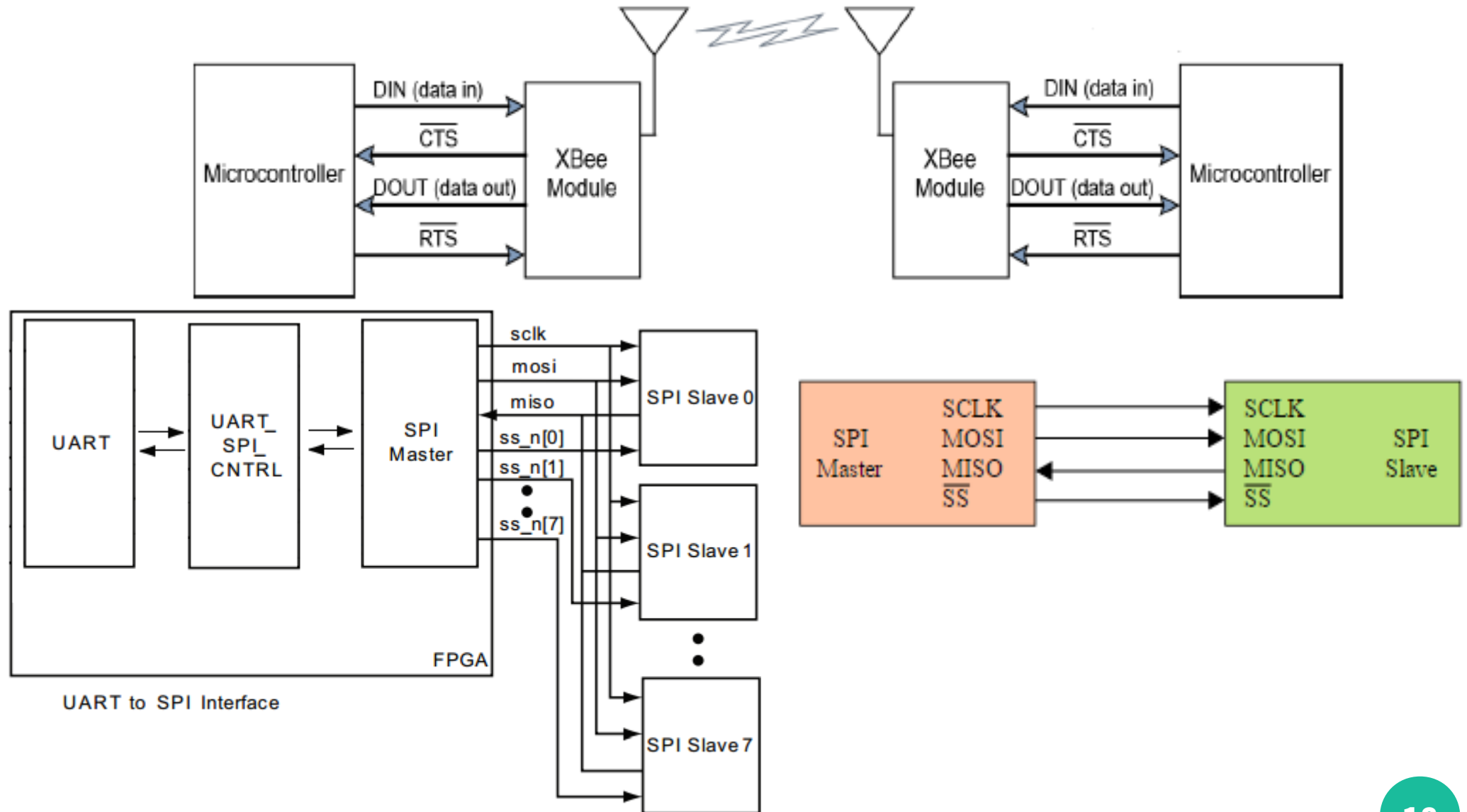
Módulo Xbee

BIT	PINO	FUNÇÃO
	1	VCC (3.3V)
	2	DOUT - SAIDA
	3	DIN-ENTRADA
12	4	DIO12
	5	RESET
10	6	RSSI/DIO10
11	7	DIO11
	8	RESERVADO
8	9	SLEEP/DIO8
	10	TERRA



FUNÇÃO	PINO	BIT
AD0 /DIO0	20	0
AD1 /DIO1	19	1
AD2 /DIO2	18	2
AD3 /DIO3	17	3
RTS /DIO6	16	6
ASSOC /DIO5	15	5
VREF	14	
ON/SLEEP	13	
CTS /DIO7	12	7
DIO4	11	4

Módulo Xbee



Módulo GSM e a Tecnologia GSM

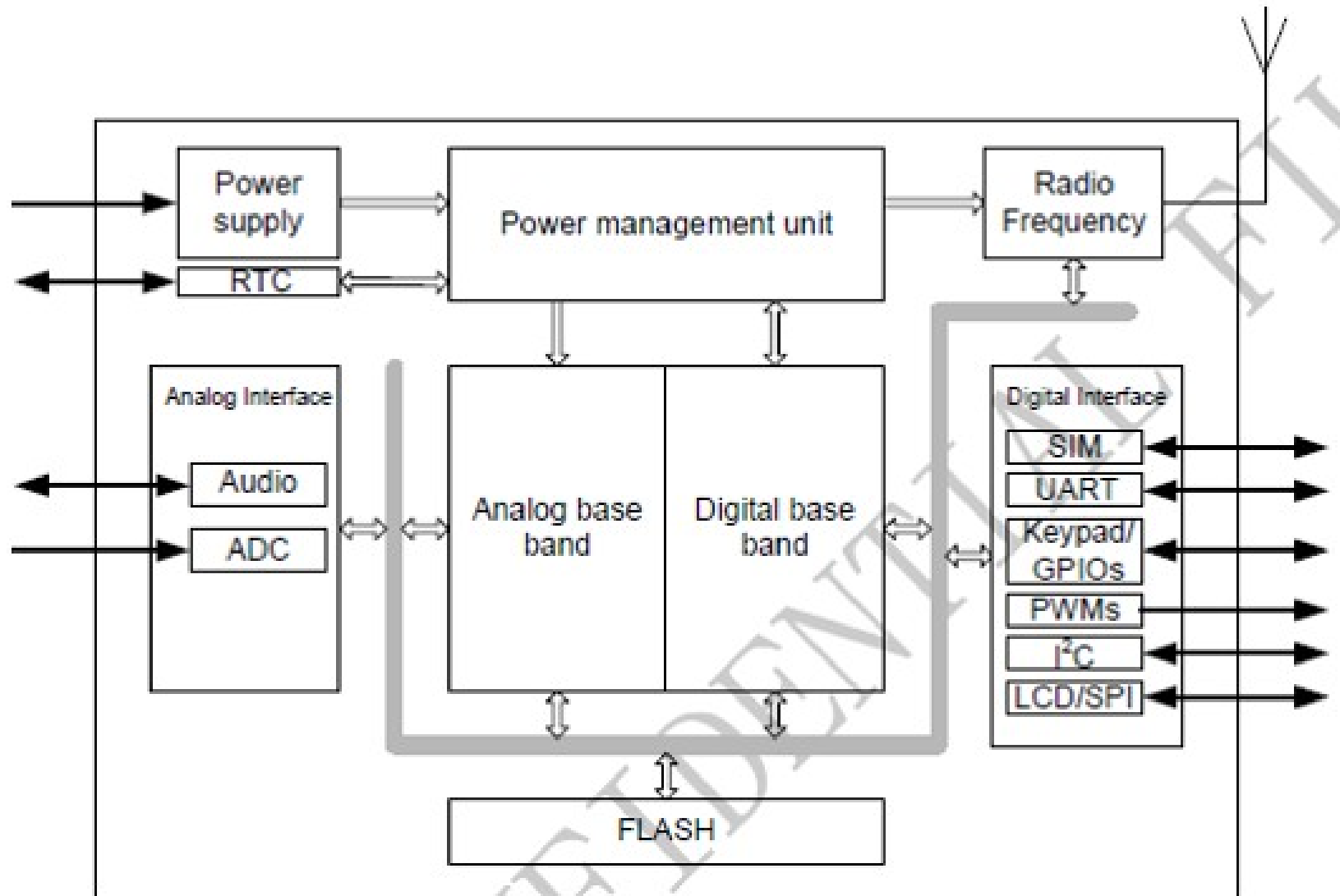
- O SIM900 é um módulo GSM/GPRS quadband com pilha TCP/IP embutida e é do tipo board-to-board (fácil soldagem e integração, até mesmo para prototipagem) e pode ser usado em aplicações onde a transmissão se dê via tecnologia GSM/GPRS, seja voz, SMS, ou dados. Consome pouca energia, corroborando ao tema do trabalho. Seu pequeno tamanho o torna ideal para os mais exigentes requisitos das aplicações industriais, como M2M (Machine-to-Machine), telemetria ou qualquer outra forma de comunicação móvel. É fabricado pela SIMCOM (SHANGHAI SIMCOM LTD.) e distribuído no Brasil pela ME Componentes.
- O Módulo SIM900 segue o padrão 3GPP TS 05.07. ANATEL - Certificado de Homologação do SIM 900 N°: 0685-14-5900 Emissão: 27/02/2014

Características Básicas – ANATEL					
Frequência	Típica Sensibilidade na Recepção	Máxima Sensibilidade na Recepção	Potência Máxima de Saída (W)	Recepção	Transmissão
GSM850	-109 dBm	-107 dBm	1,9054	869 ~ 894 MHz	824 ~ 849 MHz
EGSM900	-109 dBm	-107 dBm	1,8967	925 ~ 960 MHz	880 ~ 915 MHz
DCS1800	-109 dBm	-107 dBm	0,8472	1805 ~ 1880 MHz	1710 ~ 1785 MHz
PCS1900	-109 dBm	-107 dBm	0,8433	1930 ~ 1990 MHz	1850 ~ 1910 MHz

Shield produzido pela ElecFreaks



Diagrama lógico Shield SIM900



Shield produzido pela Elec Freaks

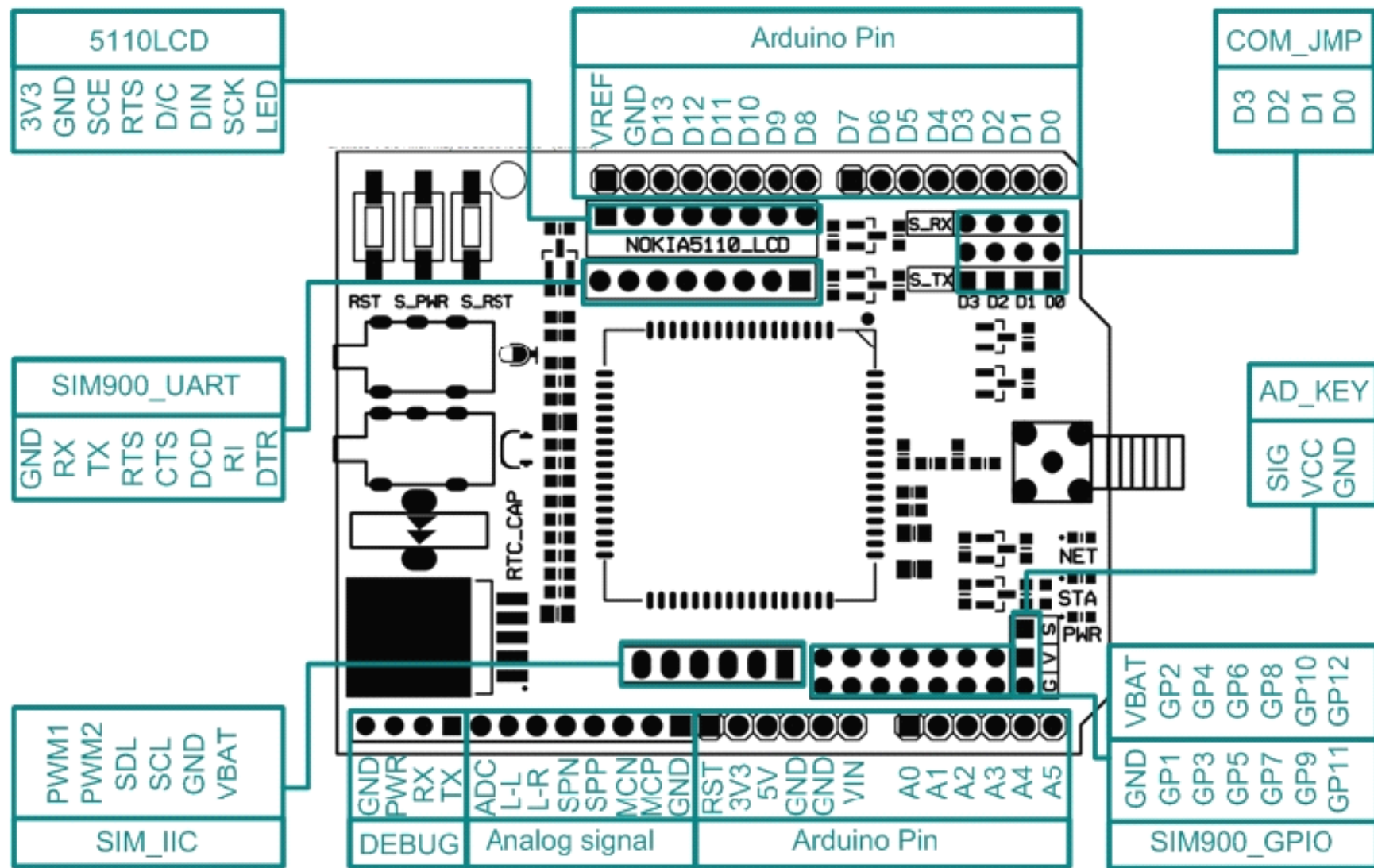


Diagrama do circuito RX-TX-RX Mega2560-SIM900-XBee

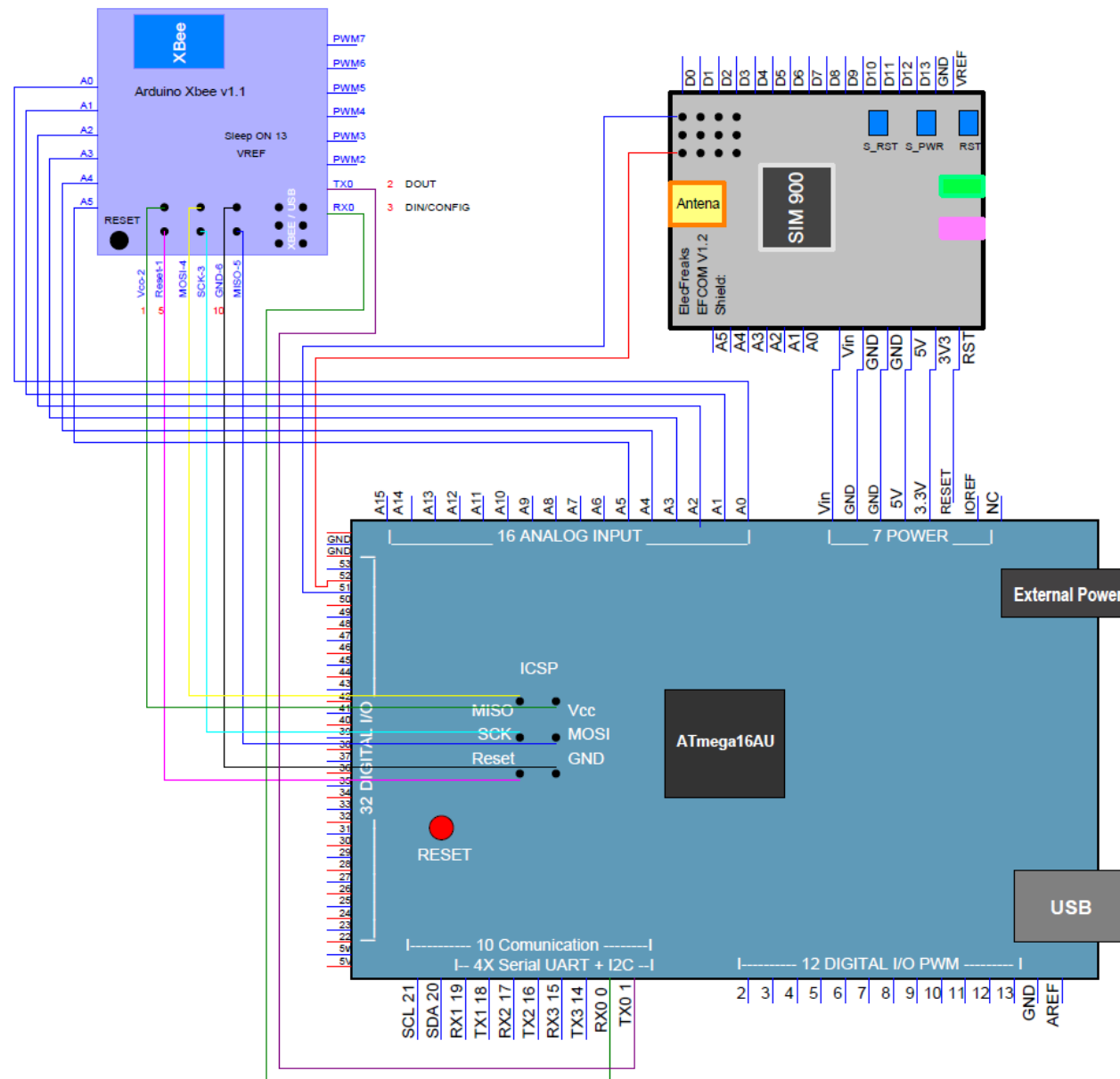
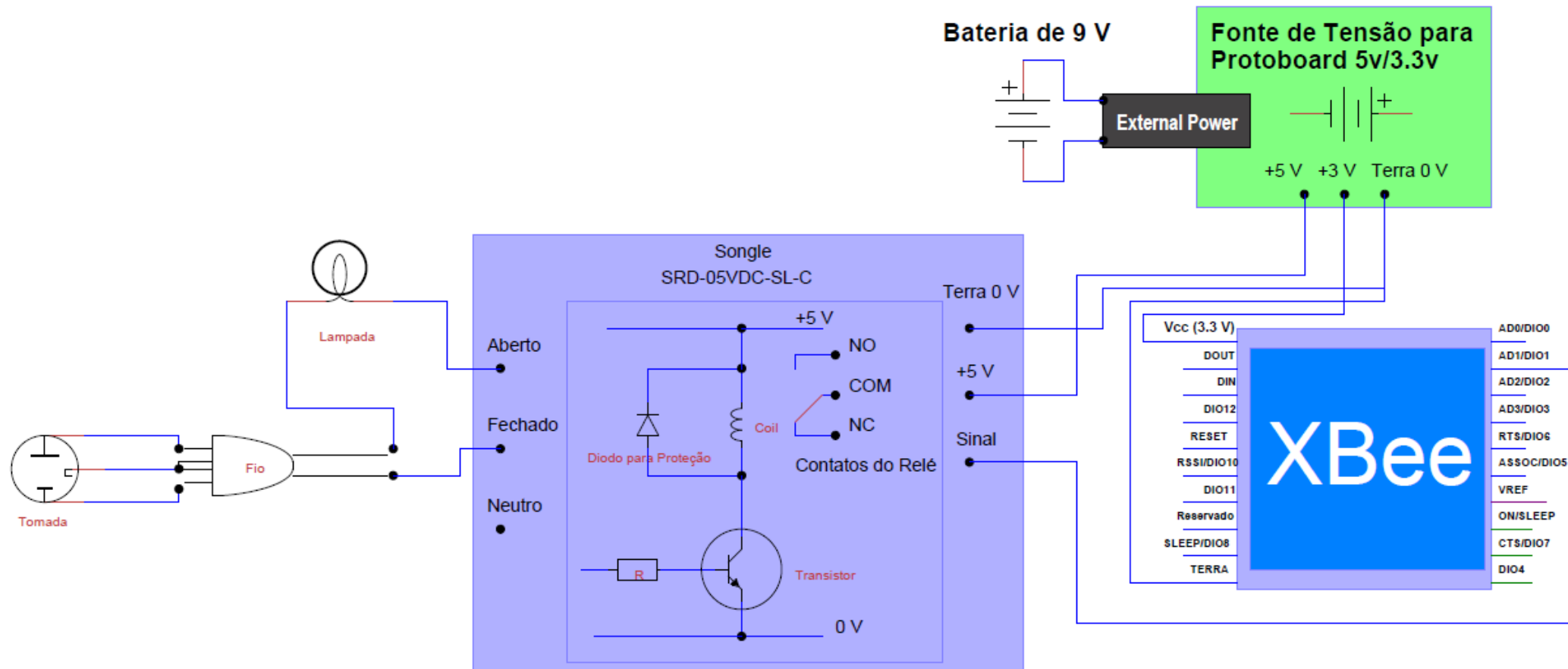


Diagrama do circuito RX-TX-RX XBee-Relé-Lâmpada



Arduino e suas Bibliotecas

- No nível de software, as bibliotecas do Arduino permite que qualquer outra linguagem de programação com capacidade de se comunicar em série nativamente, para não precisar de intermediário, exemplos Python, Processing, C/C++, Linux TTY etc.
- O ambiente de desenvolvimento integrado Arduino (IDE) é uma aplicação multiplataforma escrita em Java, derivada dos projetos Processing e Wiring.
- Um programa ou código escrito para Arduino é chamado de "Sketch".
- Os plugins Cordova para Android e etc é um ótimo exemplo da integração das bibliotecas. Cria toda uma ponte para proporcionar a interoperabilidade das tecnologias atuais, com os dispositivos móveis (smartphones).
- Essa interoperabilidade é caracterizada desde a comunicação serial até esses aplicativos híbridos para os dispositivos móveis, através desses plugins.

Código desenvolvido para o trabalho

Dois códigos:

- Apêndice III – ENVIANDO SMS AO ARDUINO

Biblioteca: SoftwareSerial.h

Comandos AT

- Apêndice IV: I/O Line Passing - 2 XBees (TX/RX)

Comandos AT

Comunicação entre o shield GSM e Arduino Mega 2560

/* EFcom/GPRS Shield Significado dos LED's.

LED	STATUS		DESCRIÇÃO
-----	-----		-----
PWR	ON	OFF	Power ON/OFF do Shield EFcom
STA	ON	OFF	Power ON/OFF do SIM900
NET		OFF	SIM900 Não está em operação
NET	ON(64ms)	OFF(800ms)	SIM900 Não está registrado da rede do GSM
NET	ON(64ms)	OFF(3000ms)	SIM900 Está registrado da rede do GSM
NET	ON(64ms)	OFF(300ms)	O Serviço GPRS está estabilizada

*/

Comunicação entre o shield GSM e Arduino Mega 2560

```
#include <SoftwareSerial.h>
```

```
SoftwareSerial mySerial(50, 51);
```

```
String linhaAtual = "";
```

```
String sms = "";
```

```
String smsAnterior = "";
```

```
boolean lendoSMS = false;
```

```
int led = 30;
```

```
void setup()
```

```
{
```

```
  mySerial.begin(9600);
```

```
  Serial.begin(9600);
```

```
  pinMode(led, OUTPUT);
```

```
  //ligandoModulo();
```

```
  Serial.println("Configurando SMS modo texto");
```

```
  mySerial.println("AT+CMGF=1");
```

```
  mostraDadosSerial();
```

```
}
```

```
void mostraDadosSerial()
```

```
{
```

```
  while(mySerial.available() != 0)
```

```
  {
```

```
    Serial.write(mySerial.read());
```

```
  }
```

```
}
```

```
/*
```

```
void ligandoModulo()
```

```
{
```

```
  Serial.println("Ligando/Reiniciando Modulo GSM...");
```

```
  if(digitalRead(6) == LOW)
```

```
  {
```

```
    digitalWrite(6, LOW);
```

```
    delay(300);
```

```
    digitalWrite(6, HIGH);
```

```
    delay(15000);
```

```
  }
```

```
  Serial.println("Modulo Ligado!");
```

```
}
```

```
*/
```



```

void loop()
{
  //Serial.println("Fazendo leitura do 1 SMS");
  mySerial.println("AT+CMGR=1");
  mostraDadosSerial();
  delay(1000);
  while (mySerial.available()>0)
  {
    char inChar = mySerial.read();
    linhaAtual += inChar;
    if (inChar == '\n')
    {
      linhaAtual = "";
    }
    if (linhaAtual.endsWith("@"))
    {
      lendoSMS = true;
      sms = "";
    }
    if (lendoSMS)
    {
      if (inChar != '<')
      {
        sms += inChar;
      }
    }
    else
    {

```

```

      lendoSMS = false;
      Serial.println(sms);

      if(sms=="@Liga LED" && smsAnterior!=sms)
      {
        Serial.println("Ligando LED");
        digitalWrite(led, HIGH);
      }

      if(sms=="@Desliga LED" && smsAnterior!=sms)
      {
        Serial.println("Desligando LED");
        digitalWrite(led, LOW);
      }

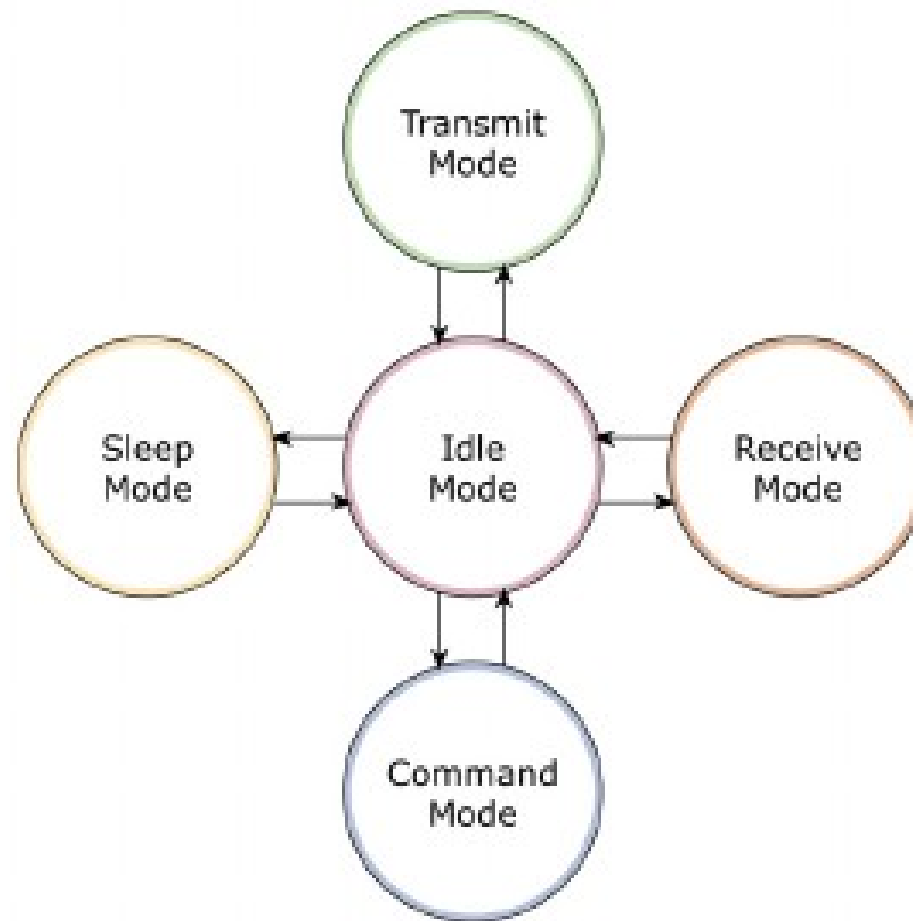
      if(sms == smsAnterior)
      {
        mySerial.println("AT+CMGD=1,4");
        sms="";
      }
      smsAnterior=sms;
    }
  }
  mySerial.println("");
}

```

Comunicação entre o XBee TX e o XBee RX

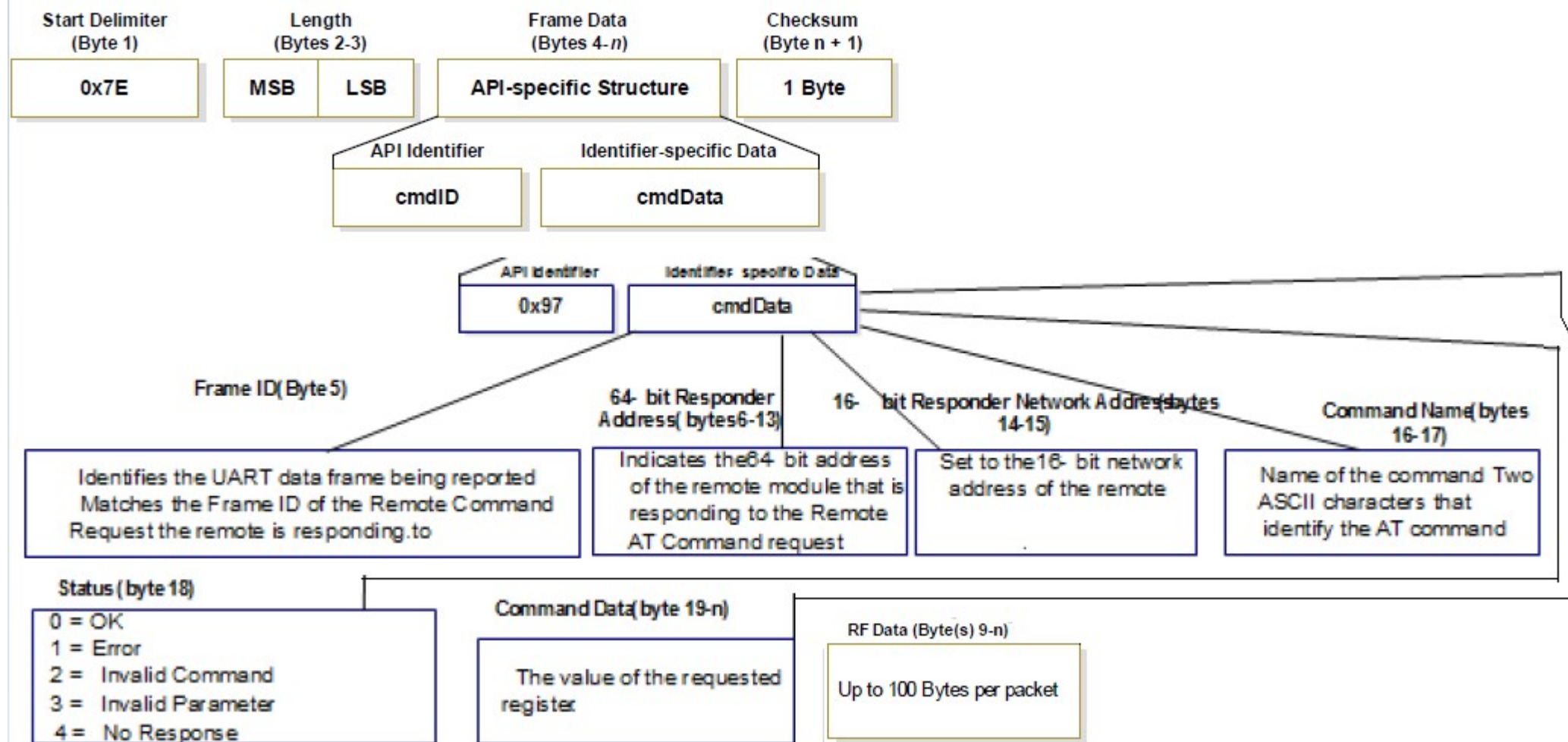
- Para configurar o XBee, existe o software do fabricante, o XCTU. Entretanto, optou-se por fazer configuração por um outro software terminal serial. O software Tera Term.
- Existem alternativas para configurar esse tipo de comunicação e pode ser usado em conjunto com o modo API, que é a principal modo de comunicação.
- O módulo XBee Série 1 suporta um simples recurso chamado I/O line passing, que permite usar qualquer um dos 8 pinos (DIO 0 até 7) para detectar mudanças do estado lógico de Alta($\sim 5\text{v}$) para Baixa ($\sim 3\text{v}$) do módulo XBee em um outro pino correspondente no módulo do XBee receptor. Basicamente espelhará o estado lógico, como criação de um “fio virtual” livre de qualquer apoio de microcontrolador externo ou de análise de série.

Modos de Operação



API (Application Programming Interface)

UART data frame structure:



Configuração do XBee TX

XBee1: XBee Base	
+++	(Entra em modo de configuração)
ATIDFAB	(Personal Area Network ID: 0xFAB)
ATMY8765	(16-bit Source Address: 0x8765)
ATDL4321	(Destination Address Low: 0x4321)
ATBD3	(Define 9600 Baud)
ATD33	(Configuração DI (digital Input) (3))
ATD23	(Configuração DI (digital Input) (3))
ATD13	(Configuração DI (digital Input) (3))
ATD03	(Configuração DI (digital Input) (3))
ATIC	(DIO Chance Detect: F)
ATNIXBee Base	(Node Identifier: XBee Base)
ATIU0	(I/O Output Disabled)
ATWR	(Grava a config.)
ATCN	(Sai do Modo Configuração)

Configuração do XBee RX

XBee2: XBee Remoto

+++	(Entra em modo de configuração)
ATIDFAB	(Personal Area Network ID: 0xFAB)
ATMY4321	(16-bit Source Address: 0x4321)
ATDL8765	(Destination Address Low: 0x8765)
ATNIXBee Remoto	(Node Identifier: XBee Remoto)
ATD33	(Configuração DO (digital Output) LOW (4))
ATD23	(Configuração DO (digital Output) LOW (4))
ATD13	(Configuração DO (digital Output) LOW (4))
ATD03	(Configuração DO (digital Output) LOW (4))
ATIC	(DIO Chance Detect: F)
ATCE0	(CE Coordenador [1] End Device [0])
ATIA8765	(Setting address to 0xFFFF will allow any received I/O packet to change outputs.)
ATWR	(Grava a config.)
ATCN	(Sai do Modo Configuração)

Relacionando as configuração entre o XBee TX e o XBee RX

Parametros	Xbee Base	Xbee Remoto
CH Channel	C	C
ID PAN ID	FAB	FAB
DH Destination Address High	0	0
DL Destination Address Low	4321	8765
MY 16-bit Source Address	8765	4321
NI Note Identifier	Xbee Base	Xbee Remoto
Sleep Mode	No Sleep [0]	No Sleep [0]
BD Interface Data Rate	9600 [3]	9600 [3]
AP API Enable	API disable [0]	API disable [0]
D3 DIO3 Configuration	DI [3]	DO Low [4]
D2 DIO2 Configuration	DI [3]	DO Low [4]
D1 DIO1 Configuration	DI [3]	DO Low [4]
D0 DIO0 Configuration	DI [3]	DO Low [4]
IU I/O Output Enable	Disabled [0]	Disabled [0]
IC DIO Change Detect	F	F
IR Sample Rate * 1 ms	14	0
IA I/O Input Address	FFFFFFFFFFFFFFFF	8765
T0 D0 Output Timeout	FF	FF
T1 D1 Output Timeout	FF	FF
... T7 D7 Output Timeout	FF	FF

Todos os parâmetros do XBee

XBee1: XBee - Base

```
<?xml version="1.0" encoding="UTF-8"?>

<data>
  <profile>
    <description_file>xb24_15_4_10ef.xml</description_file>
    <settings>
      <setting command="CH">C</setting>
      <setting command="ID">FAB</setting>
      <setting command="DH">0</setting>
      <setting command="DL">4321</setting>
      <setting command="MY">8765</setting>
      <setting command="MM">0</setting>
      <setting command="RR">0</setting>
      <setting command="RN">0</setting>
      <setting command="NT">19</setting>
      <setting command="NO">0</setting>
      <setting command="CE">0</setting>
      <setting command="SC">1FFE</setting>
      <setting command="SD">4</setting>
      <setting command="A1">0</setting>
      <setting command="A2">0</setting>
      <setting command="EE">0</setting>
      <setting command="KY"></setting>
      <setting command="NI">XBee BASE</setting>
      <setting command="PL">4</setting>
      <setting command="CA">2C</setting>
      <setting command="SM">0</setting>
      <setting command="ST">1388</setting>
      <setting command="SP">0</setting>
      <setting command="DP">3E8</setting>
      <setting command="SO">0</setting>
      <setting command="BD">3</setting>
      <setting command="NB">0</setting>
      <setting command="RO">3</setting>
      <setting command="AP">0</setting>
      <setting command="D8">0</setting>
      <setting command="D7">1</setting>
      <setting command="D6">0</setting>
      <setting command="D5">1</setting>
      <setting command="D4">0</setting>
      <setting command="D3">3</setting>
      <setting command="D2">3</setting>
      <setting command="D1">3</setting>
      <setting command="D0">3</setting>
      <setting command="PR">FF</setting>
      <setting command="IU">0</setting>
      <setting command="IT">1</setting>
      <setting command="IC">F</setting>
      <setting command="IR">14</setting>
      <setting command="P0">1</setting>
      <setting command="P1">0</setting>
      <setting command="PT">FF</setting>
      <setting command="RP">28</setting>
      <setting command="IA">FFFFFFFFFFFFFFFF</setting>
      <setting command="T0">FF</setting>
      <setting command="T1">FF</setting>
      <setting command="T2">FF</setting>
      <setting command="T3">FF</setting>
      <setting command="T4">FF</setting>
      <setting command="T5">FF</setting>
      <setting command="T6">FF</setting>
      <setting command="T7">FF</setting>
      <setting command="DD">10000</setting>
      <setting command="CT">64</setting>
      <setting command="GT">3E8</setting>
      <setting command="CC">2B</setting>
    </settings>
  </profile>
</data>
```

Dificuldades encontradas

- As bibliotecas para Arduino que a comunidade desenvolve pode não funcionam 100% com os Shields desenvolvidos para o Arduino, então os fabricantes estão adaptando algumas bibliotecas fornecidas junto aos produtos. Logo, pode não funcionar todas as funções nativas ou com outras bibliotecas em conjunto.
- A exemplo, foi necessário mudar o valor de uma variável de 64 para 256 para não ter problema com o módulo do GSM.

Biblioteca: SoftwareSerial.h

```
#define _SS_MAX_RX_BUFF 64(256) // RX buffer size
```

- Houve bastante dificuldade em se utilizar o modo API para comunicação do XBee, então a programação foi adaptada para o modo I/O Line Passing para a comunicação entre os módulos XBee.

Automação Industrial e Residencial - OS DEZ MAIORES DESAFIOS DA AUTOMAÇÃO INDUSTRIAL AS PERSPECTIVAS PARA O FUTURO

- Formação Técnica de Profissionais e Educação da Sociedade quanto à Evolução Tecnológica Proporcionada pela Automação.
- Segurança e Confiabilidade em Sistemas Críticos.
- Otimização de Informações, no Sentido de Fornecer uma Interface Homem-Máquina Adequada.
- Reconhecimento de Padrões.
- Identificação de Falhas em Sistemas de Automação.
- Comunicação Segura entre Dispositivos Heterogêneos.
- Sistemas de Automação Residencial.
- Gerência de Informações de Tempo Real.
- Aplicações na Área de Medicina.
- Impactos Sociais e Ambientes Gerados pela Automação.

Conclusão

- O projeto foi bem-sucedido no sentido de se desenvolver um protótipo plenamente funcional, demonstrando sua aplicabilidade a uma situação prática.
- Foi possível apresentar todos os conceitos envolvidos na proposta do tema do trabalho através de uma versão simplificada do sistema, no qual uma carga – no caso, uma lâmpada – liga e desliga por comandos SMS, responsáveis por acionar o relé ao qual ela está conectada, a qualquer distância dentro da área de cobertura 2G da operadora.
- A arquitetura de rede proposta foi plenamente implementada, à exceção da comunicação entre o Arduino e PC. Uma vez estabelecida essa comunicação, torna-se possível verificar os dados das variáveis e criar variáveis adicionais, como contadores, etc. Nesse sentido, é possível, sim, pensar em criar uma gerência para verificar ao menos os eventos.
- A viabilidade foi pesquisada e estudada antes da realização do presente trabalho, e entendemos ser este um objeto de estudo interessante para trabalhos futuros. Para implementar essa gerência, o requisito é realizar a comunicação do XBee receptor para o XBee transmissor, ou seja, a comunicação deve ser bidirecional. No protótipo que apresentamos, a comunicação se dá de forma unidirecional. Para isso somente no modo API.

Bibliografia

- As referencias que foi utilizado para criar os slides encontra-se na parte escrita do trabalho.
- [1] ZigBee Automated Compliance Tests
<http://www.ni.com/webcast/127/en/>
<https://decibel.ni.com/content/docs/DOC-19406>
<https://ni.adobeconnect.com/p58178044/>
- [2]RFDN Basic of ZigBee Transmitter Testing
<https://ni.adobeconnect.com/p79879863/>
<http://www.ni.com/video/383/en/>

Bibliografia

- [3] Microcontrolador 100% Nacional é Marco Histórico no Cenário Brasileiro de Microeletrônica
<http://www.ci-brasil.gov.br/index.php/pt/noticias2/150-microcontrolador-100-nacional-e-marco-historico-no-cenario-brasileiro-de-microeletronica>
- [4] <http://www.zigbee.org/zigbee-in-space-xbee-rf-modules-launched-by-nasa/>

!!! Obrigado Pela Atenção !!!

?? Perguntas ??

!!! FIM !!!

Apêndice

- 1 – Custo
- 2 – Equipamentos para testar
- 3 – Microcontrolador Brasileiro
- 4 – Curiosidades

Custo - Hardware Utilizado

Materiais	Total	Valor	Modelo	Revendedora	Fabricante
Protoboard	x1	R\$ 70,00	Protoboard 2420 Furos 4 Bornes MSB-400 ICEL	http://www.ferramentaskennedy.com.br/	ICEL Manaus
Arduino	x1	US\$ 60.20	Arduino Mega 2560 mais acessórios para Fundino	http://www.dx.com/ – MVProduct	Fundino
Modulo GSM/GPRS para Arduino	x1	US\$ 59.10	SIM900 (GSM/GPRS)	http://www.dx.com/	ElecFreaks

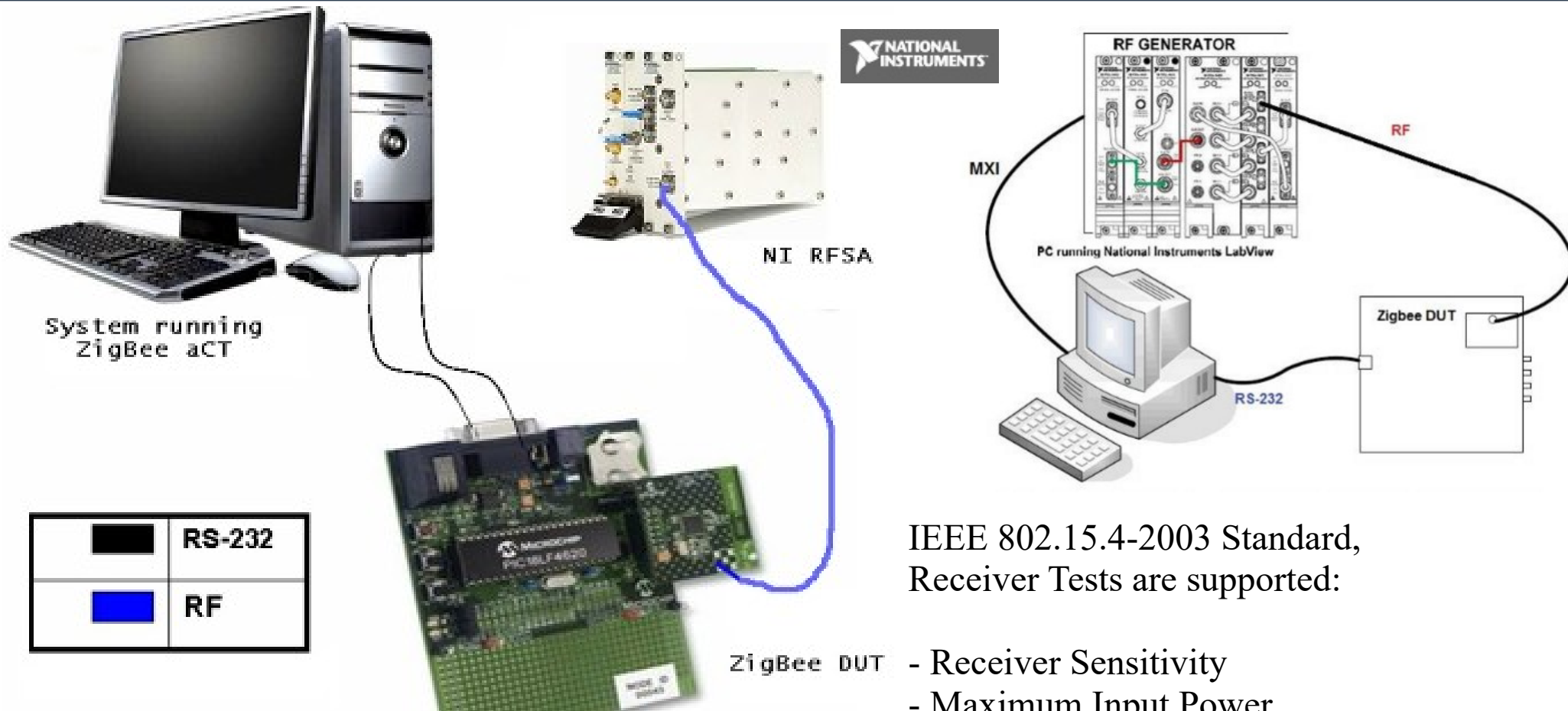
Custo - Hardware Utilizado

Materiais	Total	Valor	Modelo	Revendedora	Fabricante
Fonte para Protoboard	x1	R\$ 35,00	Fonte de alimentação para Protoboard 5v / 3.3v:	https://multilogica-shop.com/	
Bateria	x1	R\$ 20,00	Bateria Duracel de 9 Volts	Kalunga	Duracel
Xbee Shield Para Arduino	x1	R\$ 44,90	Xbee Shield v.14 – XBE0003	TECHMOUNT	DFRobot
Módulos Xbee	x2	US\$ 81.00	XB24-AUI-001 – Modulo Xbee de 1mw Série 10	Ebay – Duino Shop E Sparkfun	Digi E Maxstream

Existe Microcontroladores fabricado no Brasil com tecnologia nacional?

- Microcontrolador 100% Nacional é Marco Histórico no Cenário Brasileiro de Microeletrônica
- “A apresentação da família ZR16 na 2ª Edição da Embedded System Conference Brazil - ESC 2012 representou um marco histórico para o cenário brasileiro de microeletrônica”. A afirmação é de João Baptista dos S. Martins, Coordenador da Santa Maria Design House (SMDH) que, juntamente com a Chipus Microeletrônica, é responsável pelo projeto. “Pela primeira vez foi apresentado um chip brasileiro com potencial de mercado nacional e internacional”, complementou. (Fonte: Priscila Franco de Lima - 05/07/2012)

Testes com protocolo ZigBee



IEEE 802.15.4-2003 Standard,
Receiver Tests are supported:

- Receiver Sensitivity
- Maximum Input Power
- Adjacent Channel Jamming Resistance
- Alternate Channel Jamming Resistance

Gerador de Sinais Vetorial de RF

Referência [1]

Testes com protocolo ZigBee

RFDN Basic of ZigBee Transmitter Testing [2]
Software provides by SeaSolve Software

ZigBee Transmitter are typified by:

- Use of OQPSK modulation
- Low Power
- Medium modulation quality(measured as EVM)

Power Spectral Density (PSD)

- Measures power emitted in adjacent channels
- ZigBee standard allows -30dBm in adjacent channels
- Performed as complex FFT

EVM per Symbol

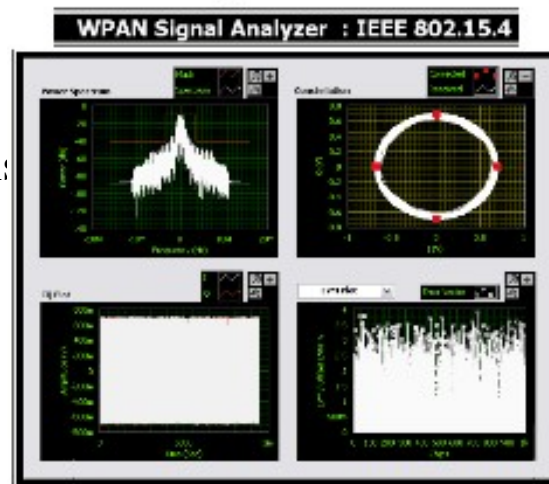
- Measures modulation accuracy
- ZigBee operates with up to 35% EVM
- Lower EVM yields better BER

Constellation Diagram

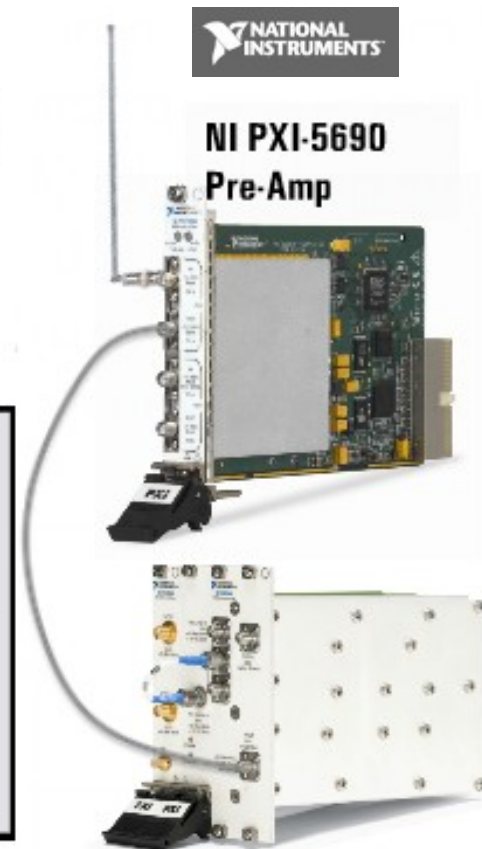
- All transition around perimeter (OQPSK)
- Use to troubleshoot impairments
 - AWGN
 - Quadrature Skew
 - IQ Gain imbalance



Freescall ZigBee EVM



SeaSolve LVSA
WiPAN Software



NI PXI-5661 VSA

NI PXI-5690: Pré-amplificador de RF

NI PXI-5661: Analisador vetorial de sinais de RF

Zigbee in space XBee RF modules launched by NASA

July 7, 2015: A historic day for ZigBee technology and Digi International, NASA successfully launched XBee RF modules into space in a sounding rocket. [4]

