



Universidade Federal  
de Campina Grande

**Centro de Engenharia Elétrica e Informática  
Programa de Pós Graduação em Engenharia Elétrica**

## **DISSERTAÇÃO DE MESTRADO**

**Automação e Monitoramento de Reatores de Bateladas  
Sequenciais para Tratamento de Águas Residuárias**

**Aluno: Icaro Gonzalez da Silva**

**Orientador: Raimundo Carlos Silvério Freire**

**Co-orientador: Sebastian Yuri Cavalcanti Catunda**

**Campina Grande – PB  
Maio – 2015**



Universidade Federal  
de Campina Grande

**Centro de Engenharia Elétrica e Informática  
Programa de Pós Graduação em Engenharia Elétrica**

## Automação e Monitoramento de Reatores de Bateladas Sequenciais para Tratamento de Águas Residuárias

*Dissertação de Mestrado apresentado à  
Coordenadoria do Programa de Pós-  
Graduação em Engenharia Elétrica da  
Universidade Federal de Campina  
Grande, em cumprimento às exigências  
para obtenção do Grau de Mestre em  
Ciências no Domínio da Engenharia  
Elétrica.*

**Área de Concentração: Processamento da Informação**

**Raimundo Carlos Silvério Freire**

Orientador

**Sebastian Yuri Cavalcanti Catunda**

Orientador

**Campina Grande – PB  
Maio – 2015**

*Aos meus pais, José Mario Ribeiro da Silva (In Memoriam) e Lenira Dorcinda Gonzalez da Silva, que sempre acreditaram no meu potencial e fizeram o possível e o impossível para que eu me desenvolvesse.*

*À minha Esposa, Thacianny, por todo amor e cumplicidade.*

## **AGRADECIMENTOS**

Aos professores Raimundo Carlos Silvério Freire e Sebastian Yuri Catunda, pela orientação desta pesquisa, além da amizade construída durante estes anos.

Aos professores Benedito Antonio Luciano e Raimundo Carlos Silvério Freire, pelas aulas ministradas durante o mestrado.

Aos colegas Vanderson de Lima Reis e Petrov Crescencio Lobo, pela grande paciência, dedicação, tempo e contribuição para a execução deste trabalho de dissertação.

Ao professor Adrianus Van Haandel pela sua colaboração no desenvolvimento deste projeto.

Ao CNPq, Conselho Nacional de Desenvolvimento Tecnológico, pela concessão da bolsa de pesquisa para a execução desta pesquisa.

Aos membros da Coordenação e Colegiado do Programa de Pós Graduação em Engenharia Elétrica da UFCG, por conceder os pedidos solicitados.

À UFCG, Universidade Federal de Campina Grande, pelo acolhimento e pela oportunidade em cursar o mestrado do Programa de Pós-Graduação em Engenharia Elétrica.

*“Determinando tu algum negócio, seja firme, e a luz  
brilhará em teus caminhos”.*

Jó 22:28

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
CISC	<i>Complex Instruction Set Computer</i>
DSSS	<i>Direct Sequence Spread Spectrum</i>
EEPROM	<i>Electrically Erasable Programmable Read-Only Memory</i>
EPROM	<i>Erasable Programmable Ready Only Memory</i>
FFD	<i>Full Function Device</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
OD	Oxigênio Dissolvido
PIC	<i>Peripheral Interface Controller</i>
PWM	<i>Pulse-Width Modulation</i>
RAM	<i>Random Access Memory</i>
RISC	<i>Reduced Instruction Set Computer</i>
ROM	<i>Read-Only Memory</i>
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
USART	<i>Universal Synchronous Asynchronous Receiver Transmitter</i>
USB	<i>Universal Serial Bus</i>
WiFi	<i>Wireless Fidelity</i>
WPAN	<i>Wireless Personal Area Network</i>
RF	Radio Frequência
RFD	<i>Reduced Function Devices</i>
SH	<i>Serial Number High</i>
SL	<i>Serial Number Low</i>
CC	Ciclos de <i>Clock</i>
UML	<i>Unified Modelling Language</i>
POO	Programação Orientada a Objeto

SGBD

Sistema de Gerenciamento de Banco de Dados

EPIs

Equipamentos de Proteção Individual

## LISTA DE FIGURAS

Figura 1 Ciclo de tratamento pelo sistema em RBS.	17
Figura 2 Reator do RBS com os sensores e atuadores.	17
Figura 3 Equipamentos que utilizam microcontrolador.	20
Figura 4 Arquitetura de Von Neumann.	20
Figura 5 Arquitetura de Harvard.	21
Figura 6 Estruturas principais dos microcontrolador.	21
Figura 7 Componentes do microcontrolador.	22
Figura 8 Estrutura para utilização do relógio[12].	23
Figura 9 Estrutura dos pinos do microcontrolador PIC18F4550[12].	25
Figura 10 Estrutura em blocos da aproximação sucessiva.	26
Figura 11 Faixa de frequência das ondas acústicas.	28
Figura 12 Transdutores emissor e receptor.	29
Figura 13 Estrutura básica do transdutor ultrassônico (adaptado [13]).	30
Figura 14 Ramificação dos tipos de sensores OD.	30
Figura 15 Composição do sensor OD eletroquímico(adaptado [16]).	31
Figura 16 Polarização do sensor OD eletroquímico(adaptado [16]).	32
Figura 17 Relação alcance e taxa de dados dos padrões (adaptado [21]).	35
Figura 18 Estrutura da topologia estrela.	36
Figura 19 Estrutura da topologia malha.	37
Figura 20 Estrutura da topologia malha.	38
Figura 21 Arquitetura do ZigBee(adaptado de [19]).	38
Figura 22 Principais Modelos do XBee.	40
Figura 23 Estrutura do <i>frame</i> [23].	42
Figura 24 Esquemático do envio de dados pelo modo API.	42
Figura 25 Relação entre usuário e caso de uso.	48
Figura 26 Relação entre atores.	48
Figura 27 Relacionamento entre casos de uso.	48
Figura 28 Exemplo de diagrama de classes.	49
Figura 29 Estrutura dos componentes principais do sistema embarcado.	52
Figura 30 Sensor de OD.	53
Figura 31 Estrutura do diagrama de pinos do sensor OD.	53
Figura 32 Condicionamento do Sensor OD(adaptado [19]).	54
Figura 33 Relação entre a temperatura e a resistência elétrica do sensor.	56
Figura 34 Gráfico relacionando $1/T$ e $\ln(R)$ .	58
Figura 35 Condicionamento do sensor de temperatura (adaptado [19]).	59
Figura 36 Transdutor Ultrassônico (HC-SR04).	61
Figura 37 Funcionamento do HC-SR04 [21]	61
Figura 38 Relação Gráfica entre tempo e distância.	62
Figura 39 Fluxograma do cálculo da distância.	63

Figura 40 Estrutura do Relé.	64
Figura 41 Condicionamento do acionamento do relé.	64
Figura 42 Modelo do display LCD.	65
Figura 43 Condicionamento do Display LCD.	65
Figura 44 Fluxograma do <i>software</i> no microcontrolador.	66
Figura 45 Dispositivo ZigBee XBP24-ZB [32].	67
Figura 46 Programa X-CTU.	68
Figura 47 Estrutura da Rede XBee.	69
Figura 48 Ferramenta para desenvolvimento NetBeans.	70
Figura 49 Diagrama de caso de uso do sistema.	71
Figura 50 Estrutura do banco de dados desenvolvido.	72
Figura 51 EPIs necessárias para o experimento.	72
Figura 52 Tela principal do sistema computacional.	73
Figura 53 Topo da tela principal.	74
Figura 54 Tela de cadastro dos reatores.	75
Figura 55 Tela de busca de Xbee.	75
Figura 56 Tela gerar relatórios.	76
Figura 57 Estrutura do protótipo primeira fase do projeto	77
Figura 58 Placa da fonte de alimentação.	78
Figura 59 Placa da comunicação XBee.	79
Figura 60 Placa central do sistema embarcado.	80
Figura 61 Placa de acionamento dos relés.	80
Figura 62 Placa de condicionamento do sensor OD e temperatura.	81
Figura 63 Sistema embarcado construído.	82
Figura 64 Validação do sensor de temperatura.	83
Figura 65 Verificação do sensor de nível.	84
Figura 66 Validação do sensor de OD.	85
Figura 68 Diagrama de atividades habilitar conexão com XBee.	90
Figura 69 Diagrama de atividades status de transmissão.	90
Figura 70 Diagrama de atividades cadastrar reatores.	91
Figura 71 Diagrama de atividades Pesquisar Reatores.	91
Figura 72 Diagrama de atividades alterar reatores.	92
Figura 73 Diagrama de atividades excluir reatores.	93
Figura 74 Diagrama de atividades Gerar PDF.	93
Figura 75 Diagrama de Classes do sistema computacional.	94
Figura 76 Diagrama elétrico da placa de alimentação parte 1.	95
Figura 77 Diagrama elétrico da placa de alimentação parte 2.	96
Figura 78 Diagrama elétrico do condicionamento do XBee.	97
Figura 79 Diagrama da placa Central do sistema embarcado.	98
Figura 80 Diagrama da placa de acionamento.	99
Figura 81 Diagrama da placa de condicionamento dos sensores.	100

## LISTA DE QUADROS

Quadro 1 Características do PIC18f4550 [12]	26
Quadro 2 Diferenças entre os padrões Wi-Fi, Bluetooth e ZigBee (adaptado [21]).	34
Quadro 3 Bandas de frequência suportadas na camada física.	39
Quadro 4 Exemplo de envio de dados pelo modo API (adaptado de [24]).	43
Quadro 5 Exemplo de recebimento de dados pelo modo API (adaptado de [24]).	44
Quadro 6 Status da transmissão (adaptado de [24]).	45
Quadro 7 Relação da resistência com a temperatura do sensor.	55
Quadro 8 Relação $1/T$ e $\ln(R)$ .	57
Quadro 9 Relação do tempo e distância.	62
Quadro 10 Características do modelo ZigBeeXBP24-ZB [32].	67

## RESUMO

Atualmente, devido ao crescimento populacional e o aumento dos descartes de águas residuárias, estudos estão sendo desenvolvidos com a finalidade de otimizar os métodos de tratamento de esgoto existentes. Dentre estes, destaca-se o regime de bateladas sequenciais – RBS que possui flexibilidade de implementação e eficiência no tratamento. O processo de tratamento do RBS é realizado em reatores com fases de tratamento definidas, as quais são executadas periodicamente de forma cíclica. O advento dos microprocessadores e computadores vem contribuindo para a automação dos processos de tratamento e na redução do consumo de energia. Nesta pesquisa é apresentado um modelo da automação e monitoramento de reatores de bateladas sequenciais para tratamento de águas residuárias. Este sistema é capaz de aferir e analisar a quantidade de oxigênio dissolvido, o nível do esgoto e a temperatura no reator. Também é capaz de adicionar e monitorar os períodos de cada fase de tratamento. O sistema embarcado utiliza o microcontrolador PIC 18F4550, um transceptor sem fio XBee, o transdutor HC-SR04, sensor de oxigênio dissolvido e Temperatura YSI 5739. Os dados medidos no reator são transferidos via RF para um computador pessoal utilizando o protocolo ZigBee, nesse computador é instalado o *software* codificado em JAVA, cuja finalidade é armazenar e apresentar de forma gráfica os dados aos usuários do sistema computacional. O sistema desenvolvido foi caracterizado e validado no laboratório de tratamento de esgoto da UEPB. No processo de validação, os dados aferidos pelos sensores do sistema embarcado foram comparados com equipamentos laboratoriais para verificar a congruência dos dados, em que apresentou pequenos desvios percentuais. O sistema de automação e monitoramento de águas residuárias apresentou um percentual de tratamento de 92% em cada batelada.

**Palavras chaves:** Sistemas RBS, lodos ativados, sistema computacional, rede de sensores sem fio, instrumentação eletrônica, sistemas embarcados.

Nowadays, due to population growth and increased discharges of wastewater, studies are being developed in order to optimize existing sewage treatment methods. Among these, it is highlighted the system of sequencing batch - RBS having implementation flexibility and efficiency in treatment. The RBS treatment process is performed in reactors with a defined processing steps, which are periodically executed cyclically. The advent of microprocessors and computers contributed to the automation of the treatment processes and in reducing energy consumption. In this work, it is presented the automation and monitoring of sequencing batch reactors for wastewater treatment. This system is capable of measuring and analyzing the amount of oxygen dissolved, the sewage level and the temperature in the reactor. It is also able to add and monitor the periods of each treatment phase. The embedded system uses the PIC 18F4550 microcontroller, a XBee wireless transceiver, the HC-SR04 transducer, YSI 5739 OD and temperature sensor. The data measured in the reactor are transferred via RF to a personal computer using the ZigBee protocol, in this computer it is installed the software coded in JAVA, whose purpose is to store and display the data graphically to users of the computer system. The developed system was characterized and validated in the laboratory of wastewater treatment of UEPB. In the validation process, the data measured by the sensors in the embedded system were compared with actual equipment to verify the consistency of the data, which showed small percentage deviations. The automation system and monitoring of wastewater proved satisfactory treatment with 92% percentage in each batch.

**Keywords:** RBS systems, activated mud, computer system, wireless sensor network, electronic instrumentation, embedded systems.

1.	Introdução	13
1.1.	Regime de bateladas sequencias – RBS	16
1.2.	Objetivos	18
1.2.1.	Objetivos Gerais	18
1.2.2.	Objetivos Específicos	18
2.	Fundamentação teórica	19
2.1.	Microcontroladores	19
2.1.1.	Arquitetura do Microcontrolador	20
2.1.2.	Características dos Microcontroladores	22
2.1.3.	Microcontrolador PIC 18F4550	25
2.2.	Sensores	27
2.2.1.	Transdutor Ultrassônico	28
2.2.2.	Sensor de Oxigênio Dissolvido	30
2.2.3.	Sensor de temperatura	32
2.3.	Comunicação Sem Fio	33
2.3.1.	Padrão ZigBee	34
2.3.2.	Redes ZigBee	36
2.3.2.1.	Topologia Estrela ( <i>Star</i> )	36
2.3.2.2.	Topologia Árvore ( <i>Tree</i> )	37
2.3.2.3.	Topologia Malha ( <i>Mesh</i> )	37
2.3.3.	Arquitetura do ZigBee.	38
2.3.4.	Módulos XBee	40
2.3.5.	Modo de Operação do XBee	41
2.3.5.1.	Modo Transparente	41
2.3.5.2.	Modo API	41
2.4.	<i>Software</i>	46
2.4.1.	Linguagem UML	47
2.4.1.1.	Diagramas de Caso de Uso	47
2.4.1.2.	Diagramas de Atividade	49
2.4.1.3.	Diagramas de Classes	49
2.4.2.	Programação orientada a objetos	50
2.4.3.	Testes de <i>software</i>	50
2.4.4.	Banco de dados	51

3.	Materiais e Método	52
3.1.	Sistema embarcado	52
3.2.	Comunicação sem fio	67
3.3.	<i>Software</i>	69
3.4.	Cuidados para o experimento	72
4.	Resultados	73
4.1.	<i>Software</i>	73
4.2.	Sistema embarcado	76
4.2.1.	Primeira Fase	76
4.2.2.	Fase Final	77
4.3.	Análise química do esgoto	85
5.	Conclusões e Trabalhos Futuros	86
	Referências Bibliográficas	88
	Apêndice A – Diagramas de Atividades	90
	Apêndice B– Diagramas de Classes	94
	Apêndice C– Diagrama elétrico da fonte de alimentação	95
	Apêndice D – Diagrama elétrico da placa de comunicação com XBee.	97
	Apêndice E – Diagrama elétrico da placa central do sistema embarcado.	98
	Apêndice F – Diagrama elétrico da placa de acionamento dos relés.	99
	Apêndice G – Diagrama elétrico da placa de condicionamento dos sinais dos sensores de OD e temperatura.	100

Estudos estatísticos sobre saneamento no Brasil foram iniciados em meados de 1970 pelo Instituto Brasileiro de Geografia e Estatística - IBGE, visando descrever a qualidade, eficiência e a relação estrutural dos sistemas de saneamento nas últimas décadas. Pesquisas atuais mostram que aproximadamente 30% dos municípios brasileiros não possuem rede coletora de esgoto. Neste contexto, é ressaltado principalmente a grave situação dos estados da região Nordeste [1].

A situação desses municípios é agravada pelo crescimento populacional e pelo uso desregrado dos recursos hídricos, aumentando consideravelmente o descarte de águas residuárias residenciais, e contribuindo para baixa eficiência dos sistemas de tratamento de esgoto existentes. Tal descarte sem tratamento ou tratamento ineficiente em meios hídricos contribui para o processo de eutrofização.

A eutrofização é o fenômeno causado pelo excesso de nutrientes como: nitrogênio e fósforo, no meio líquido. Este processo estimula a proliferação de algas, que deixa as águas com odor, baixa limpidez e baixa oxigenação. Para inibir este processo foram desenvolvidas técnicas, tais como: lagoas de estabilização sem aeração; sistemas anaeróbios simplificados; lagoas anaeróbias; ar difuso; e lodos ativados [2], [3].

O método de lodos ativados é caracterizado por um sistema mecanizado e aeróbico, em que as bactérias aumentam sua massa alimentando-se do material orgânico do esgoto. Após seu crescimento ocorre o processo de sedimentação, separando o efluente (água tratada) do lodo (bactérias). Esta técnica é amplamente utilizada em cidades, em nível secundário, com alta eficiência [2].

O escopo deste projeto consiste em uma variante da técnica de tratamento de lodos ativados, denominado Regime de Bateladas Sequenciais - RBS. Caracteriza-se no processo de tratamento de esgoto flexível que visa intensificar e acelerar o processo biológico natural das bactérias por meio da inserção de oxigênio artificialmente no reator, removendo material orgânico, nitrogênio e fósforo das águas residuárias. Este sistema é distribuído em fases de tratamento, como: alimentação; hidrólise; aeração; sedimentação; descarte do efluente; e descarte do lodo em excesso. Ocorrendo de forma sequencial e cíclica em cada reator.

A automação destes reatores contribuem para maximizar a eficiência do tratamento dos sistemas RBS, melhorando a qualidade da operação e reduzindo o consumo de energia. A utilização de dispositivos como controladores, atuadores, computadores, sensores e módulos de transmissão e recepção de dados via radiofrequência - RF contribuem para a otimização desse tratamento [4], [5].

A tecnologia de transmissão e recepção de dados via RF é usualmente utilizada em diversas áreas, tais como: monitoramento, sensoriamento, automação e controle, possibilitando aos usuários visualizar dados e controlar sistemas sem a necessidade da utilização de cabos. Dentre as tecnologias existentes, o padrão ZigBee tem se destacado pelo seu baixo custo de implantação e baixo consumo de energia.

A necessidade de métodos alternativos e eficientes em sistemas de tratamento de esgoto, com a finalidade de minimizar os desperdícios e eliminar as perdas dos sistemas estimulou a vasta quantidade de pesquisas na área, a saber: controle de reatores de bateladas sequenciais para tratamento de efluentes industriais [6]; condição mínima necessária de pH para reatores de bateladas sequenciais anaeróbicos com biomassa granular [7]; e um estudo piloto para remoção de nitrogênio biológico em reatores de bateladas sequenciais com alternância anóxica-aeróbia [8].

Esta dissertação esta organizada em cinco capítulos. No capítulo 1 é apresentado uma breve contextualização do problema, a descrição do modelo de regime de bateladas sequenciais, e os objetivos principais deste trabalho. No capítulo 2 é exposto uma revisão bibliográfica dos principais componentes que fundamentam esta pesquisa.

No capítulo 3 são apresentados os procedimentos metodológicos adotados para a realização deste projeto. E no capítulo 4 são apresentados os resultados da pesquisa. O capítulo 5 é reservado para a conclusão do trabalho.

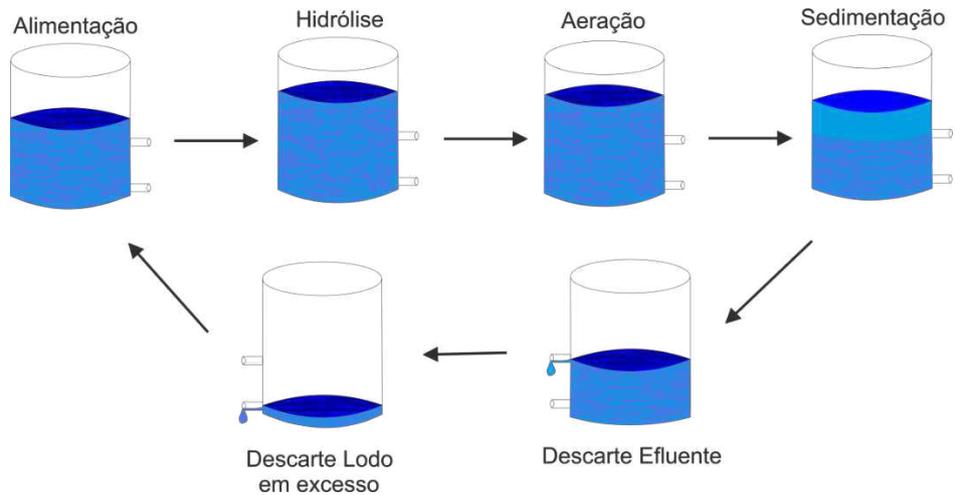
## 1.1. Regime de bateladas sequencias – RBS

Segundo Arima [9], os sistemas de RBS para tratamento em águas residuárias iniciou em meados de 1914. Porém, foi no decênio de 1970 que houve a aceitação e desenvolvimento de aplicações para o método.

O RBS é o processo de tratamento de esgoto com fases pré-definidas e sequenciais, em que todo procedimento ocorre em um único reator de forma cíclica. As fases são definidas a cada ciclo de tratamento, como [2], [9]:

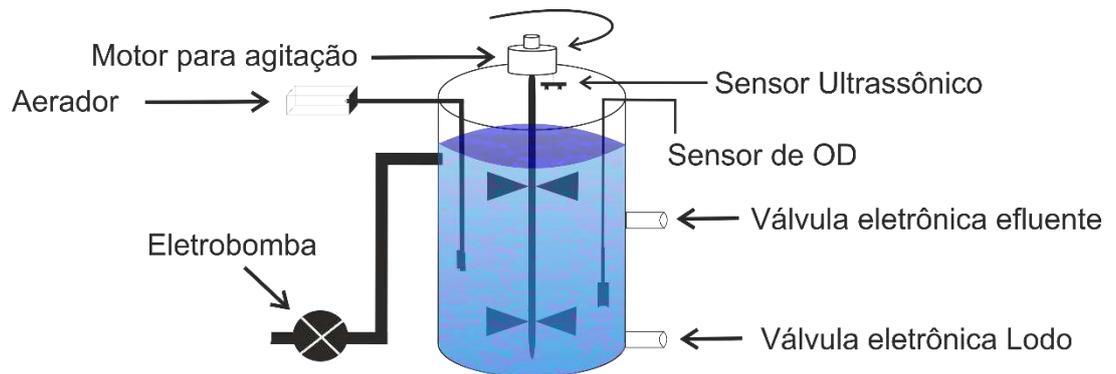
- Alimentação → fase específica para carregar uma batelada de afluente, esgoto, ao reator do sistema;
- Hidrólise → fase que mistura o lodo com o esgoto sem a inserção de oxigênio. Nesta fase as bactérias atuam na remoção biológica do esgoto;
- Aeração → fase destinada à remoção de nutrientes indesejáveis como: matéria orgânica, fosfato e amônia. Este processo é acelerado pela inserção de oxigênio no reator.
- Sedimentação → fase reservada para separar o lodo do efluente clarificado. A separação ocorre quando o regime no reator entra em repouso, por um tempo, permitindo que o lodo, mais pesado, decante;
- Descarte do efluente → fase destinada para retirar o efluente tratado do reator;
- Descarte do lodo em excesso → nesta fase é descartado uma pequena porção do lodo para manter a massa constante do lodo.

Na Figura 1 é apresentado o ciclo de fases do regime de bateladas no processo de tratamento pelo sistema RBS.



**Figura 1** Ciclo de tratamento pelo sistema em RBS.

Para dinamizar o processo de tratamento pelo método RBS utilizam-se sensores e atuadores que otimizam o tempo das fases do processo. Na Figura 2 é apresentada a descrição dos sensores e atuadores utilizados.



**Figura 2** Reator do RBS com os sensores e atuadores.

- Eletrobomba → Utilizada para encher o reator de afluente;
- Sensor Ultrassônico → Utilizado para verificar o nível do afluente no reator;
- Sensor de Oxigênio Dissolvido (OD) → Utilizado para verificar o nível de oxigênio dissolvido do fluido, indicando a ação e inibição das bactérias;
- Aerador → Utilizado para inserir oxigênio no fluido;
- Válvulas → Utilizadas para descartar o efluente e o lodo em excesso.

## **1.2. Objetivos**

Nesta seção são descritos os objetivos gerais que é o grande propósito desta pesquisa, bem como os objetivos específicos que são fundamentais.

### **1.2.1. Objetivos Gerais**

O objetivo desta pesquisa é propor e validar uma metodologia de automatizar e monitorar o processo de tratamento de águas residuárias residenciais pelo método de RBS, em escala piloto. Os dados obtidos pelo sistema embarcado nos reatores serão enviados via RF para um computador capaz de armazená-los e disponibilizá-los para o usuário.

### **1.2.2. Objetivos Específicos**

Os objetivos específicos são:

- Implantar um protótipo para a operação do RBS na unidade de tratamento anaeróbio da UEPB;
- Operar e monitorar automaticamente o protótipo utilizando os sensores e atuadores acoplados ao microcontrolador;
- Desenvolver o *software* para o microcontrolador capaz de gerenciar as fases do sistema de RBS, estipulando os seus períodos;
- Desenvolver um aplicativo computacional na linguagem de programação Java para capturar os dados obtidos dos reatores;
- Configurar os módulos XBee para criar um meio de comunicação entre aplicativo computacional e os reatores;
- Projetar um banco de dados para armazenar os dados dos reatores no aplicativo computacional.

## 2. Fundamentação teórica

Nesta seção é descrito a fundamentação teórica que acrescenta conceitos primordiais para o desenvolvimento e implementação desta pesquisa.

### 2.1. Microcontroladores

Em meados de 1969 os cientistas Bob Noyce e Gordon Moore iniciaram suas pesquisas para o desenvolvimento e fabricação de microchips. Porém, no início do decênio de 1970 foram construídos, de fato, os primeiros microprocessadores, Intel 4004, utilizados inicialmente nas calculadoras *Busicons* [10].

Com a evolução da microeletrônica, o nível de integração dos circuitos permitiu acoplar centenas de milhares de transistores em um único *chip*. Esta integração contribuiu para o agrupamento do microprocessador e periféricos no mesmo *chip*, surgindo assim os microcomputadores que posteriormente foram chamados de microcontroladores.

O microcontrolador é um sistema computacional integrado que possui inerentemente memórias internas (RAM e ROM), unidade central de processamento, do inglês CPU, e periféricos de entrada e saída, tais como: conversores analógico-digital (A/D) e conversores digital-analógico (D/A).

Atualmente, o microcontrolador é um dispositivo presente em diversos equipamentos eletrônicos que auxiliam, de alguma maneira, a vida de seus usuários. Hoje é comum ser acordado por um despertador eletrônico, esquentar as refeições no forno de micro-ondas, acionar portões por controle remoto, programar a TV para desligar em uma hora específica e locomover-se para o trabalho no carro com injeção eletrônica. Na Figura 3 são apresentados alguns exemplos de equipamentos que possuem algum tipo de microcontrolador.

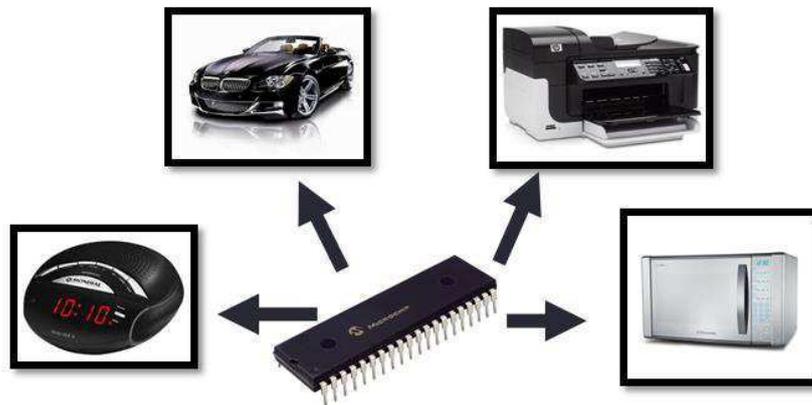


Figura 3 Equipamentos que utilizam microcontrolador.

### 2.1.1. Arquitetura do Microcontrolador

A arquitetura do microcontrolador é subdividida em dois tipos: arquitetura de Von Neumann e arquitetura de Harvard.

A arquitetura de Von Neumann é a tecnologia que utiliza o mesmo barramento para conectar as memórias de dados e de programa a CPU. Isto é, utiliza-se um único barramento para buscar instruções na memória de programa, como escrever e ler na memória de dados. Esta arquitetura gera centenas de instruções complexas para o processador, por isso, são denominados de CISC (*Complex Instruction Set Computer*). Este tipo de processamento necessita de vários ciclos de *clock* para executar uma única instrução. Na Figura 4 é apresentada a arquitetura de *Von Neumann*.

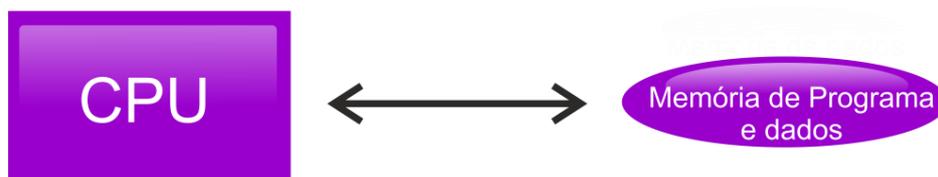
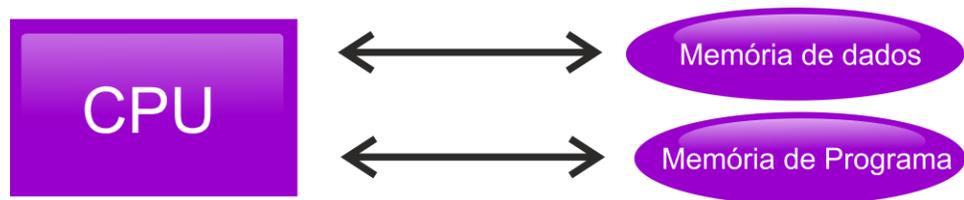


Figura 4 Arquitetura de Von Neumann.

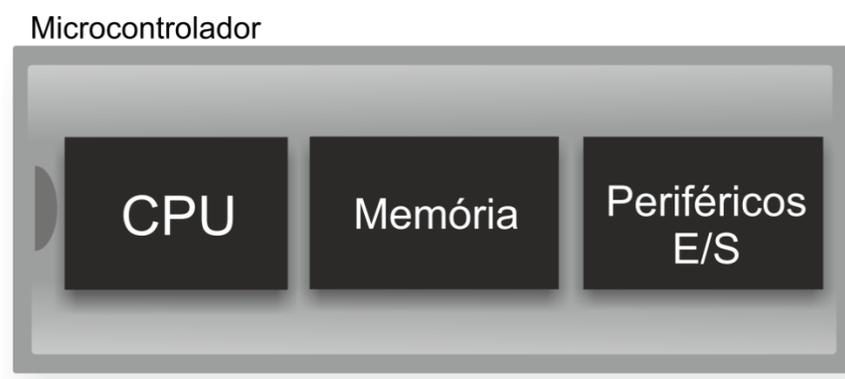
A arquitetura de Harvard diferencia-se da arquitetura de Von Neumann na utilização de dois barramentos que conectam as memórias de dados e de programa a CPU, separadamente. Os microcontroladores que utilizam esta

arquitetura produzem um conjunto simples de instruções, por isso, são denominados de RISC (*Reduced Instruction Set Computer*). Esta arquitetura permite executar uma instrução por ciclo de *clock*, tornando a execução mais rápida do que a arquitetura CISC. Na Figura 5 é apresentada a estrutura da arquitetura de Harvard [10].



**Figura 5** Arquitetura de Harvard.

A arquitetura simplificada do microcontrolador pode ser dividida em três módulos principais, que são: CPU, responsável por realizar todas as operações aritméticas e lógicas; memórias internas, responsáveis no armazenamento dos dados e na codificação do programa; e periféricos de entrada e saída (E/S), responsáveis pela comunicação entre componentes externos e internos [10], [11]. Na Figura 6 são apresentadas as principais estruturas que compõem os microcontroladores.



**Figura 6** Estruturas principais dos microcontroladores.

## 2.1.2. Características dos Microcontroladores

Os microcontroladores apresentam em sua arquitetura estendida outros componentes importantes, tais como: conversor analógico-digital, temporizadores, memória de programa e de dados, comunicação serial, CCP - Capture/Compare/PWM, relógio e interrupções. Na Figura 7 são apresentados os subsistemas do microcontrolador.

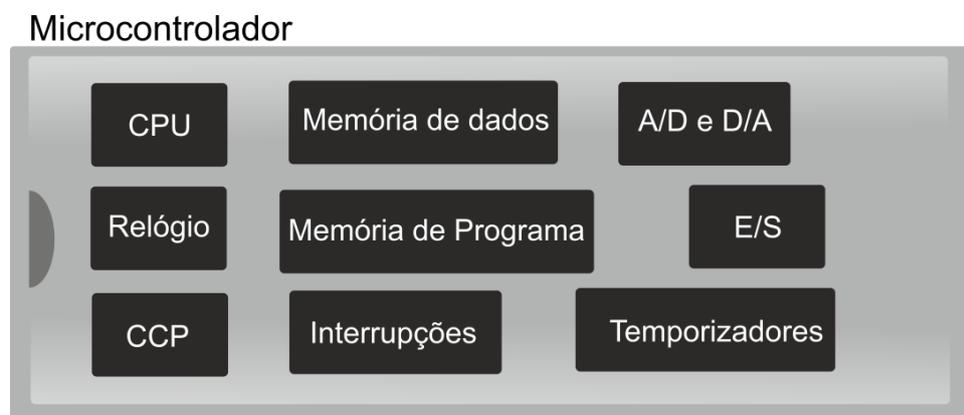


Figura 7 Componentes do microcontrolador.

- **CPU**

A CPU é um componente do microcontrolador que processa e coordena todas as informações internas, onde as instruções são decodificadas, executadas e enviadas para registradores. Estas instruções são distribuídas em operações lógicas e aritméticas [11].

- **CCP**

O CCP é um módulo utilizado para controlar e medir tempo de diversos eventos, este componente é subdividido em três: módulo de captura, utilizado para calcular o tempo de duração de um respectivo evento; módulo de comparação, utilizado para comparar os valores do temporizador (*Timer 1*) com os valores do

registrador (CCP1); e o PWM (*Pulse-Width Modulation*), que permite gerar sinais com frequência e ciclo de trabalho (*duty-cycle*) variáveis.

- **Relógio**

O Relógio (*clock*) é um sinal utilizado para sincronizar as operações internas do microcontrolador. Os dispositivos utilizados para gerar o sinal do relógio são geralmente: capacitores, ressonador ou resistor-capacitor [10]. Segundo a empresa Microchip [12], é recomendado utilizar para os microcontroladores PIC dois capacitores e um cristal. Na Figura 8 é apresentado o modelo estrutural sugerido pela Microchip para utilização de seus microcontroladores.

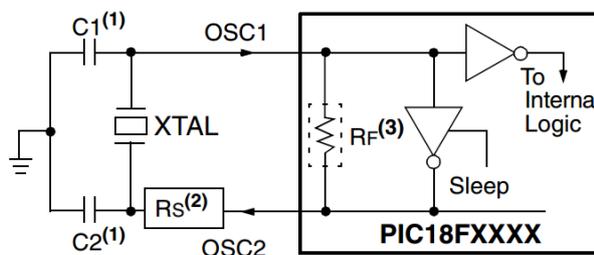


Figura 8 Estrutura para utilização do relógio[12].

- **Memórias**

As memórias, nos microcontroladores, são classificadas em dois tipos: memória de programa e memória de dados. A memória de programa é caracterizada como não volátil, pois os dados de controle armazenados não são perdidos quando a fonte de energia é desligada. A memória de dados, geralmente, é caracterizada como volátil, pois os dados são armazenados temporariamente durante a execução do programa, enquanto a fonte de energia estiver ligada.

Dentre estas categorias, existem atualmente cinco tipos de memórias: RAM (*Random Access Memory*), memória volátil que armazena dados durante a execução do programa; ROM (*Ready Only Memory*), memória não volátil que armazena os dados uma única vez; EPROM (*Erasable Programmable Ready Only Memory*), memória programável que pode ser apagada com luz ultravioleta; EEPROM (*Electrically Erasable Programmable Only Memory*), memória que pode

ser programada eletronicamente; e flash EEPROM é uma variação mais veloz da memória EEPROM [10].

A utilização da memória de programa, baseada na EEPROM, apresenta a vantagem de se reutilizar o mesmo microcontrolador em diversas aplicações, tais como: controle de produtos, controle de áreas irrigadas; automação residencial; e predial.

- **Interrupções**

Interrupções podem ser causadas por eventos associados internamente ou externamente ao microcontrolador. Quando ocorre uma interrupção, a execução da rotina principal é desviada para a rotina secundária de tratamento da interrupção, que após o seu término volta à execução da rotina principal. Nos microcontroladores existem diversos tipos de interrupções: dos temporizadores; dos osciladores e externas.

- **Temporizadores**

Os temporizadores (*Timers*) são registradores utilizados para calcular tempo a partir de pulsos de *clock*. Nos microcontroladores os *timers* são, geralmente, registradores de 8 bits ou 16 bits, dependendo do modelo.

- **E/S**

Os periféricos de entrada e saída (E/S) permitem a comunicação de dispositivos externos a módulos internos dos microcontroladores. Estes módulos externos podem ser botões, display, diodos e outros.

- **Conversor A/D**

Os conversores A/D são módulos essenciais para os microcontroladores, pois convertem sinais analógicos em sinais digitais. Os microcontroladores apresentam variações na resolução de bits e no tipo do conversor utilizado.

### 2.1.3. Microcontrolador PIC 18F4550

Os microcontroladores PIC (*Peripheral Interface Controller*) são desenvolvidos pela empresa americana Microchip Technology baseados na arquitetura de Harvard.

O PIC da família 18F apresenta alto desempenho computacional, alta resistência e baixo custo. Estes microcontroladores possuem recursos que contribuem para redução do consumo de energia, são eles: módulos de execução alternativos capazes de reduzir o consumo de energia em até 90%; e módulos de energia gerenciados que permitem que os programadores introduzam métodos econômicos no *software* [12]. Na Figura 9 é apresentada a estrutura dos pinos do PIC18F4550.

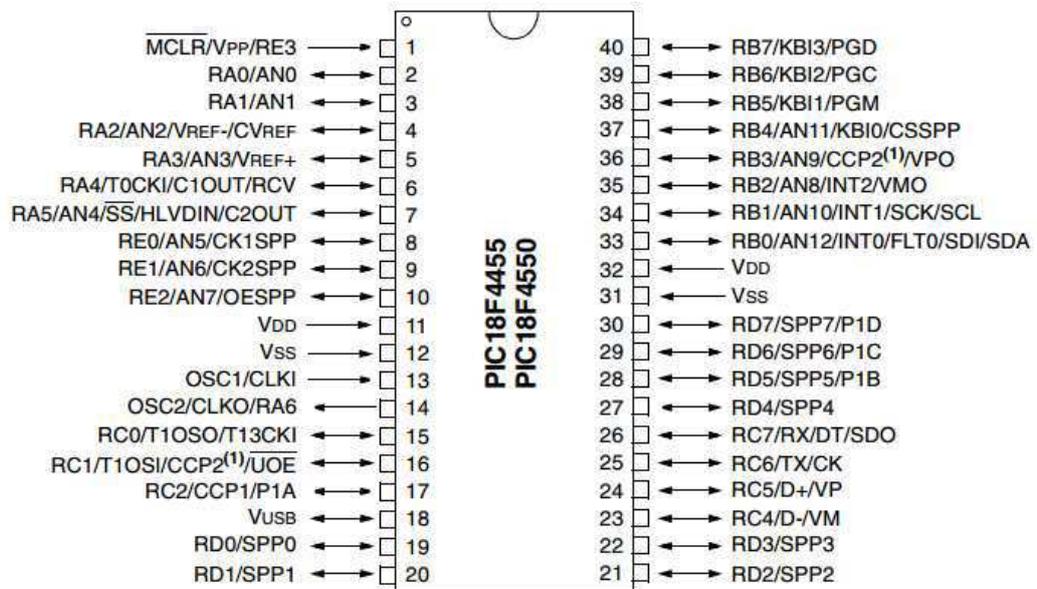


Figura 9 Estrutura dos pinos do microcontrolador PIC18F4550[12].

O PIC 18F4550 dispõe de subsistemas como: memória flash e EEPROM que suporta 100.000 e 1.000.000 de ciclos de gravação/exclusão respectivamente; módulos CCP; comunicação serial-USART; interface USB; e outros apresentados na Quadro 1 [12].

Quadro 1 Características do PIC18f4550 [12].

Características do PIC 18F4550	
Frequência Máxima de Operação	48 MHz
Memória de Programa	32768 Bytes
Memória de dados	2048 Bytes
Memória EEPROM	256 Bytes
Fontes de Interrupção	20
Portas de E/S	A, B, C, D, E
Timer	4
Conversor A/D 10-Bit	13 entradas

O conversor A/D do PIC 18F utiliza a técnica de aproximações sucessivas para converter sinais analógico em digitais. Na Figura 10 é apresentada a estrutura em blocos da técnica de aproximação sucessiva.

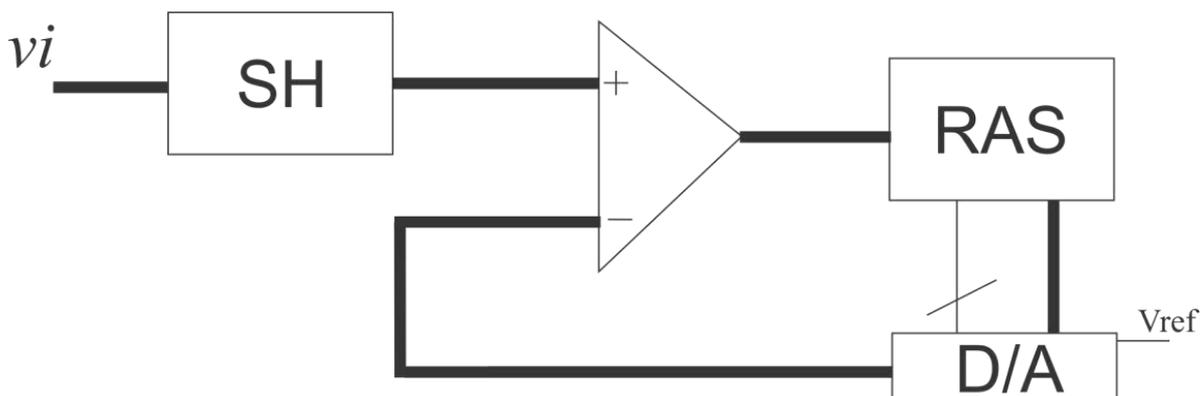


Figura 10 Estrutura em blocos da aproximação sucessiva.

A tensão de entrada  $v_i$  passa por um circuito de retenção denominado Sample/Holder - SH, o qual garante que não ocorrerá variação na entrada do comparador enquanto estiver convertendo o sinal. Esta tensão  $v_i$  é inserida no comparador junto com o valor do registrador de aproximações sucessivas - RAS, convertida em tensão pelo conversor D/A.

O RAS inicia o primeiro ciclo com o bit mais significativo em nível alto (1). Se a saída do comparador indicar que  $v_i$  ainda é maior que o valor do RAS, convertida em tensão, então este bit permanece em nível alto (1), caso contrário o bit é colocado em nível baixo (0). Em seguida o RAS atribui a seu próximo bit mais significativo nível alto e repete o mesmo ciclo anterior. Após sucessivos ciclos com os bits subsequentes, o RAS terá o valor binário da tensão inserida na entrada.

O tempo de conversão por bit - TAD aceitável pelo PIC 18FXXXX é aproximadamente de 0,8  $\mu$ s para um cristal de 20MHz. Então, para que a conversão de 10 bits seja possível, é necessário um período de 11 TAD [12].

## 2.2. Sensores

Sensores são dispositivos sensíveis a estímulos externos como temperatura, pressão, peso, velocidade, deslocamento, processos químicos ou eletroquímicos, capazes de produzir uma resposta mensurável em sua saída, permitindo que sistemas de controle possam condicionar e processar estes sinais.

Conforme Pereira [13], existem algumas características importantes para a escolha de um tipo de sensor, que são:

- Resolução  $\rightarrow$  menor variação possível na entrada para gerar uma variação na saída;
- Sensibilidade  $\rightarrow$  razão da variação de saída pela variação da entrada;
- Precisão  $\rightarrow$  capacidade de apresentar a mesma leitura para várias medições em condições iguais;
- Tempo de resposta  $\rightarrow$  é o tempo necessário para o sensor apresentar uma resposta estável.

Por outro lado, os transdutores são definidos como dispositivos capazes de converter um tipo de energia em outro. Neste contexto, utilizou-se os transdutores

ultrassônicos (pois convertem sinais do domínio elétrico para o mecânico e vice e versa), sensor de OD e sensor de temperatura.

### 2.2.1. Transdutor Ultrassônico

Ondas acústicas são vibrações que se propagam em um determinado meio. Estas são classificadas em três grupos: sônicas(audíveis); infrassônicas; e ultrassônicas. Na Figura 11 são apresentados as faixas de frequência de cada grupo.

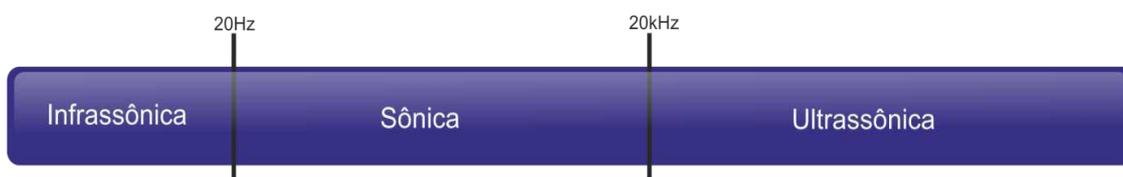


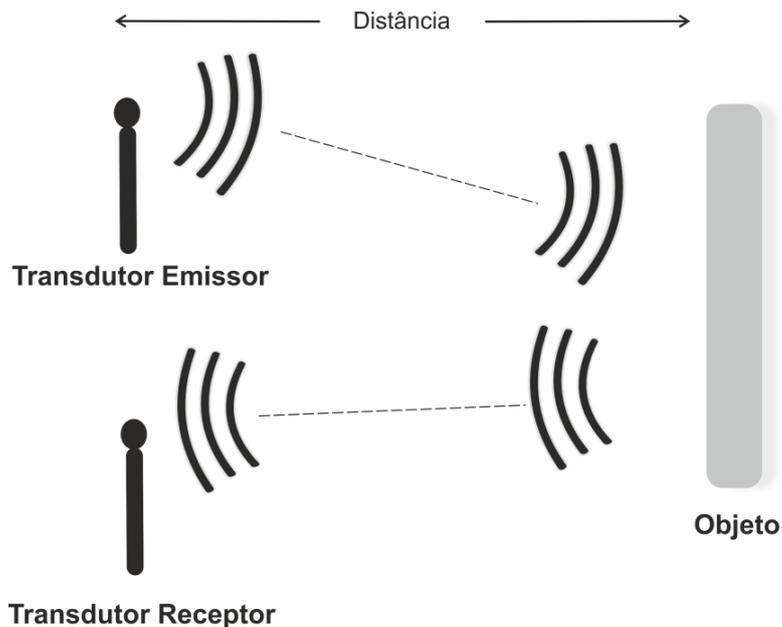
Figura 11 Faixa de frequência das ondas acústicas.

As ondas sonoras ultrassônicas são introduzidas ao meio por dispositivos emissores que produzem frequências superiores a 20 kHz. Os métodos capazes de realizar este processo são os efeitos piezoelétricos e eletromagnéticos.

O efeito piezoelétrico apresenta dois princípios: o primeiro consiste na aplicação de uma diferença de potencial na superfície do material, capaz de expandir ou contrair o material; e o segundo é o efeito piezoelétrico reverso, cujo princípio consiste na aplicação de pressão no material capaz de gerar uma diferença de potencial elétrico nas superfícies. Portanto, estes princípios contribuem para a utilização dos materiais piezoelétricos em transdutores ultrassônicos. Transformando energia mecânica em energia elétrica e energia elétrica em energia mecânica [14].

Os transdutores ultrassônicos atuam como transmissores ou receptores, possibilitando sua utilização em aplicações que aferem a distância de objetos. Para a obtenção deste fim, geralmente, são utilizados dois transdutores, um para emitir o sinal e outro para captá-lo. Isto é, um sinal de tensão externo estimula o transdutor emissor a transmitir ondas que se propagam com velocidade constante, estas ondas ao encontrar algum objeto ecoam de volta até o receptor captá-las. A

utilização de um circuito de condicionamento permite tratar estes sinais podendo ser codificados por um microcontrolador capaz de calcular o tempo decorrido entre o envio e a recepção. Após saber o tempo decorrido é possível calcular a distância que o objeto está do emissor, pois a velocidade do som é conhecida [13], [14]. Na Figura 12 é apresentada a estrutura básica do funcionamento dos transdutores ultrassônicos.



**Figura 12 Transdutores emissor e receptor.**

De acordo com Pereira [13], o circuito de condicionamento para os transdutores é, geralmente, caracterizado por comparadores, circuito detector e um circuito de saída. O comparador e o circuito detector são utilizados para comparar as frequências dos transdutores e determinar de forma representativa, em nível de tensão, o tempo decorrido entre a transmissão e recepção da onda. O circuito de saída é utilizado para condicionar o sinal, fornecendo parâmetros que possam ser controlados externamente. Na Figura 13 é apresentado a estrutura básica de condicionamento dos transdutores ultrassônicos.

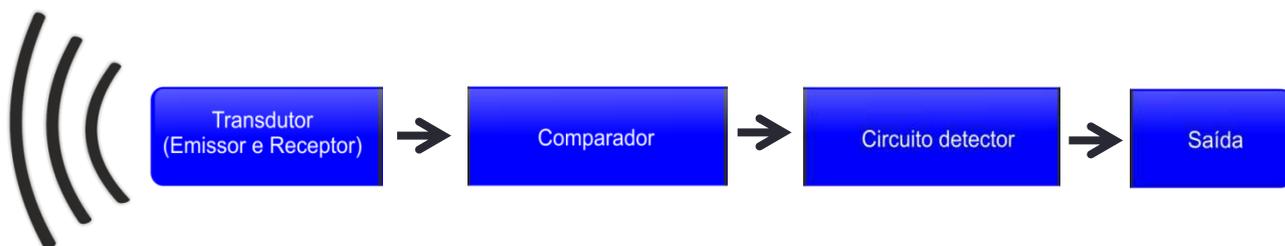


Figura 13 Estrutura básica do transdutor ultrassônico (adaptado [13]).

## 2.2.2. Sensor de Oxigênio Dissolvido

A quantificação de oxigênio dissolvido (OD) em meios líquidos é um parâmetro importante para aplicações em atividades médicas, biológicas e ambientais. Atualmente, o monitoramento do OD já é utilizado no meio industrial: na agricultura, na produção de alimentos e em estações de tratamento de águas residuárias.

O OD é um importante indicador da qualidade da água no tratamento de esgoto, pois seres vivos, especialmente os peixes, não sobrevivem a uma concentração abaixo de 4 mg/l. As observações do OD são úteis para o controle dos processos de águas residuárias [15].

Sensores de OD são construídos, geralmente, por dois tipos de tecnologias: o método óptico, conhecido como método da luminescência e o eletroquímico. Ambos são subdivididos em dois subgrupos: tempo de vida e intensidade, polarográfico e galvânicos respectivamente. Na Figura 14 são apresentadas as subdivisões dos tipos de sensores de OD.

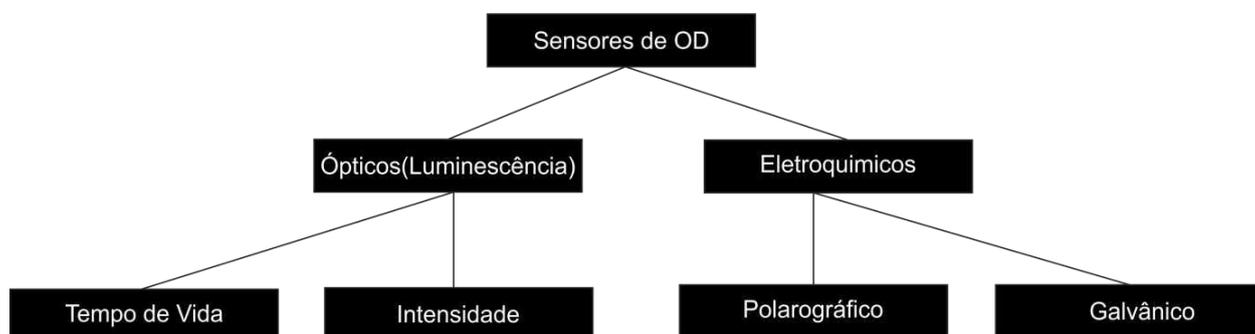
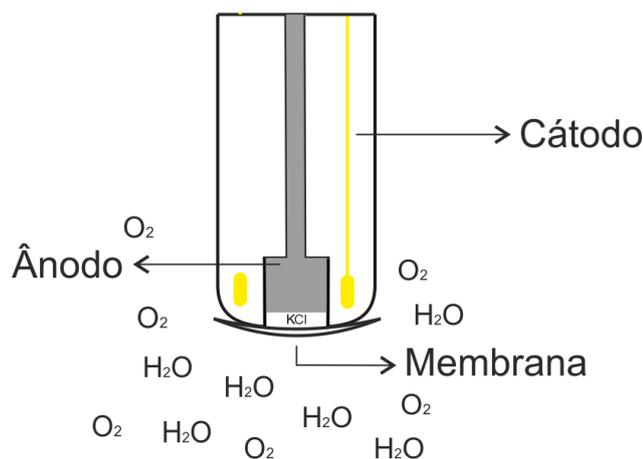


Figura 14 Ramificação dos tipos de sensores OD.

Os sensores baseados no método óptico detectam a quantidade de oxigênio dissolvido através da intensidade ou tempo de vida da luminescência, associada a corantes químicos cuidadosamente selecionados. Os valores da intensidade ou tempo de vida são inversamente proporcional à quantidade de OD do fluido. Esses sensores são constituídos, geralmente, por um indicador luminescente e um sistema óptico, que agrega fonte de excitação, filtros ópticos, guias de onda, fotodetectores e processamento do sinal.

Os sensores que utilizam a tecnologia eletroquímica são compostos por um cátodo e um ânodo que estão isolados em solução eletrolítica por uma membrana permeável. As moléculas de oxigênio são dissolvidas a uma velocidade proporcional à diferença de pressão entre a parte interna da membrana e a solução a ser medida [16]. Na Figura 15 é apresentada a estrutura do sensor de oxigênio dissolvido eletroquímico.



**Figura 15 Composição do sensor OD eletroquímico(adaptado [16]).**

A quantidade da concentração de oxigênio dissolvido é adquirida mediante a aplicação de uma tensão de polarização no cátodo, que difunde as moléculas de oxigênio através da membrana, resultando assim, em um fluxo de corrente no ânodo. Este fluxo de corrente é proporcional ao OD consumido [16]. Na Figura 16 é apresentada a forma da polarização do sensor OD eletroquímico.

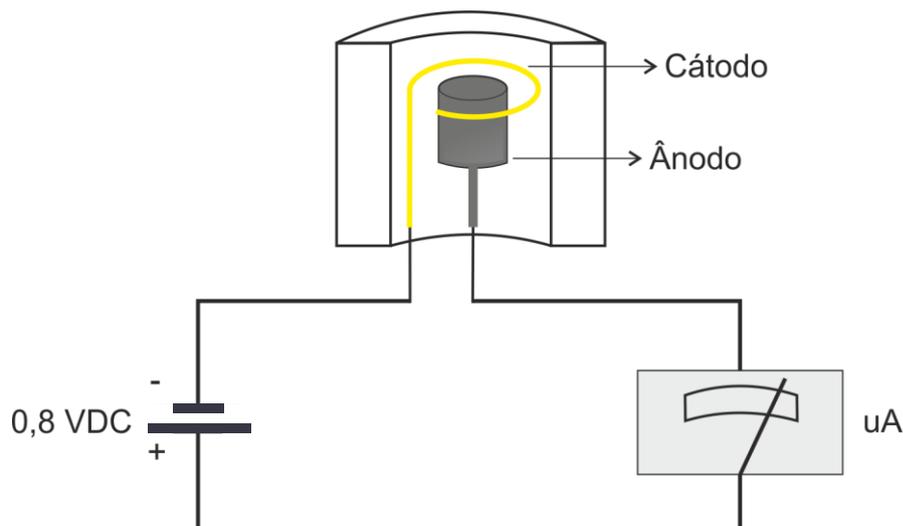


Figura 16 Polarização do sensor OD eletroquímico(adaptado [16]).

### 2.2.3. Sensor de temperatura

A temperatura é um fator importante para medição em tratamento de águas residuárias, pois sua variação altera diretamente o metabolismo das bactérias. O crescimento bacteriano é causado pelas reações catalisadas por enzimas específicas, sendo que a variação da temperatura afeta sua produção, aumentando ou diminuindo. Em geral, a temperatura considerada ótima para o crescimento das bactérias encontra-se na faixa de 25 a 36 °C [17].

Atualmente, existe uma diversidade de tipos de sensores de temperatura, tais como: termorresistivos; termopares; diodos; e transistores. Porém, para o escopo desta pesquisa será estudado apenas os termorresistivos.

Sensores do tipo termorresistivos são caracterizados pela variação da resistência elétrica através de uma variação na temperatura, seja direta ou inversa. Comumente são encontrados em dois tipos, PTC (Coeficiente de Temperatura Positivo) e NTC (Coeficiente de Temperatura Negativo). A relação da resistência elétrica e temperatura para o sensor PTC é diretamente proporcional, já para o NTC é inversamente proporcional. Para o estudo nos aprofundaremos no termistor NTC.

Para representar essas relações quantitativamente, Steinhart-Hart desenvolveu a expressão 1.

$$\frac{1}{T} = a + b \ln R + c(\ln R)^3. \quad (1)$$

Para análise dos termistores NTC a expressão 1 pode ser descrita como apresentada na Expressão 2.

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \ln \frac{R}{R_0}. \quad (2)$$

Logo,

$$R = R_0 \cdot e^{B \cdot \left(\frac{1}{T} - \frac{1}{T_0}\right)}. \quad (3)$$

Segundo [18] e [19], a relação da resistência elétrica e a temperatura para os termistores NTC pode ser descrita como apresentada na expressão 4.

$$R = A \cdot e^{B \cdot \left(\frac{1}{T}\right)}. \quad (4)$$

Em que,  $T$  é a temperatura (K),  $R$  é a resistência ( $\Omega$ ), e  $A$  e  $B$  são constantes que variam para cada termistor.

### 2.3. Comunicação Sem Fio

A comunicação sem fio, atualmente, é uma das tecnologias em ascensão em dispositivos eletrônicos, possibilitando a transferência de dados, à distância, sem a utilização de fios e cabos. Devido a sua flexibilidade e confiabilidade este tipo de comunicação abrange uma vasta quantidade de aplicações, tais como: controle remoto; mouses; teclados; transmissão de vídeo; áudio; acionamento de lâmpadas; sistemas de monitoramento e controle.

A transmissão de dados via radiofrequência utiliza o ar (ondas de rádio) como meio de propagação das informações. Esta propagação ocorre em uma faixa de frequência padrão em cada dispositivo.

Os protocolos de comunicação sem fio mais conhecidos são: Wi-Fi e Bluetooth, porém percebeu-se a necessidade de um padrão específico que trabalhasse de forma mais eficiente nas aplicações de monitoramento, sensoriamento, controle e automação. Esta necessidade contribuiu para o surgimento do padrão ZigBee.

### 2.3.1. Padrão ZigBee

O desenvolvimento do padrão ZigBee iniciou no ano de 2000, por um grupo denominado IEEE 802 WPAN (*Wireless Personal Area Network*). Este grupo começou a definir um novo protocolo de comunicação sem fio, IEEE 802.15.4 para as redes WPAN, visando dispositivos com baixo consumo de energia, baixo custo e maior segurança, tanto para ambientes residências como indústrias. Com a tecnologia em expansão a ZigBee™ Alliance disponibilizou sua primeira especificação em 2004, com base no controle de acesso (MAC) e na camada física (PHY) do protocolo IEEE 802.15.4 [20].

Os padrões sem fio conhecidos por Wi-Fi, Bluetooth e ZigBee apresentam diferenças em suas características, as quais são apresentadas na Quadro 2.

Quadro 2 Diferenças entre os padrões Wi-Fi, Bluetooth e ZigBee (adaptado [21]).

<b>Características</b>	<b>ZigBee</b>	<b>Bluetooth</b>	<b>Wi-Fi</b>
<b>Camada Física</b>	IEEE 802.15.4	IEEE 802.15.1	IEEE 802.11x
<b>Alcance</b>	10 m a 3 km	10 a 100m	100 m
<b>Taxa de transmissão</b>	250 kbps	1 Mbps	54 Mbps
<b>Frequências</b>	2,4GHZ; 868MHz; 902 a 928 MHz	2,4 GHz	2,4 a 5 GHz

<b>Nº de nós</b>	+65000	8	+100
<b>Modulação</b>	DSS	FHSS	OFDM, DSSS
<b>Consumo</b>	30 mA	65 a 170 mA	350 mA

Os dispositivos que utilizam o padrão ZigBee apresentam uma baixa taxa de transmissão de dados. No entanto, para aplicações que não necessitem de uma alta velocidade de transmissão, este padrão apresenta vantagens como: grande número de nós na rede; baixo consumo de energia; e alcance. Estas vantagens contribuem para aplicações em dispositivos remotos que utilizam baterias como fonte de alimentação.

O padrão ZigBee não foi desenvolvido com a finalidade de substituir alguma tecnologia existente, e sim, acrescentar melhorias na área de transmissão de dados entre dispositivos. Na Figura 17 é apresentada a relação entre taxa de dados e alcance dos protocolos de comunicação sem fio.

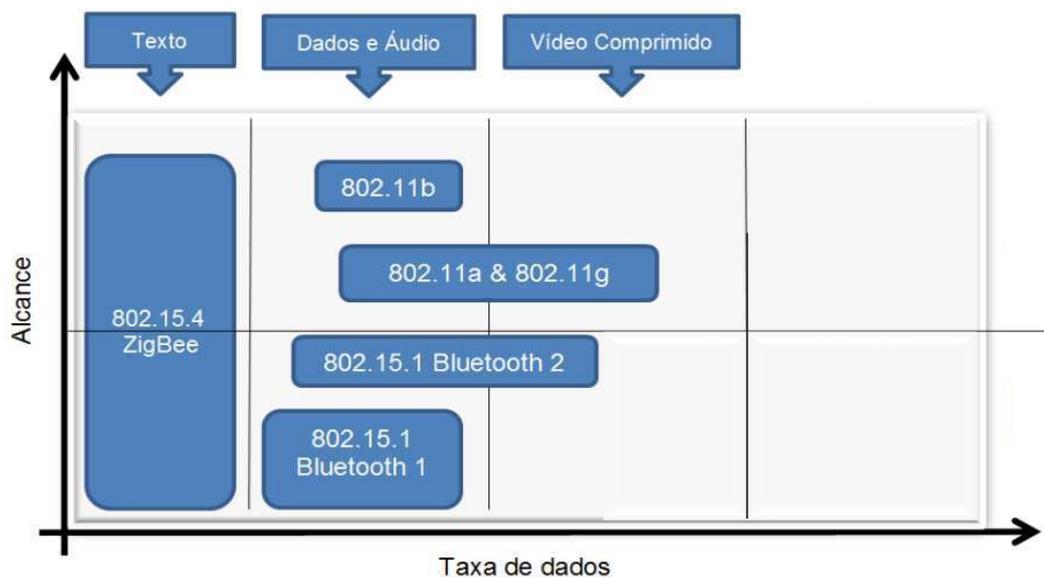


Figura 17 Relação alcance e taxa de dados dos padrões (adaptado [21]).

### 2.3.2. Redes ZigBee

As redes ZigBee são capazes de definir três tipos de dispositivos: roteador; coordenador e o dispositivo final (*end device*). Estes apresentam as seguintes características [19]:

- Coordenador → é responsável pelo gerenciamento da rede, organizando o endereçamento e o roteamento dos pacotes. Existe apenas um tipo deste dispositivo por rede.
- Roteador → é responsável pelo roteamento dos dados na rede, armazenando tabelas de endereçamento de dispositivos finais, de roteamento e de transmissão. Estes interligam os dispositivos finais ao coordenador.
- Dispositivo Final → é responsável por captar dados externos e enviá-los ao coordenador. Estes são unidos e gerenciados pelos roteadores ou coordenadores.

Estes dispositivos contribuem para a organização estrutural de três topologias de rede, que são: estrela (*Star*), malha (*Mesh*) e árvore (*Tree*).

#### 2.3.2.1. Topologia Estrela (*Star*)

Na topologia estrela, a rede é controlada por um único dispositivo coordenador capaz de iniciar e manter os dispositivos na rede, controlando e coordenando todos os outros dispositivos finais que são diretamente conectados ao dispositivo coordenador [22]. Na Figura 18 é apresentada a estrutura da topologia estrela.

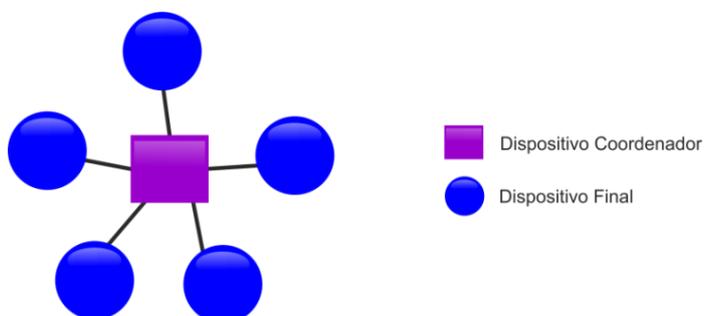


Figura 18 Estrutura da topologia estrela.

### 2.3.2.2. Topologia Árvore (*Tree*)

A topologia em árvore é estruturada semelhante à topologia estrela, porém com o acréscimo de dispositivos roteadores. Possui um único dispositivo coordenador capaz de iniciar e manter os dispositivos na rede, controlando e coordenando, os roteadores e dispositivos finais [22]. Na Figura 19 é apresentada a estrutura da topologia árvore.

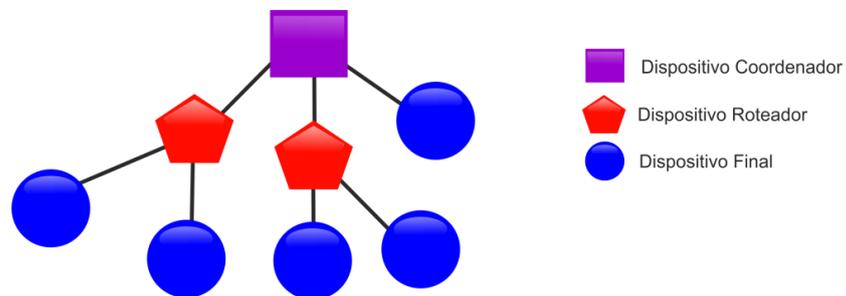


Figura 19 Estrutura da topologia malha.

### 2.3.2.3. Topologia Malha (*Mesh*)

A topologia malha é semelhante à topologia em árvore, porém permite que os nós roteadores se comuniquem com nós roteadores adjacentes. A rede é controlada por um único dispositivo coordenador capaz de iniciar e manter os dispositivos na rede, controlando e coordenando, os roteadores e dispositivos finais. Os roteadores são responsáveis por enviar dados e mensagem de controle através da rede utilizando as tabelas de roteamento [22]. Na Figura 20 é apresentada a estrutura da topologia malha.

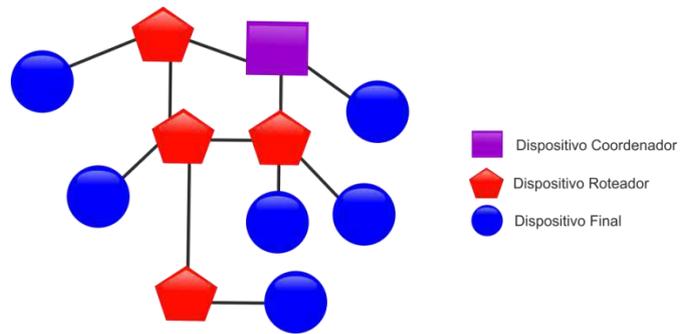


Figura 20 Estrutura da topologia malha.

### 2.3.3. Arquitetura do ZigBee.

O padrão ZigBee apresenta uma arquitetura dividida em camadas, designadas como: camada física (PHY - *Physical Layer*), camada MAC (*Media Access Control*), camada de rede (NWK - *Network Layer*), camada de aplicação (APL - *Application Layer*) e suas subcamadas: AF (*Application Framework*); APS (*Application Support Sub Layer*); e ZDO (*ZigBee Device Objects*), SSP (*Security Service Provider*) e a camada de aplicação do usuário. Na Figura 21 é apresentada a arquitetura do padrão ZigBee.

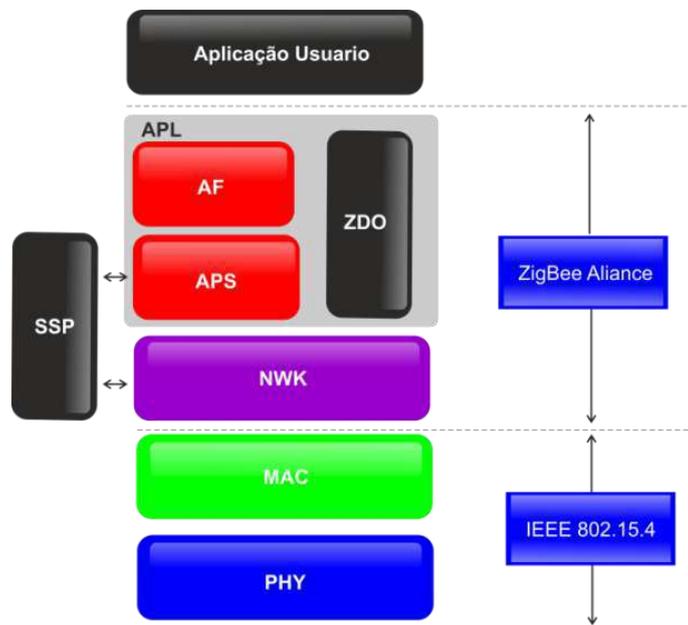


Figura 21 Arquitetura do ZigBee(adaptado de [19]).

As duas primeiras camadas, PHY e MAC, foram definidas pelo padrão IEEE 802.15.4, porém as camadas de rede (NWK), segurança (SSP), suporte e aplicação (APL) foram definidas pela ZigBee™ Alliance. A última camada é desenvolvida pelos fabricantes de cada dispositivo, conforme suas necessidades de aplicação.

A camada física (PHY) é responsável pela definição das operações do dispositivo: sensibilidade de recepção; potência de saída; detecção de canais; e especificação da taxa de transmissão. Esta camada suporta três bandas de frequências apresentadas na Quadro 3.

**Quadro 3** Bandas de frequência suportadas na camada física.

<b>Frequência</b>	<b>Canais Utilizados</b>	<b>Taxa de Transmissão</b>
<b>2,4 GHz</b>	16	250 kbps
<b>915 MHz</b>	10	40 kbps
<b>868 MHz</b>	1	20 kbps

A frequência de 2,4 GHz é mundialmente utilizada, principalmente no Brasil, sendo que a de 868 MHz é específica para os países europeus e a de 915 MHz pelos Estados Unidos da America (EUA) e Canadá [21].

A camada MAC é responsável no controle do canal de rádio usando o mecanismo de prevenção de colisão CSMA-CA (*Carrier Sense Multiple Acces - Collision Avoidance*), como também no gerenciamento dos *beacons*, que são pacotes de controle. Essa camada apresenta a configuração de dois dispositivos, como: dispositivo com função reduzida (*Reduced Function Devices - RFD*) e dispositivo com função completa (*Full Function Device - FFD*). Os RFDs são configurados como dispositivos finais e os FFDs podem ser configurados como coordenador, roteador e dispositivos finais.

A camada de rede (NWK) é responsável pela organização estrutural da rede, possibilitando a utilização das topologias estrela, malha e árvore, como também o roteamento de pacotes, transmissão em *broadcast*, e a validação da entrega dos pacotes.

A camada de segurança (SSP) é responsável para garantir que os dados criptografados transmitidos entre a camada de NWK e APS sejam preservados.

A camada de aplicação (APL) e suas subcamadas são responsáveis por definir tipos de dispositivos (coordenador, roteador ou dispositivo final), como também gerenciar as tabelas de armazenamento de nós e ligações da rede, filtrar pacotes e realizar a retransmissão de pacotes.

#### 2.3.4. Módulos XBee

Dentre os dispositivos que utilizam o padrão ZigBee, o modelo XBee, desenvolvido pela empresa *Digi International*, está recentemente em ascensão. A estrutura dos módulos é basicamente composta por um microcontrolador e um transceptor, em que as atuais arquiteturas permitem que os microcontroladores sejam programados.

Os módulos XBee são encontrados no mercado em diversas versões com diferentes características, a saber: modelo da antena; alcance; frequência de operação; potência de transmissão; sensibilidade do receptor; protocolo; e encapsulamento. Na Figura 22 são apresentados os principais modelos de dispositivos XBee.



Figura 22 Principais Modelos do XBee.

### 2.3.5. Modo de Operação do XBee

Os módulos XBee foram construídos para funcionarem de duas maneiras: no modo transparente e no modo API (*Application Programming Interface*).

#### 2.3.5.1. Modo Transparente

No modo de operação transparente, conhecido como modo AT, os dados recebidos da UART (*Universal Asynchronous Receiver/Transmitter*) através do pino de entrada DIN, entram na fila de transmissão e logo são enviados via RF para outro dispositivo. Os dados que são recebidos pela antena via RF são transmitidos para o pino DOUT do dispositivo [23].

#### 2.3.5.2. Modo API

O modo API é uma alternativa ao modo AT, pois é estruturado em quadros (*frames*), que definem como as operações, comandos e mensagens serão enviados pelo dispositivo. Este modo tem a vantagem de enviar *frames* para vários dispositivos, como também verificar o status do recebimento dos dados e identificar a origem de cada pacote [23].

O modo API pode ser habilitado de duas maneiras: operação API e operação API com caracteres de escape. A diferença é que a segunda operação permite escapar bytes de controle inseridos nos quadros que acarretariam em erros de pacote. Estes bytes são: 0x7E delimitador de início do pacote; 0x7D byte de escape; 0x11 byte reservado XON; 0x13 byte que indica que está desassociado do coordenador.

A estrutura do *frame*, destas operações, é subdividida em: o primeiro byte é utilizado para indicar o início do pacote, em que qualquer byte diferente de 0x7E é descartado; o segundo e terceiro byte informam o tamanho da *frame*; o quarto byte até o n-ésimo é reservado para os dados que estão sendo transmitidos; o último byte, n-ésimo + 1, é reservado para o valor do *checksum*. Este é utilizado para

informar a ocorrência de algum erro no pacote de envio. Na Figura 23 é apresentada a estrutura do *frame* da operação API [23].

### Estrutura do Pacote API



Figura 23 Estrutura do *frame* [23].

A verificação do *checksum* corresponde ao somatório dos bytes enviados menos o valor 0xFF em hexadecimal. Na expressão 5 é apresentada a fórmula para o cálculo do valor do *checksum*.

$$checksum = 0xFF - \sum_{i=4}^n byte[i]. \quad (5)$$

A transmissão de dados utilizando o modo API é baseado em três princípios, o envio dos dados para o destinatário, o recebimento dos dados pelo destinatário e o envio do status da transmissão para o remetente. Na Figura 24 é apresentado o esquemático do envio de dados no modo API.

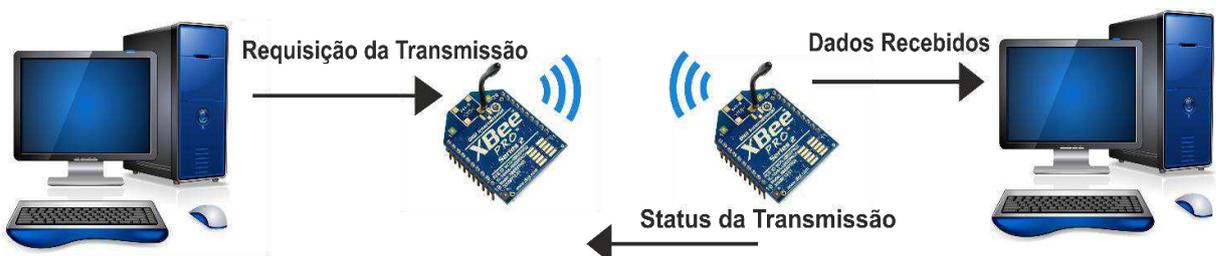


Figura 24 Esquemático do envio de dados pelo modo API.

O dispositivo XBee ao iniciar uma transmissão de dados envia uma sequência de bytes de forma estruturada como apresentado no Quadro 4. Neste exemplo, é enviado para o dispositivo com o endereço SH = 0x13A200 e SL = 0x40A0A668 a mensagem com o conteúdo “teste”.

Quadro 4 Exemplo de envio de dados pelo modo API (adaptado de [24]).

Pacote	Índice	Byte	Descrição
<b>Delimitador de início</b>	0	0x7E	Valor obrigatório para pacote API
<b>Tamanho do Pacote</b>	MSB 1	0x00	A soma dos bytes MSB e LSB representam a quantidade de bytes existentes entre o tamanho do pacote e o <i>checksum</i> . 0x13 = 19 bytes.
	LSB 2	0x13	
<b>Tipo</b>	3	0x10	0x10 – Transmissão de dados
<b>ID do pacote</b>	4	0x01	Byte de identificação do pacote
<b>Endereço de 64 bits de destino</b>	5	0x00	SH – 0x0013A200; SL – 0x40A0A668; Os endereços também suportados são: <b>0x0000000000000000</b> : Coordenador da rede; <b>0x000000000000FFFF</b> : Endereço de Broadcast;
	6	0x13	
	7	0xA2	
	8	0x00	
	9	0x40	
	10	0xA0	
	11	0xA6	
<b>Endereço de 16 bits de destino</b>	12	0x68	
	13	0xFF	O valor 0xFFFE é utilizado caso o destinatário seja desconhecido ou a transmissão é broadcast
14	0xFE		
<b>Broadcast</b>	15	0x00	

<b>Opções</b>	16	0x00	0x00 – Valor padrão; 0x01 – Desabilita resposta ACK;
<b>Mensagem</b>	17	0x74 – ‘t’	Dados que são transmitidos via RF
	18	0x65 – ‘e’	
	19	0x73 – ‘s’	
	20	0x74 – ‘t’	
	21	0x65 – ‘e’	
<b>Checksum</b>	22	0x29	0xFF – o somatório dos bytes entre o tamanho do pacote e o <i>checksum</i>

Este pacote de dados será recebido pelo dispositivo de destino de forma estruturada como apresentado no Quadro 5. Todo pacote recebido é identificado pelo endereço do remetente, neste exemplo: SH = 0x0013A200 e SL = 0x408CF6EF.

Quadro 5 Exemplo de recebimento de dados pelo modo API (adaptado de [24]).

<b>Pacote</b>	<b>Índice</b>	<b>Byte</b>	<b>Descrição</b>
<b>Delimitador de início</b>	0	0x7E	Valor obrigatório para pacote API
<b>Tamanho do Pacote</b>	MSB 1	0x00	A soma dos bytes MSB e LSB representam a quantidade de bytes existentes entre o tamanho do pacote e o <i>checksum</i> . 0x11 = 17 bytes.
	LSB 2	0x11	
<b>Tipo</b>	3	0x90	0x10 – Recebimento de dados
<b>Endereço de 64 bits do remetente</b>	4	0x00	SH – 0x0013A200; SL – 0x408CF6EF;
	5	0x13	
	6	0xA2	
	7	0x00	
	8	0x40	
	9	0x8C	
	10	0xF6	

	11	0xEF	
<b>Endereço de 16 bits de destino</b>	12	0xB4	O valor 0xFFFE é utilizado caso o destinatário seja desconhecido ou a transmissão é broadcast
	13	0x2D	
<b>Opções</b>	14	0x01	0x01 – Pacote reconhecido; 0x02 – Pacote broadcast; 0x20 – Pacote encriptado om APS; 0x40 – Pacote enviado de um dispositivo final;
<b>Mensagem recebida</b>	15	0x74 – ‘t’	Dados que são recebidos via RF
	16	0x65 – ‘e’	
	17	0x73 – ‘s’	
	18	0x74 – ‘t’	
	19	0x65 – ‘e’	
<b>Checksum</b>	20	0x29	0xFF – o somatórios dos bytes entre o tamanho do pacote e o <i>checksum</i>

Após o recebimento dos dados o dispositivo destinatário envia para o remetente um status da transmissão como apresentado no Quadro 6.

Quadro 6 Status da transmissão (adaptado de [24]).

<b>Pacote</b>	<b>Índice</b>	<b>Byte</b>	<b>Descrição</b>
<b>Delimitador de início</b>	0	0x7E	Valor obrigatório para pacote API
<b>Tamanho do Pacote</b>	MSB 1	0x00	A soma dos bytes MSB e LSB representam a quantidade de bytes existentes entre o tamanho do pacote e o <i>checksum</i> . 0x11 = 17 bytes.
	LSB 2	0x07	
<b>ID do pacote</b>	3	0x01	Byte de identificação do pacote
<b>Tipo</b>	4	0x8B	0x8B – Status da transmissão
<b>Endereço de 16 bits do remetente</b>	5	0x00	
	6	0x00	

<b>Tentativas de retransmissão</b>	7	0x01	
<b>Status de entrega</b>	8	0x00	0x00 = Sucesso 0x22 = Não registrado na rede 0x24 = Endereço não encontrado 0x25 = Roteador não encontrado 0x74 = O pacote é muito grande
<b>Status de descoberta</b>	9	0x01	0x00 = Não foi encontrado 0x01 = Endereço encontrado 0x02 = Roteador encontrado 0x40 = Expirou o tempo limite de procura
<b>Checksum</b>	20	0xEC	0xFF – o somatório dos bytes entre o tamanho do pacote e o <i>checksum</i>

## 2.4. Software

O *software* é caracterizado por um conjunto de sequências de instruções que são interpretadas por um computador com a finalidade de executar uma tarefa específica. O conceito de *software* abrange desde o código implementado, como a documentação e os dados de configuração do sistema.

Para a obtenção de um bom desenvolvimento do *software* foi utilizado alguns conceitos de engenharia de *software*. Este é o ramo da engenharia relacionada a aspectos de produção de *software*, desde o levantamento de requisitos até a manutenção do sistema.

Para a implementação do sistema é importante observar o conceito de qualidade de *software* que é a combinação de fatores que podem ser medidos diretamente ou indiretamente. Os fatores diretos são expressões como tempo de processamento e tempo de aparecimento de erros, os indiretos como usabilidade e confiabilidade do sistema.

### 2.4.1. Linguagem UML

A UML é uma linguagem de modelagem com a finalidade de especificar, construir e documentar o *software* independentemente do paradigma utilizado na codificação, desde sistemas computacionais corporativos até aplicativos [25], [26].

A linguagem UML é utilizada para modelar sistemas de forma diagramática com objetivo de facilitar a compreensão e construção do mesmo. Os diagramas são representações gráficas que contribuem para a visualização do sistema em diferentes perspectivas. Estes diagramas podem ser especificados como: diagramas de classes; de objeto; de caso de uso; de sequência; de colaboração; de atividades; de estados; de componentes e implantação.

Os diagramas construídos na linguagem UML facilitam a compreensão do funcionamento do *software* para as pessoas envolvidas no desenvolvimento, como também o entendimento para posteriores alterações caso necessário.

Dentre os diagramas citados serão apresentados apenas os de caso de uso, de atividade e de classes, pois estes fornecem uma visão geral do funcionamento do sistema computacional implementado.

#### 2.4.1.1. Diagramas de Caso de Uso

Os diagramas de caso de uso representam a relação entre os usuários do sistema e os casos de uso, que são as principais funcionalidades do sistema. Os objetivos deste diagrama são: modelar todo contexto do sistema mostrando suas fronteiras; e modelar os requisitos do sistema, em que a ênfase é buscar suas tarefas [25].

Diagramas de caso de uso são compostos por quatro partes:

- Cenário → define a área de atuação do sistema;
- Ator → é um usuário do sistema;
- Caso de Uso → define uma função do sistema;

- Relacionamento → são as relações entre ator e o caso de uso, entre atores e entre casos de uso.
  - Entre ator e caso de uso. Na Figura 25 é apresentada a relação entre o ator e o caso de uso.



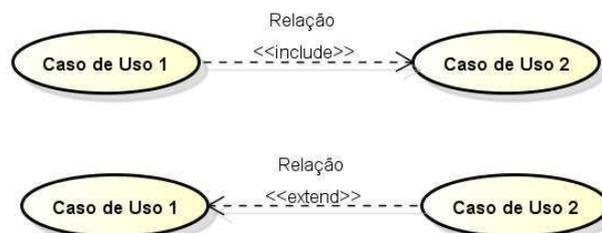
**Figura 25** Relação entre usuário e caso de uso.

- Entre atores, os casos de uso que estiverem contidos no Ator 2 também estarão no Ator 1. Na Figura 26 é apresentada a relação entre os atores.



**Figura 26** Relação entre atores.

- Entre casos de uso. Existem dois tipos desta relação, *include* e *Extend*. O *include* é um relacionamento que indica dependência entre os casos de uso, para o *extend* a relação é uma extensão do caso de uso. Na Figura 27 é apresentada a relação entre os casos de uso.



**Figura 27** Relacionamento entre casos de uso.

### 2.4.1.2. Diagramas de Atividade

Os diagramas de atividades apresentam aspectos dinâmicos do sistema computacional de forma sequencial ou concorrente. Estes são basicamente gráficos de fluxos que modelam as interações entre uma atividade e outra, descrevendo os passos percorridos para conclusão de um método, alterando o estado atual de um sistema [27].

Os diagramas assemelham-se com os fluxogramas, porém apresentados mais detalhadamente o fluxo de controle de uma atividade para outra.

### 2.4.1.3. Diagramas de Classes

Os diagramas de classes apresentam de forma estática as relações internas entre as classes especificando seus atributos, as entidades externas, as interfaces, os tipos de objetos, e os seus relacionamentos. Esses diagramas, em geral, são mais utilizados na modelagem de sistemas orientados a objetos [27].

O diagrama é representado por um retângulo com três divisões, que são: descrição da classe, seus atributos e seus métodos. Na Figura 28 é apresentado a estrutura de um diagrama de classes.

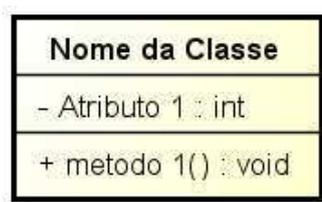


Figura 28 Exemplo de diagrama de classes.

## 2.4.2. Programação orientada a objetos

A Programação Orientada a Objeto (POO) é um padrão de desenvolvimento que utiliza como base o paradigma de orientação a objetos, ou seja, conceito de classificar, organizar e abstrair coisas. Esta linguagem foi criada tentando aproximar o mundo virtual do mundo real, por isso, adotou o conceito de objeto como estrutura principal. As linguagens orientadas a objetos apresentam alguns características, tais como [28]:

- Classes → constituídas de métodos e atributos que representam as características do objeto.
- Atributos → representados pelas variáveis pertencentes à classe.
- Métodos → similares às funções da programação procedural, separando as tarefas que são constituídas de operações e executando-as quando solicitadas;
- Herança → tem como finalidade permitir a reutilização de métodos descritos em classes distintas evitando assim uma duplicidade do código.
- Polimorfismo → permite o envio da mensagem para diferentes objetos e que cada objeto na sua determinada classe executa sua própria operação.

## 2.4.3. Testes de *software*

Os testes de *softwares* são basicamente definidos pela inserção de um conjunto de entradas o qual é comparado com o conjunto de resultados obtidos na saída. Esta comparação define a situação do desenvolvimento do sistema computacional.

O teste é indispensável no desenvolvimento de um *software*, pois contribui para encontrar os possíveis erros existentes no mesmo. Em geral, estes erros sempre

irão existir no desenvolvimento de *software*, pois, em geral, a codificação humana insere falhas em qualquer fase da construção do sistema.

Os testes podem ser estruturados a partir do planejamento do projeto e executados sistematicamente na construção. Mesmo que o sistema passe por uma bateria de testes, em geral, não pode ser afirmado que o sistema está totalmente livre de erros, pois pode existir um teste que não foi validado [29].

#### **2.4.4. Banco de dados**

Os bancos de dados foram criados com a finalidade de aumentar a capacidade de armazenamento de informações e melhorar a organização dos dados. O banco de dados é definido como um conjunto de dados de uma organização que podem ser consultados e alterados, evitando duplicidades de registros e garantindo uma maior proteção nos dados [30].

O banco de dados é constituído por três elementos básicos, que são: os campos, espaço reservado para um dado; registro, representado por um conjunto de campos; e tabelas, utilizadas para armazenar os registros. Para melhor a manipulação dos dados em bancos foram desenvolvidos os Sistemas de Gerenciamento de Banco de Dados (SGBD). O SGDB é basicamente um programa que administra a criação e manipulação dos registros, os mais conhecidos são Oracle, MySQL e PostgreSQL.

### 3. Materiais e Método

Para o desenvolvimento desta dissertação escolheu-se dividi-lo em três etapas, que são: sistema embarcado; comunicação sem fio e *software*. Estas são descritas nas seções seguintes.

#### 3.1. Sistema embarcado

O sistema embarcado possibilita a interação dos usuários ao sistema de tratamento, permitindo configurar os tempos das fases de tratamento e possibilitando habilitar ou desabilitar os sensores. Estas alterações são realizadas por meio de botões acoplados ao sistema embarcado. Para facilitar a visualização e a interação dos dados foi acrescentado um display LCD 16x2.

O tempo das fases do tratamento é contado por um temporizador do PIC 18F4550 que foi configurado para incrementar segundos. Na Figura 29 é apresentado uma estrutura geral em blocos com os principais componentes que compõem o sistema embarcado (*hardware*) proposto.

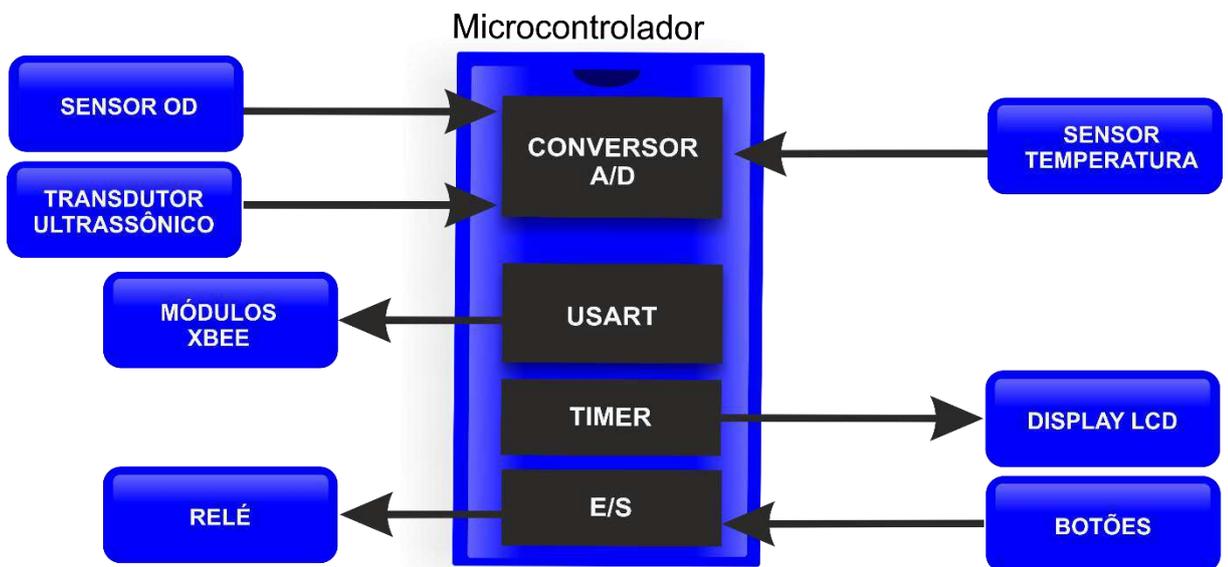


Figura 29 Estrutura dos componentes principais do sistema embarcado.

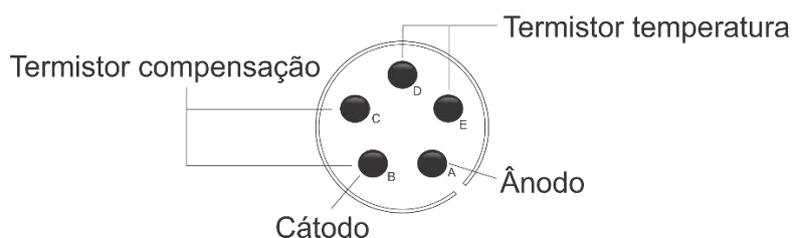
O sensor de OD é utilizado para determinar os períodos de acionamento e desligamento do aerador na fase da aeração. Ou seja, quando OD atinge uma faixa limitante inferior, escolhida pelo usuário, é acionado o aerador que adiciona oxigênio no reator. Com o acréscimo do oxigênio, as bactérias aeróbias, que necessitam de oxigênio, são estimuladas em sua alimentação. Este processo consome de forma gradual o oxigênio inserido e também evita uma alta variação do OD. Porém, quando as bactérias vão ficando inativas, a concentração de oxigênio aumenta até atingir a faixa limitante superior escolhida pelo usuário, indicando ser desnecessário a inserção de mais oxigênio no reator. Foi escolhido o sensor de OD polarográfico YSI 5739 para o desenvolvimento deste trabalho. Na Figura 30 é apresentado o sensor de OD.



**Figura 30 Sensor de OD.**

Foram realizados aferições no sensor com multímetro  $6\frac{1}{2}$  Digit Multimeter para identificar as referências do diagrama de pinos, pois o fabricante não deixa explícito estas informações, apenas relatando a existência de dois sensores termorresistivos. Na Figura 31 é apresentado o diagrama de pinos do sensor OD.

- A - Ânodo
- B - Cátodo
- B,C - Termistor
- D,E - Termistor



**Figura 31 Estrutura do diagrama de pinos do sensor OD.**

Os sinais oriundos do sensor de OD são condicionados para valores de tensão compatíveis com as referências do conversor A/D do PIC. A polarização do sensor é estruturada com um amplificador operacional configurado como um seguidor de tensão com uma entrada no terminal não inversor de aproximadamente -0,8 V. O condicionamento do sinal do ânodo nos permite uma corrente máxima de 25 $\mu$ A, pois a tensão máxima de entrada do PIC é 5V. Na Figura 32 é apresentado o condicionamento para obtenção do OD.

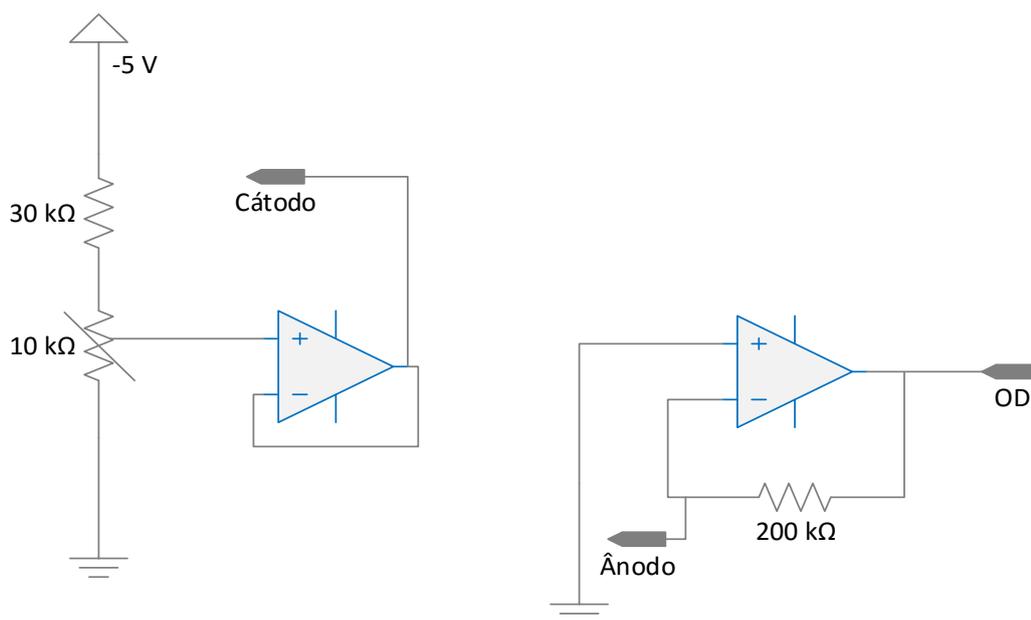


Figura 32 Condicionamento do Sensor OD(adaptado [19]).

O sensor YSI 5739 apresenta em seu encapsulamento dois sensores de temperatura, os quais não são especificados pelo fabricante. Então, para descobrir a relação da resistência com a temperatura foram desenvolvidos experimentos para obtenção de tal relação.

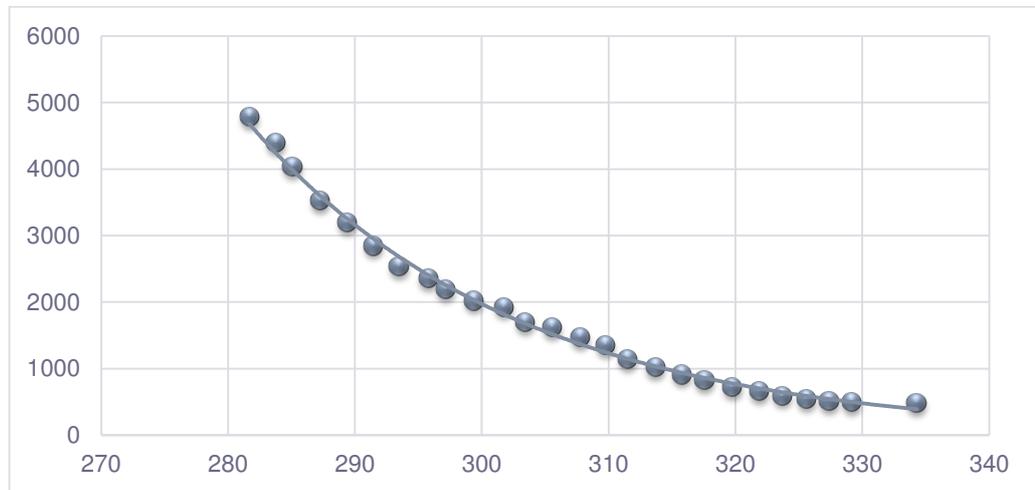
Foi realizado a aferição da resistência do sensor termorresistivo a partir da variação da temperatura em uma escala entre 9 a 62 °C, com a finalidade de obter uma relação matemática que ajuste esta relação. No Quadro 7 é apresentada a relação entre temperatura e a resistência medida pelo multímetro.

Quadro 7 Relação da resistência com a temperatura do sensor.

Temperatura (K)	Resistência ( $\Omega$ )
281,7	4780
283,8	4390
285,1	4040
287,3	3530
289,4	3196
291,5	2845
293,5	2540
295,8	2360
297,2	2190
299,4	2019
301,8	1932
303,4	1705
305,6	1618
307,8	1470
309,8	1348
311,5	1147
313,7	1023
315,8	912
317,6	826
319,8	729
321,9	656
323,7	596
325,6	546
327,4	516
329,2	495
334,3	484

Após o experimento percebeu-se que o sensor termorresistivo possui características de um termistor NTC, pois a temperatura é inversamente

proporcional a resistência elétrica. Na Figura 33 é apresentado de forma gráfica a relação entre a temperatura e a resistência elétrica obtidas no experimento.



**Figura 33** Relação entre a temperatura e a resistência elétrica do sensor.

A partir da expressão 4 é possível utilizar os valores experimentais para encontrar as constantes A e B. Foi realizado inicialmente uma alteração na expressão 4 para encontrar uma relação matemática que a representasse como uma equação da reta reduzida. Na expressão 5 é apresentado a fórmula geral da equação da reta reduzida.

$$Y = aX + b. \tag{5}$$

Foi aplicada na expressão 4 o logaritmo natural nos dois lados, chegando assim na expressão 6.

$$\ln R = \ln A + \ln e^{\frac{B}{T}} \qquad \ln R = \ln A + \frac{B}{T} \tag{6}$$

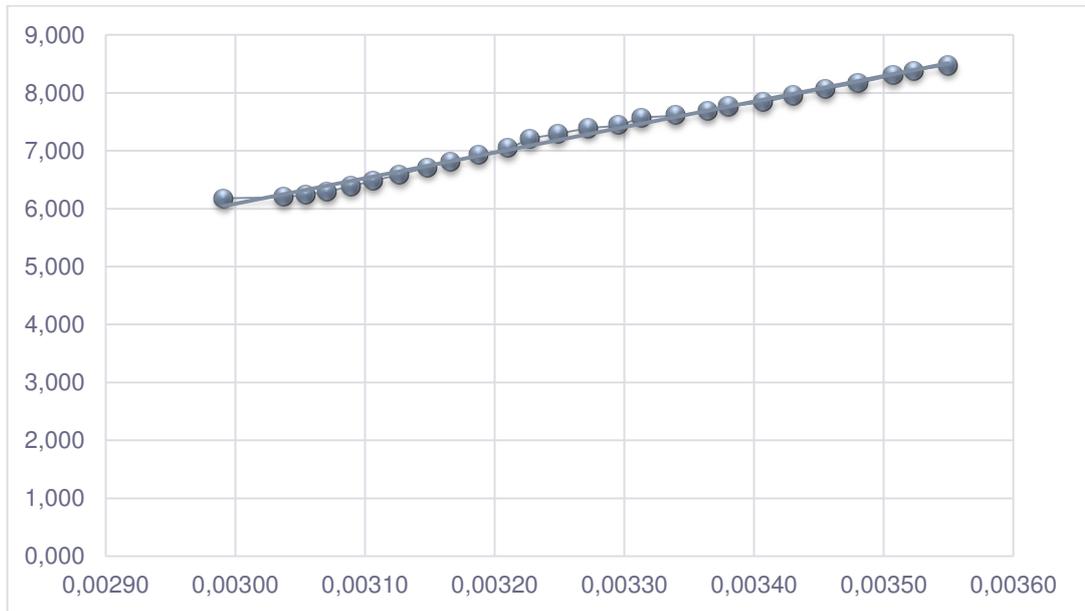
Adotando a similaridade entre as expressões 5 e 6, tem-se que:

$$Y = \ln R \qquad a = B \qquad b = \ln A \qquad X = \frac{1}{T} \tag{7}$$

Utilizando os valores fornecidos da temperatura e resistência elétrica no Quadro 7 nas equações de  $Y$  e  $X$  encontradas na expressão 7, são obtidos pontos experimentais com comportamento linear como apresentado no Quadro 8 e na Figura 34.

**Quadro 8** Relação  $1/T$  e  $\ln(R)$ .

<b>1/T</b>	<b>ln(R)</b>
0,00355	8,472
0,00352	8,387
0,00351	8,304
0,00348	8,169
0,00346	8,070
0,00343	7,953
0,00341	7,840
0,00338	7,766
0,00336	7,692
0,00334	7,610
0,00331	7,566
0,00330	7,441
0,00327	7,389
0,00325	7,293
0,00323	7,206
0,00321	7,045
0,00319	6,930
0,00317	6,816
0,00315	6,717
0,00313	6,592
0,00311	6,486
0,00309	6,390
0,00307	6,303
0,00305	6,246
0,00304	6,205
0,00299	6,182



**Figura 34 Gráfico relacionando 1/T e ln(R).**

Utilizando o método dos mínimos quadrados com os valores obtidos no Quadro 8, são calculados os valores do coeficiente linear ( $b$ ) e angular ( $a$ ) da equação da reta. Estes cálculos são descritos nas expressões 8 e 9.

$$a = \frac{N * \sum_{i=1}^N x_i * y_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i}{N \sum_{i=1}^N x_i^2 - (\sum_{i=1}^N x_i)^2} = 4417,7. \quad (8)$$

$$b = \frac{\sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i - \sum_{i=1}^N x_i * y_i \sum_{i=1}^N x_i}{N \sum_{i=1}^N x_i^2 - (\sum_{i=1}^N x_i)^2} = -7,169. \quad (9)$$

Para verificar a confiabilidade da relação entre os pontos e a reta encontrada é calculado o coeficiente de determinação ( $R^2$ ) dos valores experimentais do Quadro 8. Na expressão 10 é apresentado o coeficiente de determinação ( $R^2$ ) da relação.

$$R^2 = \left( \frac{N \cdot \sum_{i=1}^N x_i \cdot y_i - \sum_{i=1}^N x_i \cdot \sum_{i=1}^N y_i}{\sqrt{N \sum_{i=1}^N x_i^2 - (\sum_{i=1}^N x_i)^2} \cdot \sqrt{N \sum_{i=1}^N y_i^2 - (\sum_{i=1}^N y_i)^2}} \right)^2 = 0,992. \quad (10)$$

O coeficiente de determinação caracteriza que 99,2% da variação de Y pode ser explicada pela equação da reta encontrada e que apenas 0,08% não podem ser explicadas.

A relação entre a temperatura e a resistência elétrica do termistor NTC acoplado ao sensor OD é apresentada na expressão 11.

$$R = A \cdot e^{B \cdot \left(\frac{1}{T}\right)} \rightarrow R = 0,0008 \cdot e^{\left(\frac{4417,7}{T}\right)}. \quad (11)$$

Sabendo da expressão que relaciona a resistência elétrica do termistor NTC a temperatura, é necessário condicionar o circuito para que o valor da resistência possa ser relacionado a uma tensão compatível a entrada do PIC de até 5 V. Na Figura 35 é apresentado o circuito de condicionamento do sensor de temperatura acoplado ao sensor de OD.

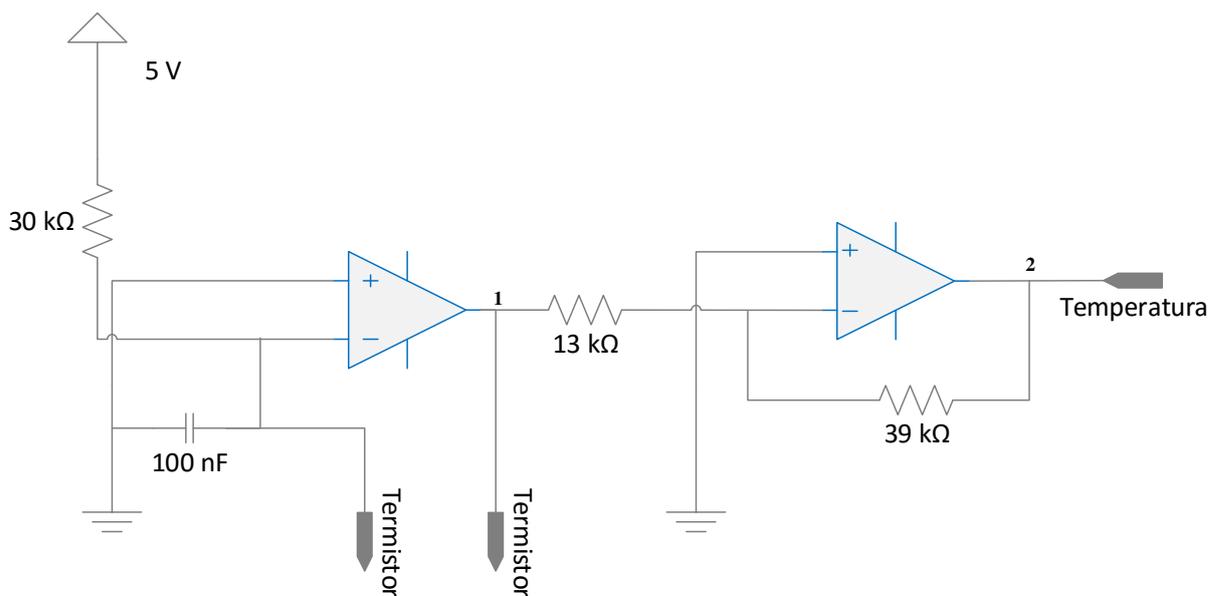


Figura 35 Condicionamento do sensor de temperatura (adaptado [19]).

A tensão de saída do amplificador operacional no ponto 1 é descrita pela Expressão 12. Em que  $R_t$  corresponde a resistência do termistor NTC.

$$V_1 = -R_t * \frac{5}{30 \times 10^3}. \quad (12)$$

Para o ponto 2 é encontrado a expressão 13.

$$V_2 = -39 \times 10^3 * \frac{V_1}{13 \times 10^3}. \quad (13)$$

Logo, a tensão  $V_2$  pode ser descrita como:

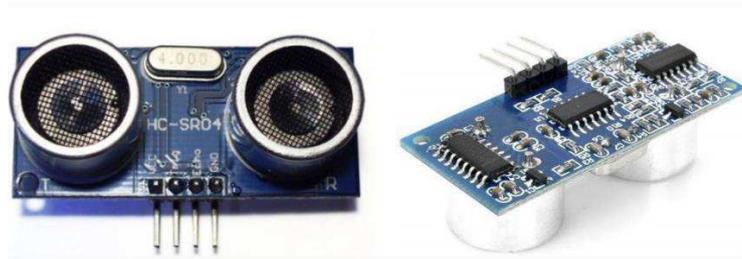
$$V_2 = 5,04 * 10^{-4} R_t. \quad (14)$$

A Expressão 14 relaciona a tensão de saída do circuito com a resistência elétrica do termistor NTC. Esta relação combinada com a equação da reta e os valores encontrados dos coeficientes linear e angular nos possibilita o cálculo da temperatura aferida pelo sensor. Na expressão 15 é apresentada a relação da temperatura com a tensão de saída do circuito de condicionamento.

$$\ln R = \ln A + \frac{B}{T} \rightarrow T = B / (\ln R - \ln A) \rightarrow \quad (15)$$

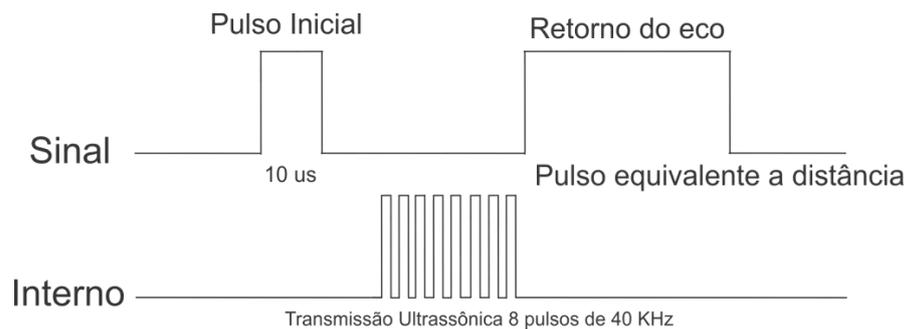
$$T = 4417,7 / (\ln(1984,13 * V_2) - \ln(0,0008)).$$

O transdutor ultrassônico HC-SR04 é utilizado para indicar o nível do reator, estimando o momento da ativação e desativação da eletrobomba de alimentação. O HC-SR04 possui algumas características importantes, que são: emissor e receptor de ondas de altas frequências; alcance entre 2 a 400 cm; resolução de 0,3 cm e ângulo efetivo de 15°. Na Figura 36 é apresentada a estrutura do transdutor HC-SR04.



**Figura 36 Transdutor Ultrassônico (HC-SR04).**

Este transdutor é estruturado com quatro pinos: o primeiro é utilizado para alimentação do transdutor com 5 V; o segundo é o pulso inicial (*trigger*), inicialmente é necessário inserir um pequeno pulso 10  $\mu$ s; o terceiro, denominado retorno (*echo*), é responsável pelo sinal de resposta do transdutor que é proporcional à distância do objeto; e o quarto, é o terra do sensor HC-SR04 [31]. Na Figura 37 é apresentado de forma gráfica o funcionamento do transdutor.



**Figura 37 Funcionamento do HC-SR04 [21]**

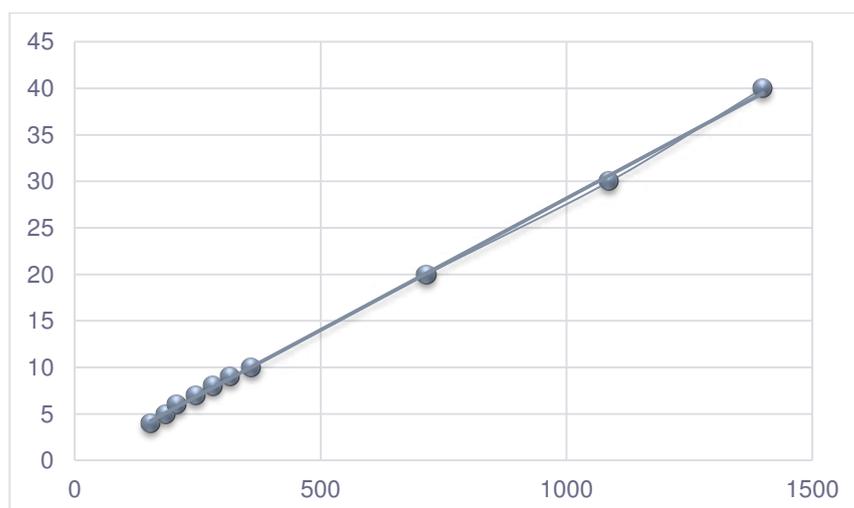
Para obter a distância que o transdutor está aferindo é necessário calcular o tempo que o sinal de retorno (retorno do eco) está em nível alto. Por este motivo, é configurado o temporizador, *Timer 1*, do microcontrolador para realizar esta contagem. A fórmula padrão para cálculo da distância é apresentado na expressão 16.

$$Distância = (tempo\ gasto * Velocidade\ do\ som) / 2. \quad (16)$$

Porém a expressão 16 não abrange todo escopo do transdutor porque não leva em consideração variáveis como a temperatura do ar, umidade relativa e pressão atmosférica. Então, para obter uma maior exatidão na medida do sensor foram realizados experimentos para encontrar uma regressão linear que relacionasse o tempo a distância. No Quadro 9 e na Figura 38 são apresentados a relação experimental entre o tempo do temporizador e a distância encontrada.

**Quadro 9** Relação do tempo e distância.

Tempo(CC)	Distância(cm)
154	4
186	5
208	6
248	7
282	8
316	9
360	10
716	20
1088	30
1400	40

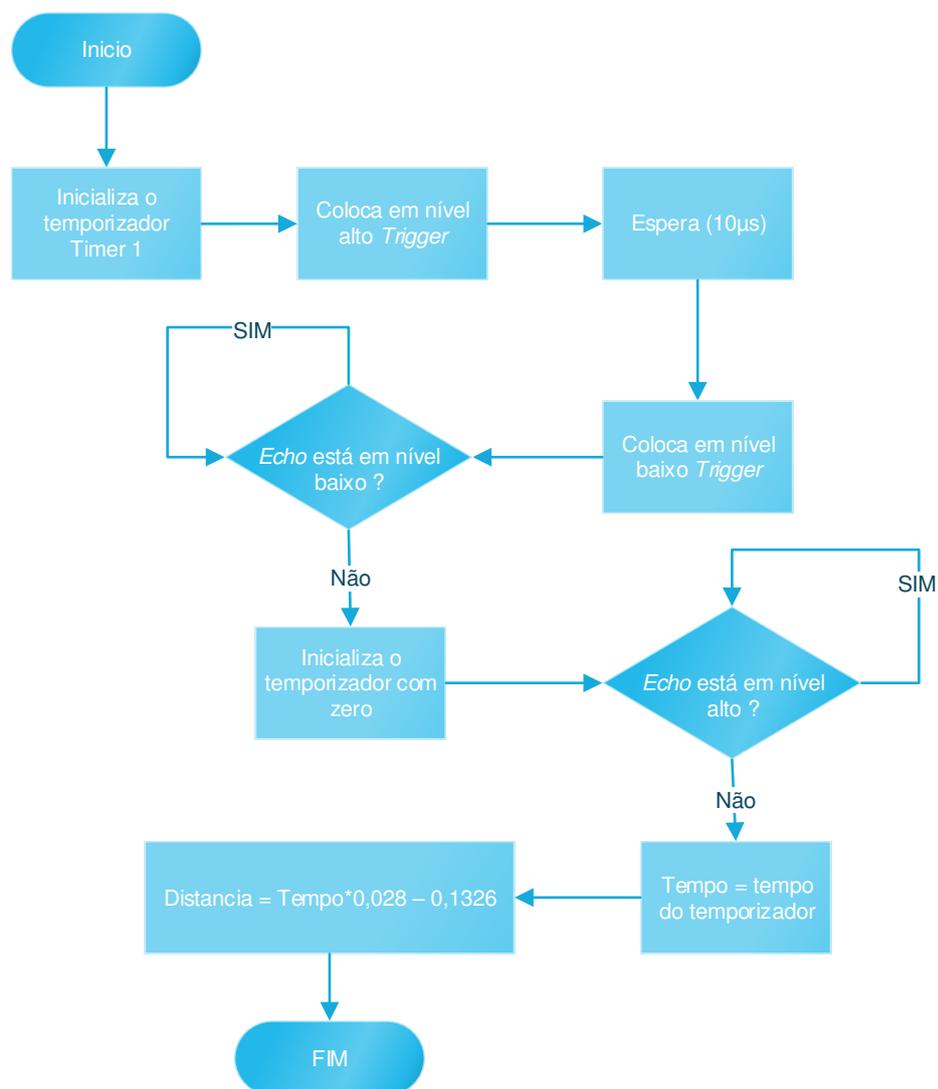


**Figura 38** Relação Gráfica entre tempo e distância.

Nota-se que a partir dos valores obtidos é possível montar uma expressão linear que represente significativamente a relação. Na Expressão 17 é apresentada a relação de  $y$  (equivale a distância) e  $x$  (o tempo medido pelo temporizador).

$$y = 0,0283x - 0,1326. \quad (17)$$

A configuração do cálculo da distância no *software* do sistema embarcado foi estruturado conforme o fluxograma apresentado na Figura 39.



**Figura 39 Fluxograma do cálculo da distância.**

Relés eletromagnéticos são utilizados para acionar: a eletrobomba; aerador; motor de giro e na válvulas eletrônicas dos reatores. Esses são acionados com uma diferença de potencial de 5 V nos pinos 1 e 3, isto é em sua bobina, gerando um campo magnético o qual permite haver chaveamento entre o pino 5 e 4, permitindo a comutação do sinal no pino 2. Na Figura 40 é apresentada a estrutura do relé.

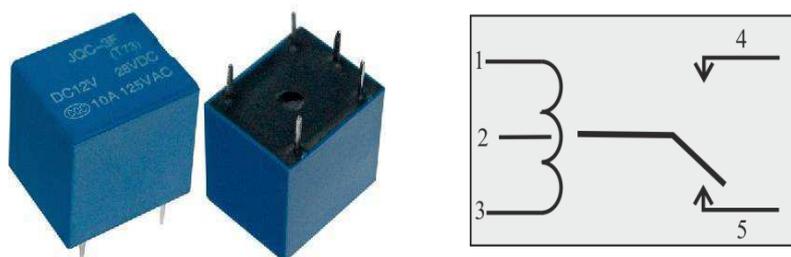


Figura 40 Estrutura do Relé.

O microcontrolador PIC não fornece uma corrente de saída no pino necessária para acionar o relé, por isso, precisou-se utiliza um transistor funcionando como uma chave. Na Figura 41 é apresentado o condicionamento necessário para o acionamento do relé.

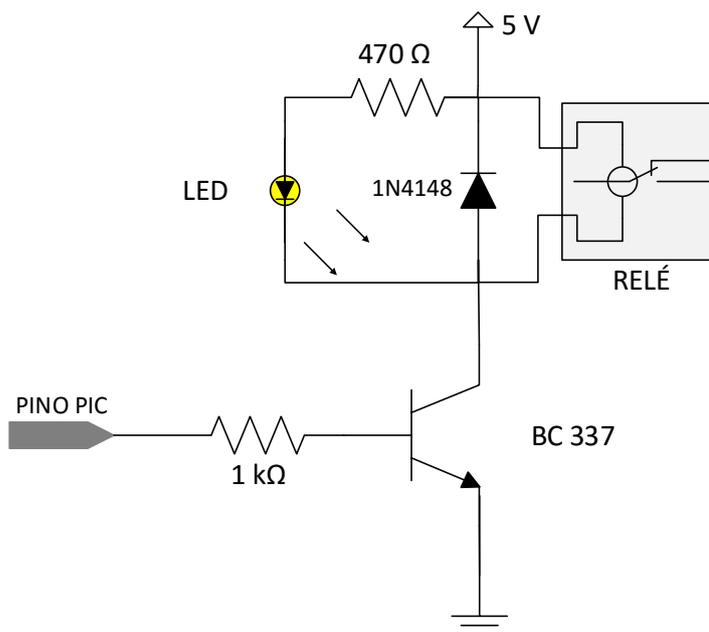


Figura 41 Condicionamento do acionamento do relé.

O display LCD 16x2 é utilizado para apresentar de forma visual as informações do sistema aos usuários, contribuindo também na configuração dos tempos de cada fase como na visualização da situação corrente do sistema. Na Figura 42 é apresentado o modelo do display LCD utilizado.

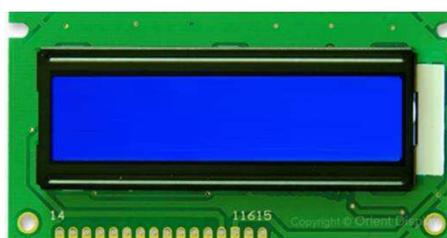


Figura 42 Modelo do display LCD.

Para enviar os dados do microcontrolador para o LCD foi utilizado a porta D do PIC para configurar (pinos RS, RW e E) e enviar os dados (pinos D4 a D7) para o LCD. O ajuste do brilho do LCD é realizado por um potenciômetro de 10 k $\Omega$  no terceiro pino do LCD. Na Figura 43 é apresentado o diagrama elétrico do LCD.

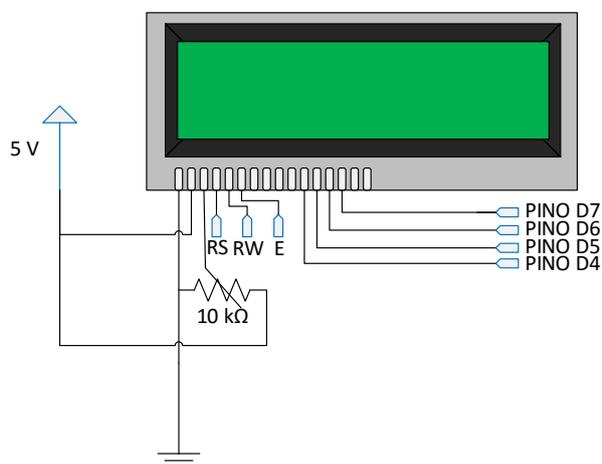


Figura 43 Condicionamento do Display LCD.

O gerenciamento dos atuadores, sensores e todo processo de configuração do sistema são realizados por um *software* instalado no microcontrolador. Este foi desenvolvido na linguagem de programação C. Na Figura 44 é apresentado o fluxograma geral da estrutura do *software* instalado no PIC.

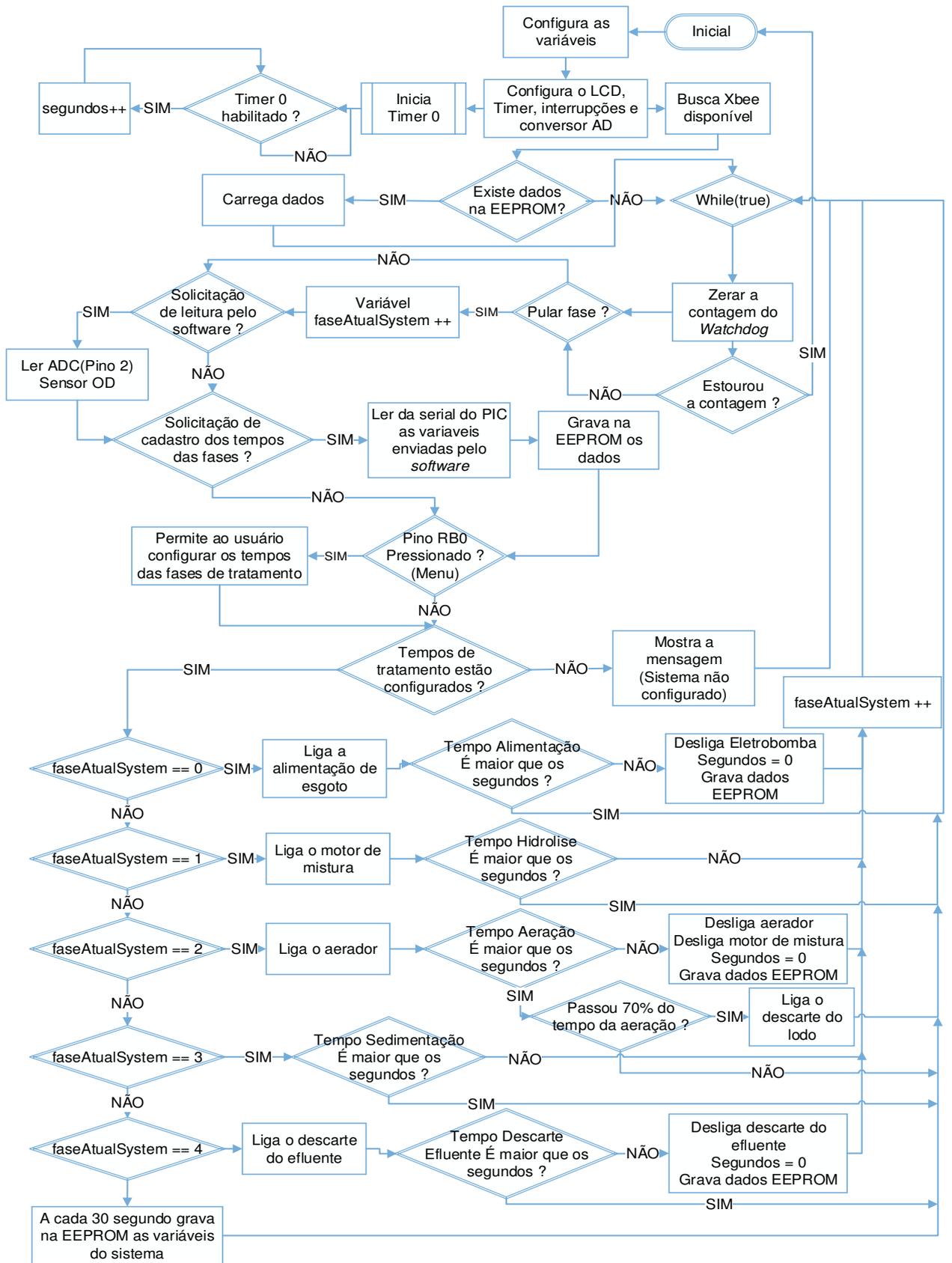


Figura 44 Fluxograma do *software* no microcontrolador.

### 3.2. Comunicação sem fio

Os módulos XBee estão disponíveis nas séries 1 e 2. Eles apresentam a mesma estrutura física com pinos e tamanho iguais, porém os *firmwares* dos dispositivos são diferentes. A série 1 se baseia no *chipset freescale* e a série 2 se baseia no *chipset Ember* impossibilitando a comunicação entre estes dispositivos.

Para solucionar o objetivo deste pesquisa, dentre os dispositivos existentes, foi escolhido o módulo ZigBee XBP24-ZB. Este é alocado a dispositivos da série 2 com a antena de transmissão e recepção *wire*, a qual apresenta um alcance suficiente para aplicação desta pesquisa. Na Figura 45 é apresentado o modelo do ZigBee escolhido para o desenvolvimento deste estudo.



Figura 45 Dispositivo ZigBee XBP24-ZB [32].

As principais características do dispositivo são descritas no Quadro 10.

Quadro 10 Características do modelo ZigBeeXBP24-ZB [32].

<b>Frequência</b>	2,4 GHz
<b>Potência de transmissão</b>	50 mW
<b>Alcance máximo estimado</b>	1,6 km
<b>Topologias de rede</b>	Start, Mesh, Tree
<b>Taxa de transmissão</b>	250 Kbps
<b>Segurança</b>	128-bit AES
<b>Modo de operação</b>	AT e API

Para a configuração dos módulos XBee a empresa *Digi International* desenvolveu um *software*, denominado X-CTU, que permite a configuração dos dispositivos de forma simples e amigável para os usuários. Este *software* é

disponibilizado, gratuitamente, pela própria empresa. Na Figura 46 é apresentada a tela principal do *software* X-CTU.

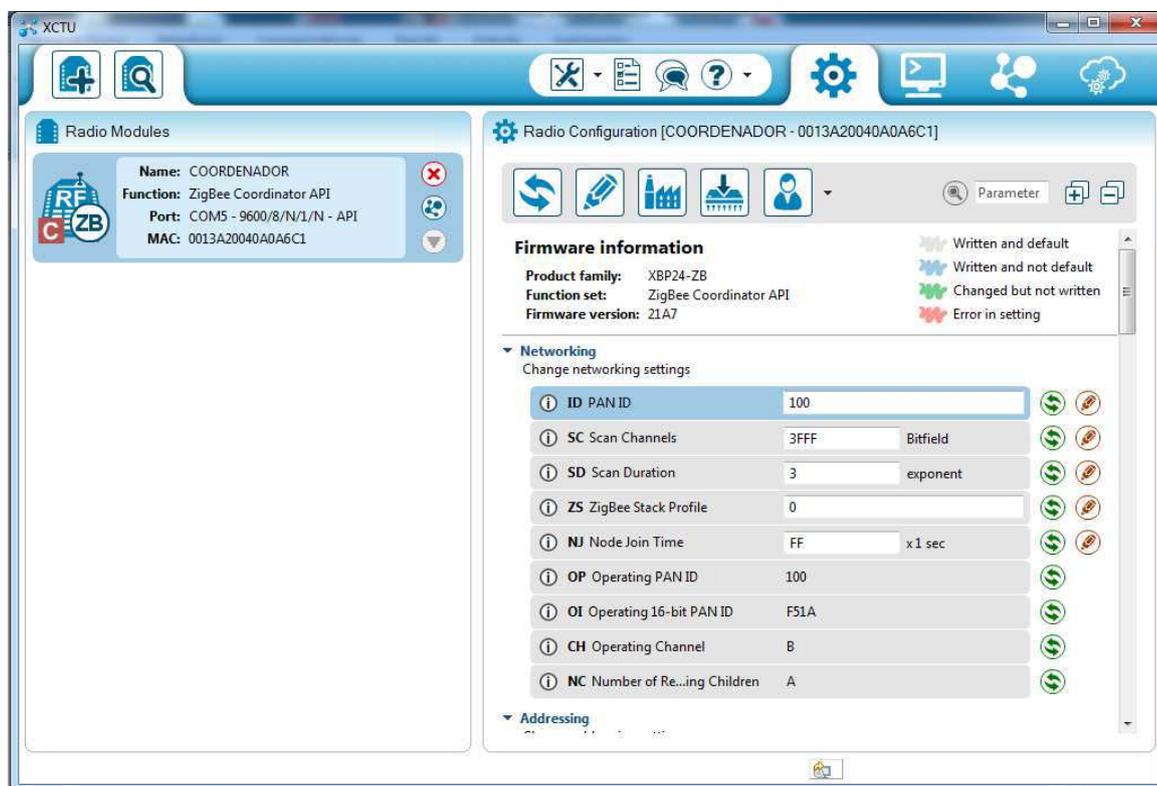


Figura 46 Programa X-CTU.

No bloco esquerdo da Figura 46, em *Radio Modules*, são listados os módulos conectados ao computador. Nesta primeira visualização é possível observar o nome, função, endereço e a porta em que o dispositivo foi conectado.

No bloco direito da Figura 46 é possível realizar a configuração dos parâmetros do XBee no X-CTU. Isto acontece quando pressionado o botão *Radio Configuration*, em que permite selecionar o firmware que será instado no XBee, o tipo de operação, o tipo do dispositivo e as variáveis do XBee. Estas são algumas variáveis principais, que são: nome para o dispositivo (NI); definir o endereço de destino da mensagem (DL e DH); canal de comunicação (CH); endereço de rede (PAN ID); velocidade de comunicação (*baud rate*); comportamento dos pinos de entrada e saída e o tipo de criptografia dos dados.

A rede XBee foi configurada para uma topologia estrela, em que o coordenador foi conectado ao computador e os reatores, que são dispositivos finais, ficam

conectados ao seu respectivo reator. Porém para análise experimental foi utilizado apenas um reator. Na Figura 47 é apresentado a estrutura da rede XBee.

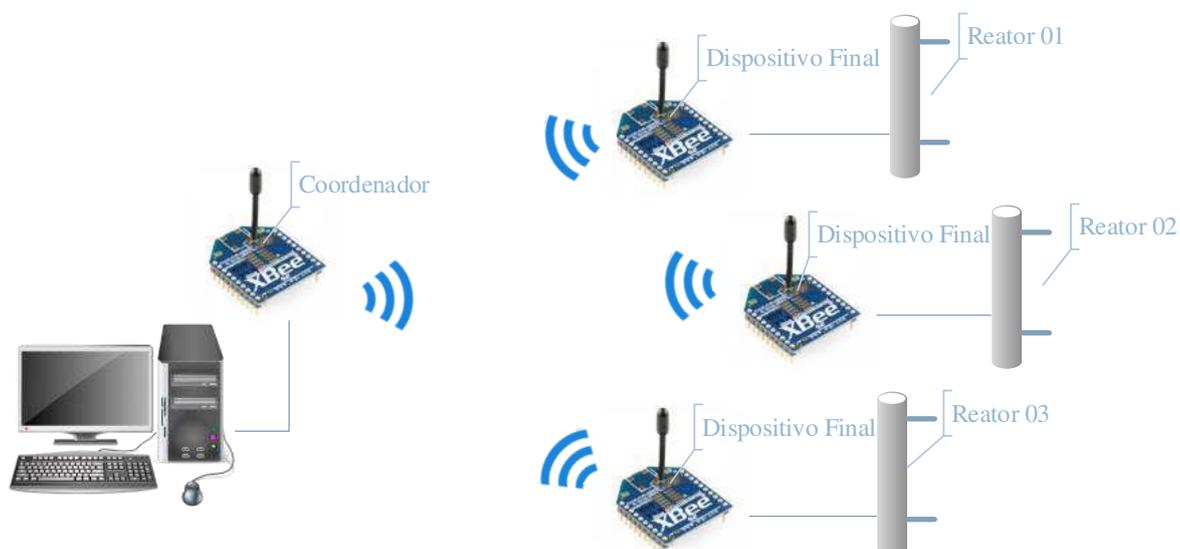


Figura 47 Estrutura da Rede XBee.

Para verificar a confiabilidade da transmissão de dados entre os dispositivos foi realizado um experimento aumentando a distância entre o coordenador e o dispositivo final. O experimento foi baseado no envio de 10 pacotes de dados pelo dispositivo final e verificado seu recebimento pelo coordenador. Para a finalidade desta pesquisa, foi testado até uma distância de 100 metros, o qual não apresentou perdas de pacotes.

### 3.3. Software

O desenvolvimento do *software* foi realizado na linguagem de programação JAVA, trazendo para o sistema vantagens como: programação orientada a objetos, reutilização de códigos; codificação estruturada; portabilidade, a qual admite a instalação em várias plataformas; e a utilização de multi-thread.

O sistema computacional foi desenvolvido com a finalidade de facilitar o usuário na configuração das variáveis de tratamento de águas residuárias, como também visualizar os dados em tempo real no computador. Este sistema possui telas fáceis

de se utilizar, possibilitando para o usuário maior agilidade e organização nos dados obtidos no projeto.

Estes dados são transmitidos pelo sistema embarcado, que contém um módulo XBee configurado como dispositivo final, e enviados para módulo XBee coordenador que é acoplado a porta USB do computador. Os dados recebidos são tratados e enviados para um banco de dados construído localmente, ou seja, o banco de dados será instalado no computador que o sistema esteja executando.

Para a codificação do *software* foi utilizado uma ferramenta gratuita denominada NetBeans IDE, que atualmente está na versão 8.0.2, fornecendo aos desenvolvedores os instrumentos necessários para a criação de aplicativos computacionais de alto desempenho. Esta ferramenta é utilizada para uma vasta quantidade de desenvolvimento de aplicativos para desktop, para web e para dispositivos móveis.

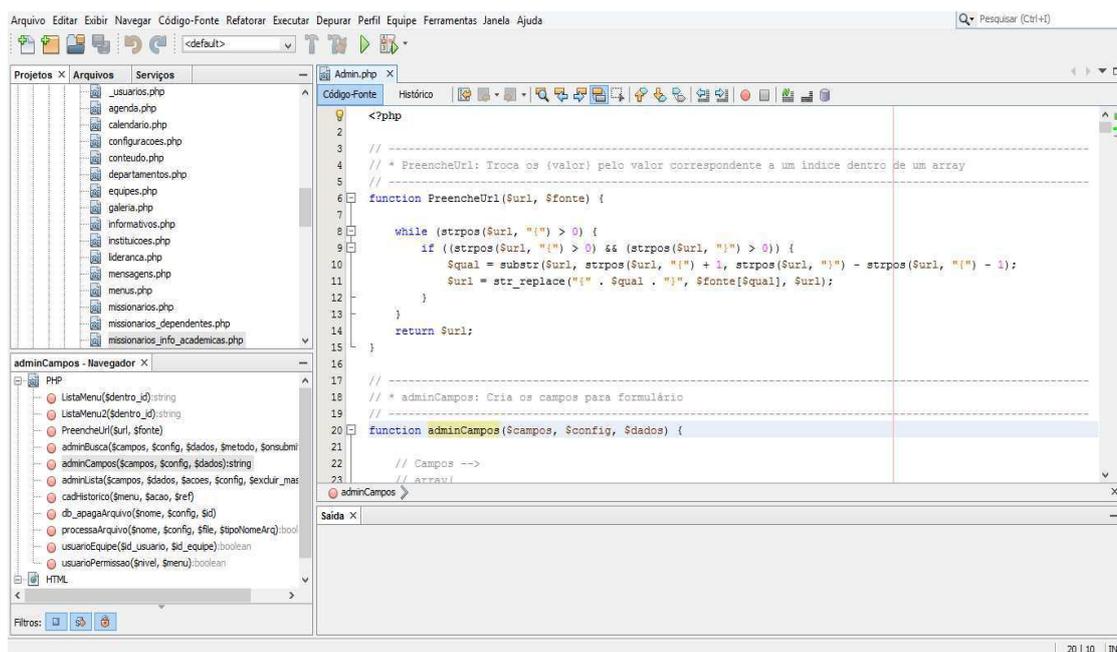


Figura 48 Ferramenta para desenvolvimento NetBeans.

Para o desenvolvimento dos diagramas foi utilizado a ferramenta *Astah* que possui a estrutura necessária para a modelagem. A vantagem de sua utilização é a distribuição de uma versão gratuita, conhecida como *Astah Community*.

Foram construídos os diagramas de caso de uso, atividade e classes, os quais são foco deste projeto. Na Figura 49 é apresentado o diagrama de casos de uso com as principais ações do usuário ao sistema computacional desenvolvido.

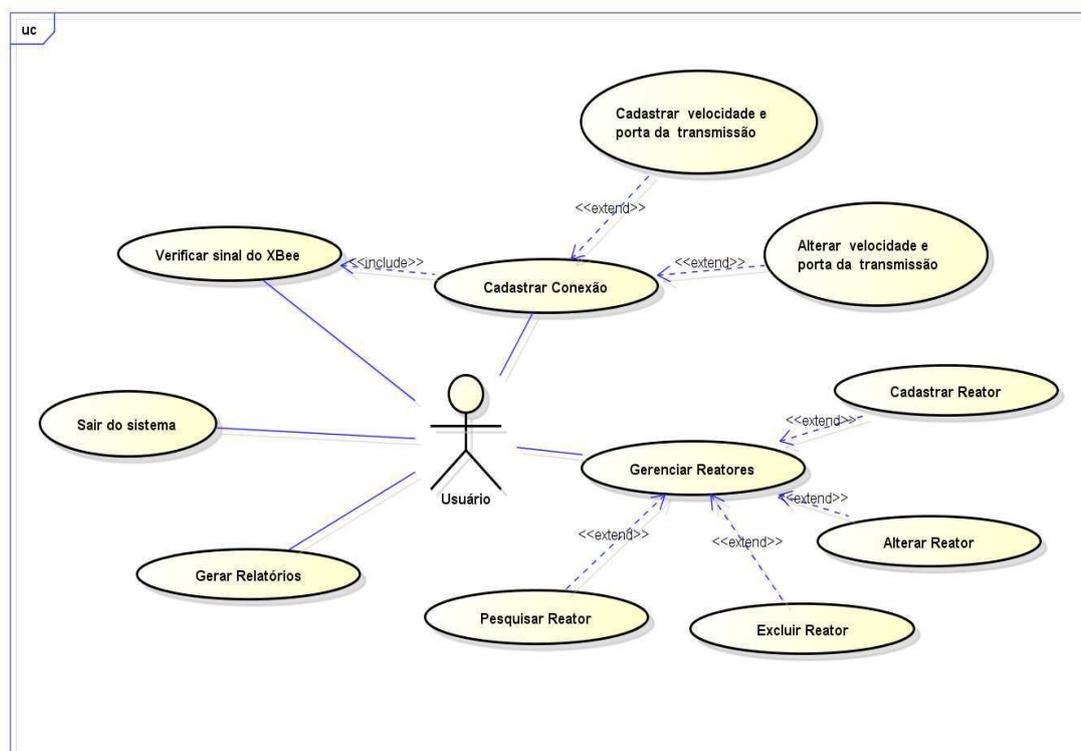


Figura 49 Diagrama de caso de uso do sistema.

Os diagramas de atividades e de classes estão descritos nos Apêndices A e B respectivamente.

Para o armazenamento dos tempos gastos nas fases no tratamento do esgoto foi estruturado um banco de dados com a ferramenta DBDesigner, *open source*, que permiti a modelagem e estruturação do banco de dados. Esta ferramenta possibilita a exportação do SQL para vários sistemas de gerenciamentos de banco de dados, como: postgresQL e MySQL. Foi utilizado o SGBD postgresQL para esquematizar as tabelas do banco para o armazenamento dos dados necessários. Na Figura 50 é apresentada o esquema do banco de dados desenvolvido para o sistema computacional no DBDesigner.

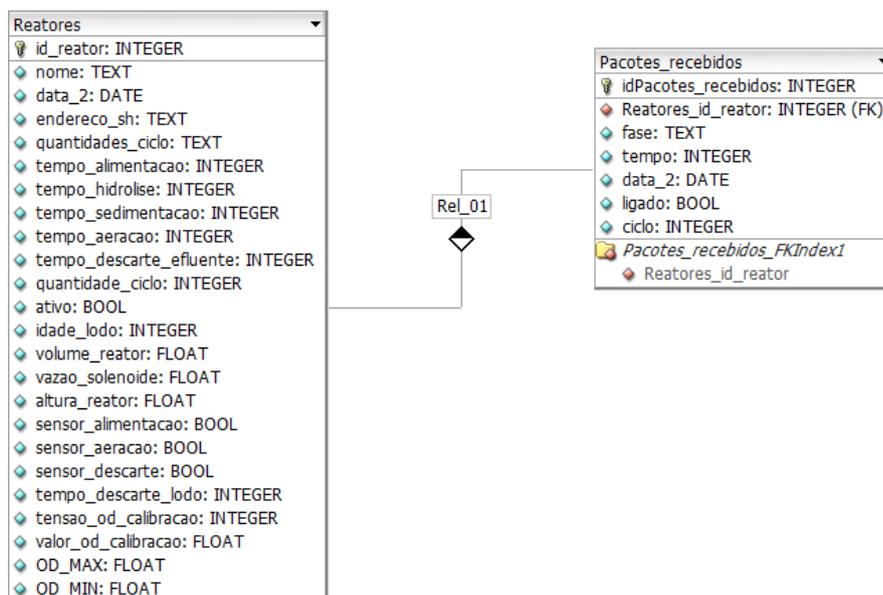


Figura 50 Estrutura do banco de dados desenvolvido.

### 3.4. Cuidados para o experimento

Alguns cuidados são necessários para a realização do experimento. É necessário que o operador do sistema esteja utilizando os Equipamentos de Proteção Individual – EPIs, evitando o contato direto com os resíduos. Dentre eles podem ser citados: luvas de borracha, óculos, máscara e jaleco. Na Figura 51 é apresentado os EPIs para a realização do experimento.



Figura 51 EPIs necessárias para o experimento.

## 4. Resultados

Neste capítulo são apresentados e discutidos os resultados obtidos a partir da implementação da automação e do monitoramento de reatores de bateladas sequenciais para tratamento de águas residuárias. A exposição dos resultados foi dividida em *software*, sistema embarcado e análise química do esgoto.

### 4.1. Software

Foi desenvolvido um sistema computacional capaz de armazenar as variáveis do sistema de tratamento de águas residuárias. Estas informações podem ser vistas em tempo real como em relatórios posteriores gerados pelo sistema. Após a estruturação dos diagramas foi possível a codificação do sistema de forma mais rápida e ágil.

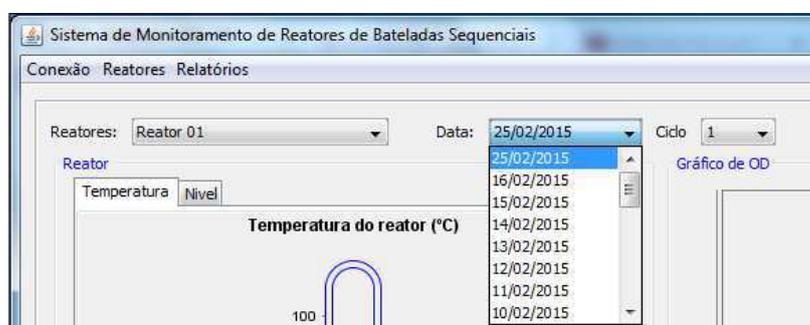
Neste capítulo são apresentadas as telas que compõem o sistema computacional. Na Figura 52 é apresentada a tela principal do sistema computacional desenvolvido.



Figura 52 Tela principal do sistema computacional.

Observar que a tela principal é subdividida em quatro blocos, que são: reator, apresenta as configurações em tempo real do nível e da temperatura do reator; pacotes, mostra a estrutura dos pacotes recebidos pelo sistema embarcado por meio do XBee; estado do sistema, apresenta em tempo real a fase que está o tratamento e o tempo restante da fase; e o Gráfico OD, capaz de mostrar em tempo real a medida de OD na fase de aeração do sistema.

No topo da tela principal é possível buscar informações de outros reatores, previamente configurados, como datas de tratamento ou até o ciclo de tratamento. Na Figura 53 é apresentado o topo da tela principal do sistema computacional.

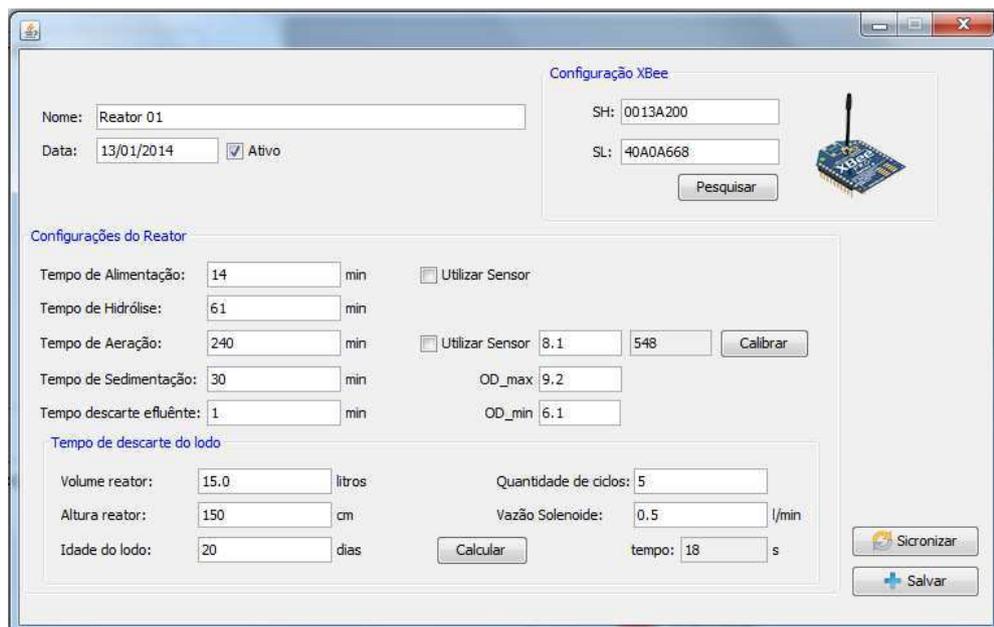


**Figura 53** Topo da tela principal.

O *software* foi desenvolvido visando a interação de vários reatores. Por isso foi desenvolvido uma tela que é capaz de fazer o cadastramento destes reatores. Nesta mesma tela foi programado para realizar as alterações das variáveis do reator caso necessário.

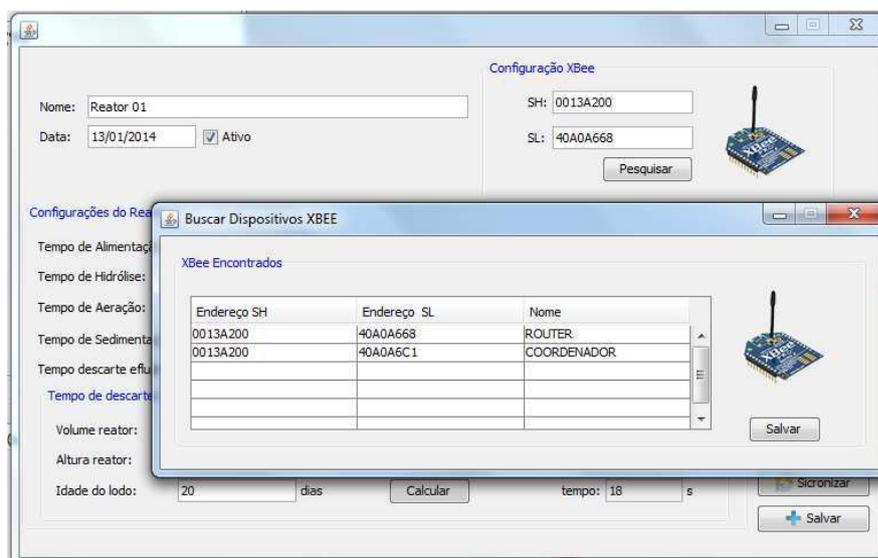
Nesta tela é possível também calibrar o sensor de OD. Ao pressionar o botão calibrar é enviado uma solicitação via RF para o sistema embarcado enviar o valor do conversor AD correspondente ao do sensor OD.

Caso seja pressionado o botão sincronizar é enviado um pacote de dados pelo Xbee para o sistema embarcado passando todas as informações dos tempos de tratamento e a calibração do sensor OD. Na Figura 54 é apresentada a tela de cadastro dos reatores.



**Figura 54 Tela de cadastro dos reatores.**

Na tela de cadastro é possível pesquisar os dispositivos Xbee conectados à rede, isto é, ao pressionar o botão pesquisar é solicitado ao Xbee realizar o comando ATND buscando os endereços de todos os dispositivos Xbee conectados à mesma rede. Na Figura 55 é apresentada a tela de busca de Xbee.



**Figura 55 Tela de busca de Xbee.**

Foi desenvolvido um módulo no sistema computacional capaz de gerar relatórios no formato de arquivo PDF dos ciclos de tratamentos, permitindo ao

usuário ter análises mais detalhadas. Neste módulo é necessário escolher o reator, selecionar o período do relatório e pressionar a tecla gerar. Na Figura 56 é apresentada a tela de gera relatórios.

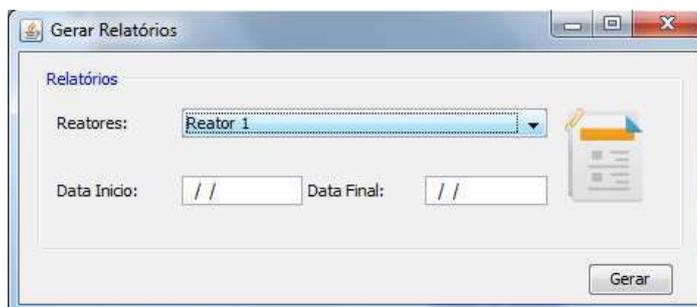


Figura 56 Tela gerar relatórios.

## 4.2. Sistema embarcado

Para a obtenção dos resultados do sistema embarcado decidiu-se dividir o projeto em duas fases de construção.

### 4.2.1. Primeira Fase

Foi desenvolvido um protótipo para a automação do sistema embarcado com a finalidade de verificar o comportamento de cada fase do tratamento de águas residuárias, como também, testar a eficiência do *software* instalado no microcontrolador.

Foi construída uma placa de circuito impresso, com botões, que permite os usuários configurar manualmente os tempos de cada fase do tratamento. Após a configuração das fases, o sistema embarcado fica autônomo para os acionamentos das eletrobombas e válvulas eletrônicas. Na Figura 57 é apresentada a estrutura do protótipo desenvolvido.



Figura 57 Estrutura do protótipo primeira fase do projeto

Esta etapa contribuiu bastante para o projeto final, pois foi nela que foi encontrado alguns erros de construção como de *software*. Foi analisado nesta fase que existia um erro no condicionamento do regulador de tensão, causando um aquecimento excessivo no componente, como também, foi observado que quando o sistema era desligado por algum problema na rede elétrica, o *software* não permitia voltar para o mesmo estágio de tratamento de esgoto que estava anteriormente.

#### 4.2.2. Fase Final

Nesta fase foram solucionados todos os problemas encontrados na primeira fase. Para a obtenção da fase final escolheu-se subdividir o sistema embarcado em 4 partes, que são: placa central; do condicionamento dos sensores; de acionamento; fonte de alimentação; e da comunicação com Xbee.

- **Placa da Fonte de alimentação**

Foi desenvolvida uma fonte simétrica capaz de fornecer tensões de 12 e -12 V, 5 e -5 V e variável entre 1,2 até 10 V. Na Figura 58 é apresentada a placa da fonte de alimentação do sistema embarcado.



Figura 58 Placa da fonte de alimentação.

Os diagramas elétricos da placa de alimentação do sistema são apresentados no Apêndice C.

- **Placa de Comunicação com XBee**

Foi desenvolvida uma placa para a transmissão dos dados do microcontrolador via RF. Nesta placa é condicionada a tensão de alimentação e a tensão de entrada, no pino Din, do dispositivo XBee, pois é recomendável utilizar uma tensão máxima de 3,3 V. Na Figura 59 é apresentado a placa de comunicação com o dispositivo XBee.



Figura 59 Placa da comunicação XBee.

O diagrama elétrico da placa da comunicação com XBee é apresentado no Apêndice D.

- **Placa Central**

A placa central é onde existe todo o gerenciamento do sistema, é nela que os sinais oriundos dos sensores e os sinais dos botões são processados para posteriores tomadas de decisões. Dentre estas são: acionamentos dos relés, o envio dos dados para o LCD, a entrada dos sinais dos botões, entrada e saída dos dados via RF, entrada do sinal do sensor OD, da temperatura e do nível do reator. Na Figura 60 é apresentada a estrutura da placa central do sistema embarcado desenvolvido.

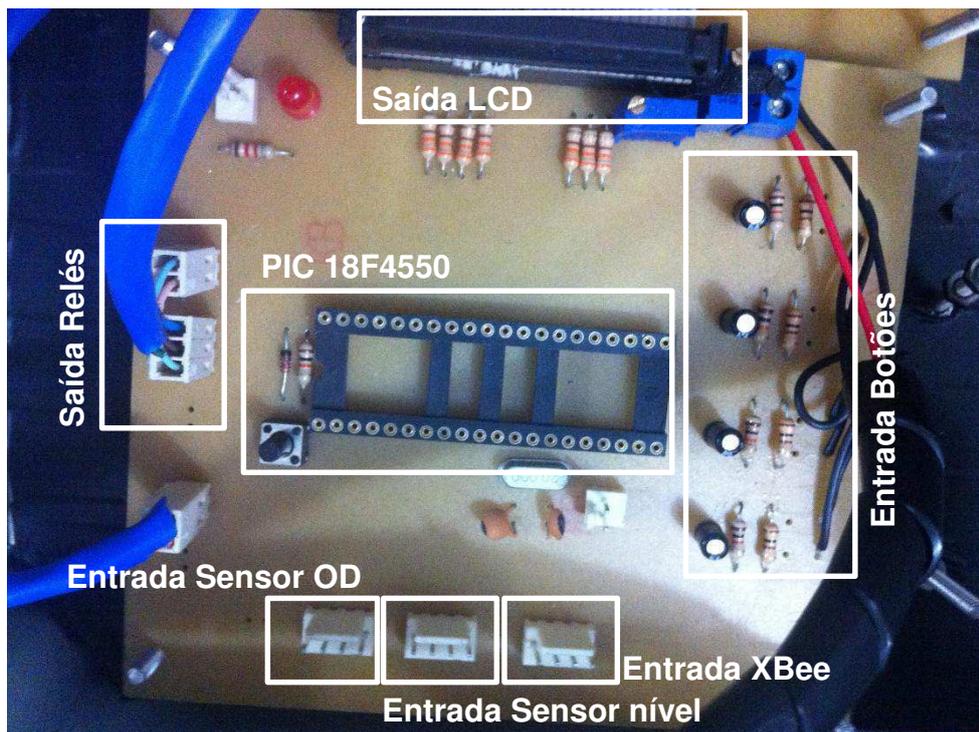


Figura 60 Placa central do sistema embarcado.

O diagrama elétrico da placa central é apresentado no Apêndice E.

- **Placa de Acionamento**

Foi desenvolvida uma placa para acionamento, a partir dos pinos do PIC, das eletrobombas e válvulas. Na Figura 61 é apresentada a placa de acionamento.

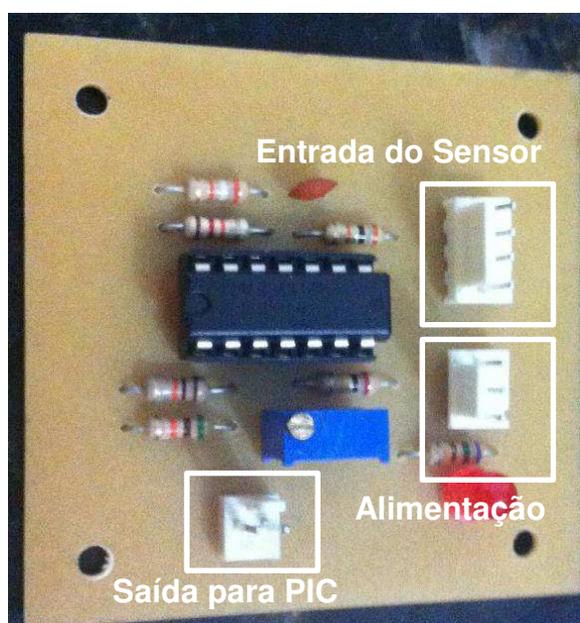


Figura 61 Placa de acionamento dos relés.

O diagrama elétrico da placa de acionamento é apresentado no Apêndice F.

- **Placa de Condicionamento de Sensores**

Foi construída a placa de condicionamento dos sinais dos sensores de OD e temperatura visando o envio destas informações para o microcontrolador. Devido a utilização de um amplificador operacional foi necessária a utilização de uma alimentação simétrica de 5 e -5 V. Na Figura 62 é apresentada a placa de condicionamento dos sinais dos sensores.



**Figura 62** Placa de condicionamento do sensor OD e temperatura.

O diagrama elétrico da placa de condicionamento dos sensores de temperatura e OD é apresentado no Apêndice G.

Após a construção e testes de funcionamento de cada placa separadamente do sistema embarcado foram agrupadas e instaladas no reator de bateladas sequenciais para o tratamento de águas residuárias. Após a instalação foi constatado a necessidade de pequenos ajustes no *software* do microcontrolador

para melhor adequação do usuário. Na Figura 63 é apresentado o sistema embarcado construído para o tratamento de esgoto.



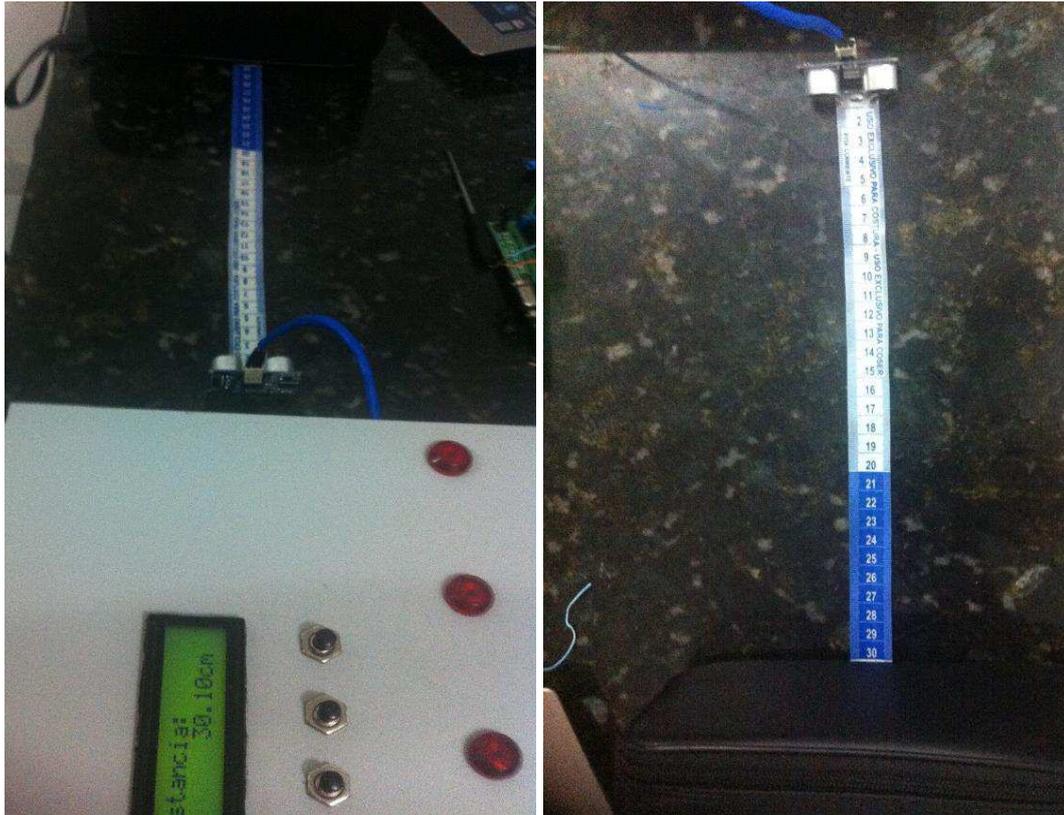
Figura 63 Sistema embarcado construído.

Para validar a temperatura aferida pelo sensor foi utilizado o instrumento medidor de temperatura que utiliza termopares como sensor, TD-990. Percebeu-se que o condicionamento escolhido foi adequado, porém o *software* desenvolvido está limitado a mostrar medições de valores de temperaturas exatas e não valores decimas. Por isso, a medição apresenta um erro entre  $\pm 0,5$  °C, pois utiliza-se no *software* os princípios de arredondamento. Na Figura 64 é apresentado a comparação entre o instrumento e o valor do sensor de temperatura utilizado.



Figura 64 Validação do sensor de temperatura.

Para validar o valor aferido pelo sensor de nível foi realizado um experimento comparando com os valores de distâncias obtidos por uma trena, em centímetros. Constatou-se que os valores aferidos do sensor foram adequados, apenas apresentando um erro entre  $\pm 3$  cm. Na Figura 65 é apresentada a comparação do valor do sensor de nível com o valor real da trena.



**Figura 65 Verificação do sensor de nível.**

A verificação dos valores obtidos do condicionamento do sensor OD foi comparado com a medição do sensor OD no equipamento YSI model 58. Neste experimento foi constatado que os valores obtidos pelo condicionamento do sensor OD estão bem próximos dos valores medidos pelo equipamento, apresentando um erro de  $\pm 0,2 \text{ mg/l}$ . Na Figura 66 é apresentado a comparação dos valores do sensor OD condicionados e dos valores do sensor OD no equipamento YSI model 58.



Figura 66 Validação do sensor de OD.

### 4.3. Análise química do esgoto

O desempenho do sistema RBS utilizado nesta pesquisa teve sua eficiência avaliada por meio da remoção de material orgânico, obtido por meio da análise de DQO (Demanda Química de Oxigênio), fração comumente analisada quando se operam sistemas de tratamento biológico de esgotos. Foram realizados 3 testes de DQO, sendo um por semana, e os dados obtidos foram DQO afluente média de 724 mg/L e o efluente do RBS apresentou concentração média de DQO de 56 mg/L o que representa em termos percentuais uma eficiência de remoção superior a 92%.

Este elevado percentual de remoção indica uma estabilidade do sistema operado, e um indicativo de que sistemas de lodo ativado quando bem operados e monitorados alcançam percentuais de remoção satisfatórios. A boa remoção de material orgânico indica ainda que este sistema provavelmente também apresenta desempenho satisfatório para a remoção dos nutrientes nitrogênio e fósforo, os grandes vilões do tratamento de esgotos domésticos.

## 5. Conclusões e Trabalhos Futuros

Nesta dissertação, foi apresentado o desenvolvimento de um sistema de automação e monitoramento de reatores de bateladas sequenciais para tratamento de águas residuárias. O sistema tem a capacidade de adquirir dados dos sensores, determinar o período de cada fase de tratamento, monitorar o processo de tratamento no reator e proporcionar para o usuário um *software* capaz de monitorar o sistema a distância. O desenvolvimento do sistema foi dividido em duas etapas: a construção do sistema embarcado e a codificação do sistema computacional.

O sistema computacional foi codificado em linguagem JAVA com a finalidade de armazenar e apresentar para os usuários os dados dos reatores em forma de gráficos, quadros e relatórios. Estes relatórios são gerados no formato PDF, contribuindo para posteriores análises dos usuários. Todos os dados do tratamento são armazenados em um banco de dados local, gerenciados pelo PostgreSQL.

Os dados recebidos no *software* são enviados pelo sistema embarcado via RF por meio do dispositivo XBee, baseado no protocolo ZigBee. Também, por meio do sistema computacional, é permitido aos usuários alterações nas variáveis do sistema embarcado, a distância, trazendo para os usuários maior comodidade nas análises do tratamento de águas residuárias.

Foram realizados testes no sistema computacional com o propósito de averiguar a confiabilidade do sistema computacional, nestes testes percebeu-se que o *software* apresentou excelentes resultados, sem erros que pudessem afetar o desenvolvimento da pesquisa.

O sistema embarcado foi desenvolvido visando automatizar os reatores de bateladas sequencias. Para esta automatização foi utilizado o microcontrolador PIC 18F4550 que gerência todo o processo de obtenção de dados dos sensores e os tempos das fases de tratamento, como também o envio destes dados serialmente para o XBee, o qual retransmite as informações para o sistema computacional.

Para a obtenção do sistema embarcado foram projetados e implementados circuitos de condicionamento e processamento, tais como: circuito de regulação de tensão para fornecer alimentação à todos os circuitos do sistema; circuito de condicionamento e processamento dos sinais obtidos nos sensores de OD, de nível e de temperatura; circuito de condicionamento para o dispositivo XBee e acionamento dos relés.

Os testes de validação do sistema foram realizados no laboratório de tratamento anaeróbio da UEPB, utilizando um reator em escala piloto para o tratamento do esgoto. A partir dos resultados obtidos, foi observado que o sistema funcionou satisfatoriamente com um percentual de tratamento superior de 92% a cada ciclo. A maior deficiência encontrada do sistema reside nas interferências causadas pelo acionamento das eletrobombas.

A partir do sistema desenvolvido, trabalhos futuros podem ser realizados para a melhoria das informações do tratamento de águas residuárias.

Para a continuação deste projeto, propõe-se:

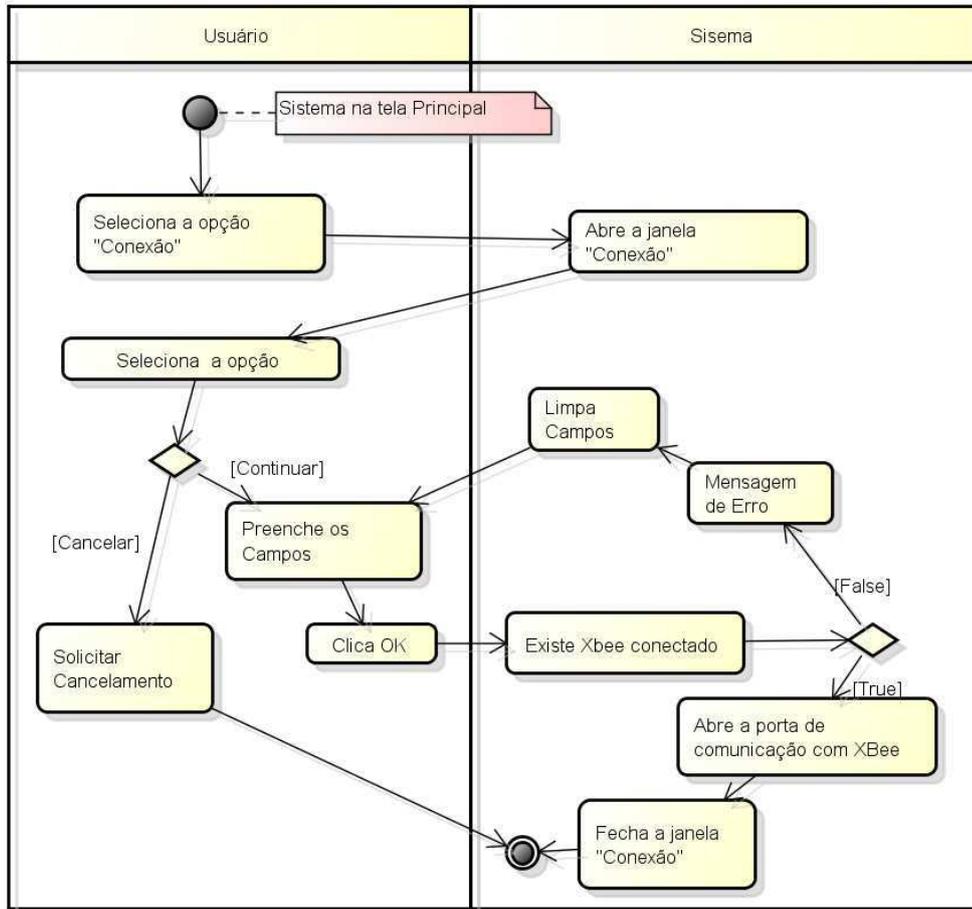
- Incorporar ao sistema proposto um *software* baseado em linguagem de programação web. Trazendo para o sistema maior abrangência no monitoramento do tratamento.
- Incorporar o monitoramento do sistema a um aplicativo para celulares. Aumentando ainda mais a usabilidade do sistema de tratamento.
- Desenvolver um refinamento no condicionamento do sinal dos sensores, de forma a minimizar ruídos.
- Integrar ao sistema um sensor de pH para fornecer subsídios adicionais para a caracterização do comportamento das bactérias no tratamento do esgoto.

## Referências

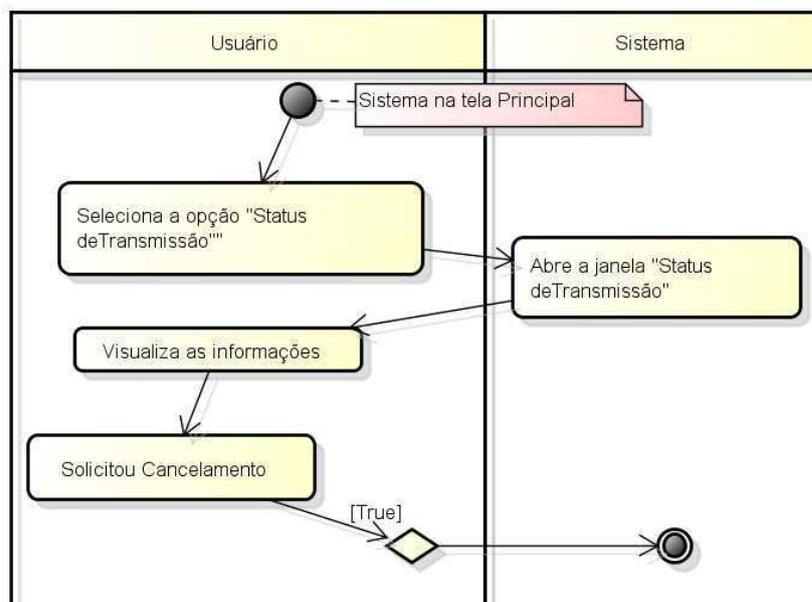
- [1] Instituto Brasileiro de Geografia e Estatística, *Atlas de saneamento 2011*. Rio de Janeiro: Instituto Brasileiro de Geografia e Estatística, 2011.
- [2] F. C. Thans, “Controle Operacional de Reator em Bateladas Seqüenciais (RBS): Ajustes na Concentração de Oxigênio Dissolvido Visando a Remoção de Nutrientes,” Dissertação, Universidade Federal de Santa Catarina, Florianópolis, 2008.
- [3] D. P. Brasil, “Remoção Biológica de Matéria Orgânica e Nutrientes de Esgotos Sanitários Utilizando reatores em Bateladas Seqüenciais,” Dissertação, Campina Grande, 2010.
- [4] V. V. Lira, P. R. Barros, J. S. da Rocha Neto, and A. C. van Haandel, “Automation of an anaerobic-aerobic wastewater treatment process,” in *Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference, 2001. IMTC 2001*, vol. 2, pp. 1177–1182 vol.2, 2001.
- [5] V. V. Lira, J. S. da Rocha Neto, P. R. Barros, and A. C. van Haandel, “Automation of an anaerobic-aerobic wastewater treatment process,” *IEEE Trans. Instrum. Meas.*, vol. 52, no. 3, pp. 909–915, Jun. 2003.
- [6] J. Moreno, “Optimal time control of sequencing batch reactors for industrial wastewater treatment,” in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 1, pp. 826–827 vol.1, 1997.
- [7] Y. Xiu-ping, Q. Zhao-xia, and L. Ya-xin, “Minimum pH Condition for Anaerobic Sequencing Batch Reactor Containing Granulated Biomass,” in *The 2nd International Conference on Bioinformatics and Biomedical Engineering, 2008. ICBBE 2008*, pp. 3587–3590, 2008.
- [8] S. Wan-sheng, S. You-ju, and F. Chuan-ping, “A pilot study on biological nitrogen removal in a frequent anoxic-aerobic alternation sequencing batch reactor,” in *2011 International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 991–993, 2011.
- [9] G. A. Arima, “Otimização de reator sequencial em batelada para remoção biológica de carbono e nitrogênio com controle difuso,” Tese, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2005.
- [10] D. Ibrahim, *PIC BASIC projects: 30 projects using PIC BASIC and PIC BASIC PRO*. Amsterdam ; Boston: Elsevier, 2006.
- [11] E. C. L. do Nascimento, “Um fotômetro microcontrolado led-nir, portátil e de baixo custo para análise screening de gasolinas tipo c,” Tese, Universidade Federal da Paraíba, João Pessoa, 2008.
- [12] Microchip, “Datasheets do PIC18F2455/2550/4455/4550.” 2007.
- [13] L. A. M. Pereira, “Robô Autômato para Monitoramento de Rebanho e Medição da Forragem do Pasto,” Tese, Universidade de São Paulo, Pirassununga, 2009.
- [14] J. M. M. Villanueva, “Fusão de Dados das Técnicas de Tempo Transito Utilizando transdutores Ultra-sônicos para Medição da Velocidade do Vento,” Dissertação, Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2009.
- [15] M. A. C. F. Ferreira, “Desenvolvimento de Sensores de Oxigênio Dissolvido Utilizando Métodos Eletroquímicos e Ópticos para Monitoramento em Tempo Real da Qualidade de Água,” Tese, Universidade de São Paulo, São Paulo, 2007.
- [16] YSI Incorporated, “The Dissolved Oxygen Handbook.” 2009.
- [17] H.S. Antunes Filho, “Nitrificação Em Sistemas De Lodo Ativado,” Dissertação, Universidade Federal de Campina Grande, Campina Grande, 2009.

- [18] M. Pecht, P. Lall, G. Ballou, and N. Angelopoulos, “The Electrical Engineering Handbook.” Richard C. Dorf, 2000.
- [19] S. Y. C. Catunda, “Sistema para verificação da qualidade de afluentes de sistemas de tratamento de águas residuárias,” Dissertação, Universidade Federal da Paraíba, Campina Grande, 1996.
- [20] P. Ran, M. Sun, and Y. Zou, “ZigBee Routing Selection Strategy Based on Data Services and Energy-Balanced ZigBee Routing,” pp. 400–404, 2006.
- [21] F. Monsignore, “Sensoriamento de ambiente utilizando o padrão ZigBee,” Dissertação, Universidade de São Paulo, 2007.
- [22] Z. Alliance, “ZIGBEE SPECIFICATION.” 2008.
- [23] D. International, “XBee / XBee-PRO RF Modules.” 2014.
- [24] S. A. F. Soares, “Rede de Sensores Sem Fio Para Localização e Monitoramento de Pequenos Ruminantes,” Trabalho de Conclusão de Curso, Universidade Federal do Vale do São Francisco, Juazeiro, 2012.
- [25] R. A. RAMOS, *Treinamento Prático em UML*. Universo dos Livros Editora LTDA, 2006.
- [26] I. G. S. Brito, “Sistema Computacional para o Monitoramento da Irrigação em Viveiro de Plantas,” Dissertação, Universidade Federal do Vale do São Francisco, Juazeiro, 2012.
- [27] G. Booch, J. Rumbaugh, and I. Jacobson, *UML: guia do usuário*. Elsevier Brasil, 2006.
- [28] H. M. Deitel, *Java: como programar*. Pearson Prentice Hall, 2005.
- [29] E. Braude, *Projeto de Software: Da programação à arquitetura: Uma abordagem baseada em Java*. Bookman Companhia Ed, 2005.
- [30] F. A. Ferrari, *Crie banco de dados em MySQL*. Universo dos Livros Editora, 2012.
- [31] C. Technologies, “User’s Manual HC-SR04.” 2013.
- [32] XBEE STORE, “Dispositivo XBee.” [Online]. Available: <http://xbeestore.lojavirtualfc.com.br>. [Accessed: 20-Aug-2014].

## Apêndice A – Diagramas de Atividades



**Figura 67 Diagrama de atividades habilitar conexão com XBee.**



**Figura 68 Diagrama de atividades status de transmissão.**

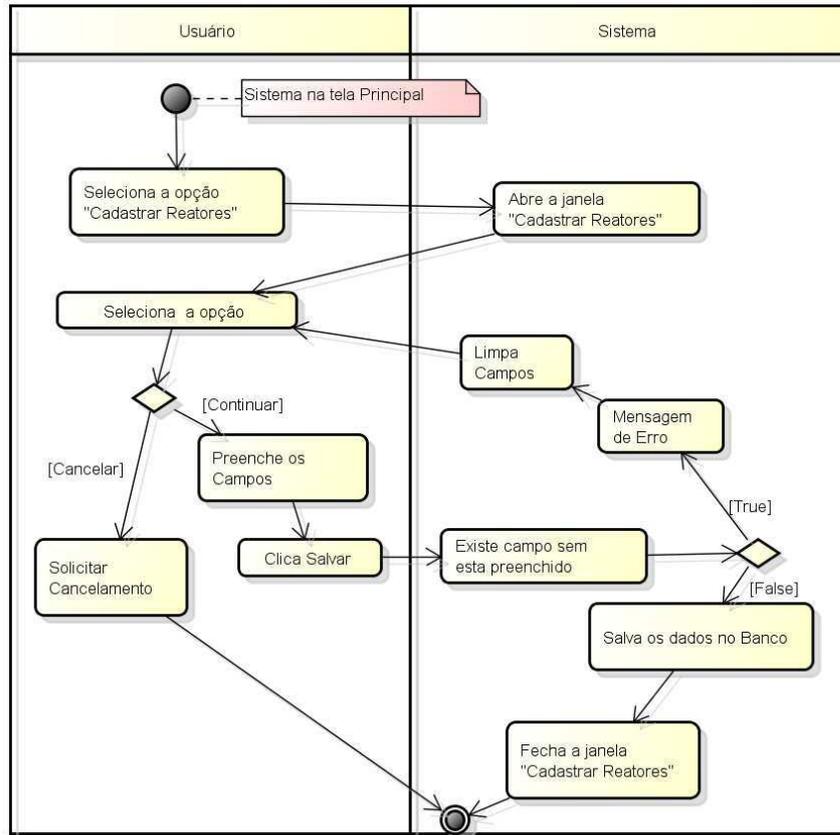


Figura 69 Diagrama de atividades cadastrar reatores.

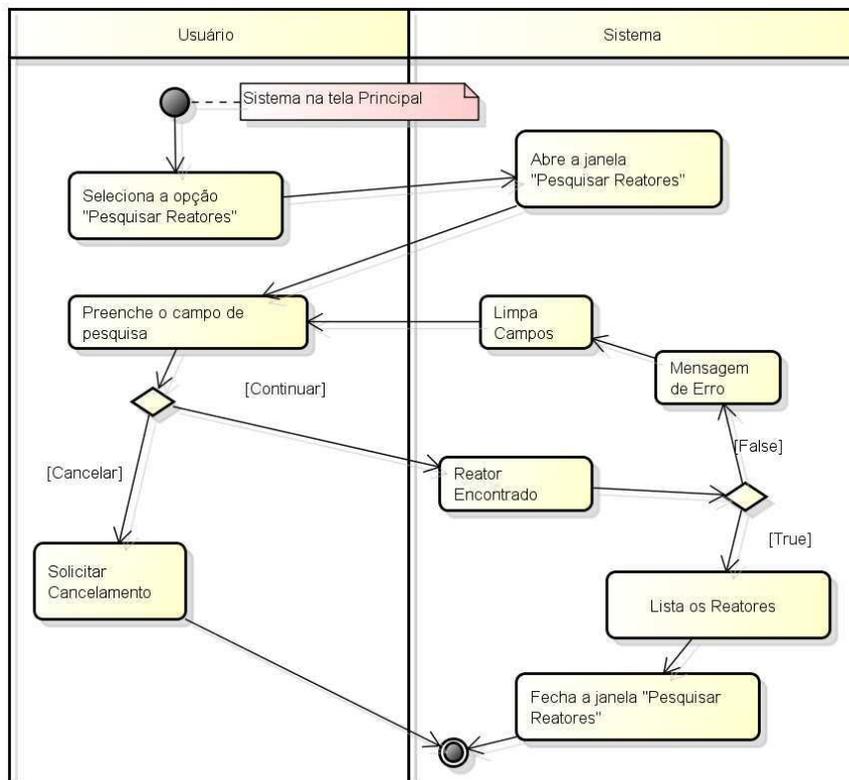


Figura 70 Diagrama de atividades Pesquisar Reatores.

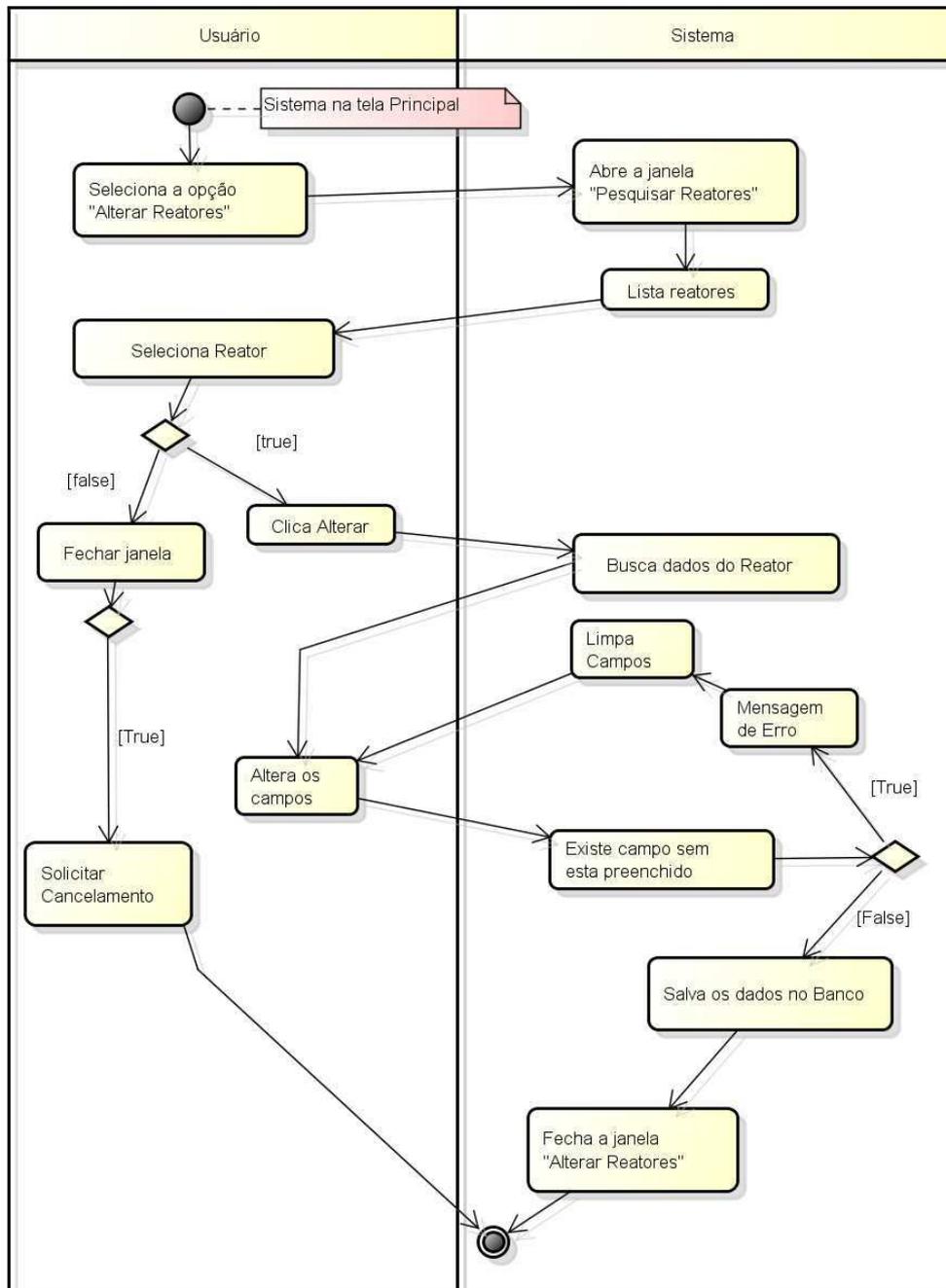


Figura 71 Diagrama de atividades alterar reatores.

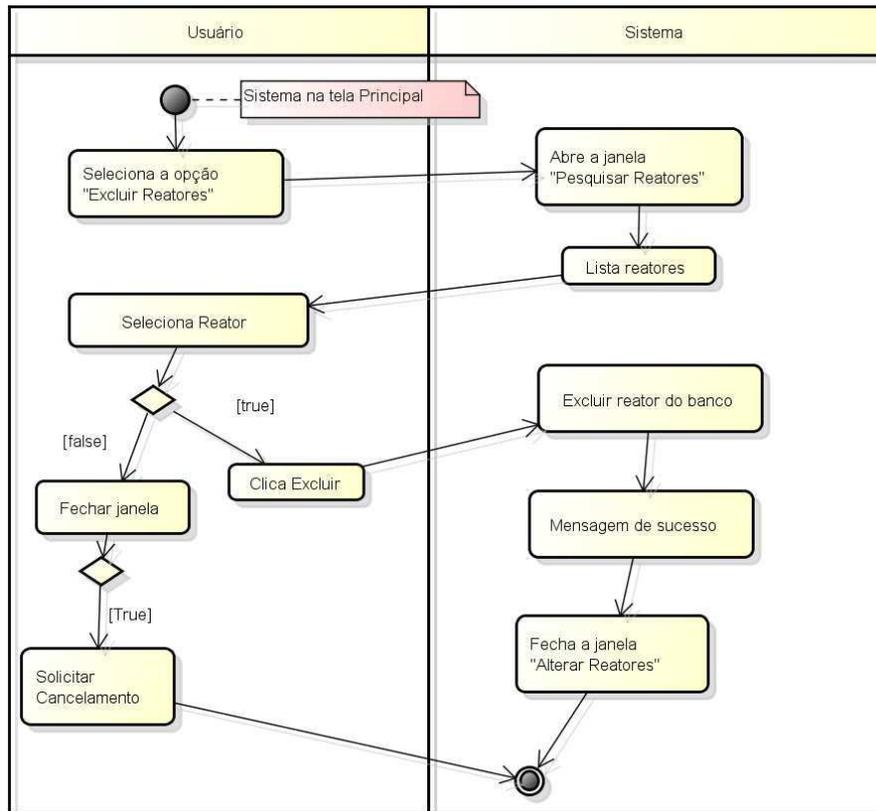


Figura 72 Diagrama de atividades excluir reatores.

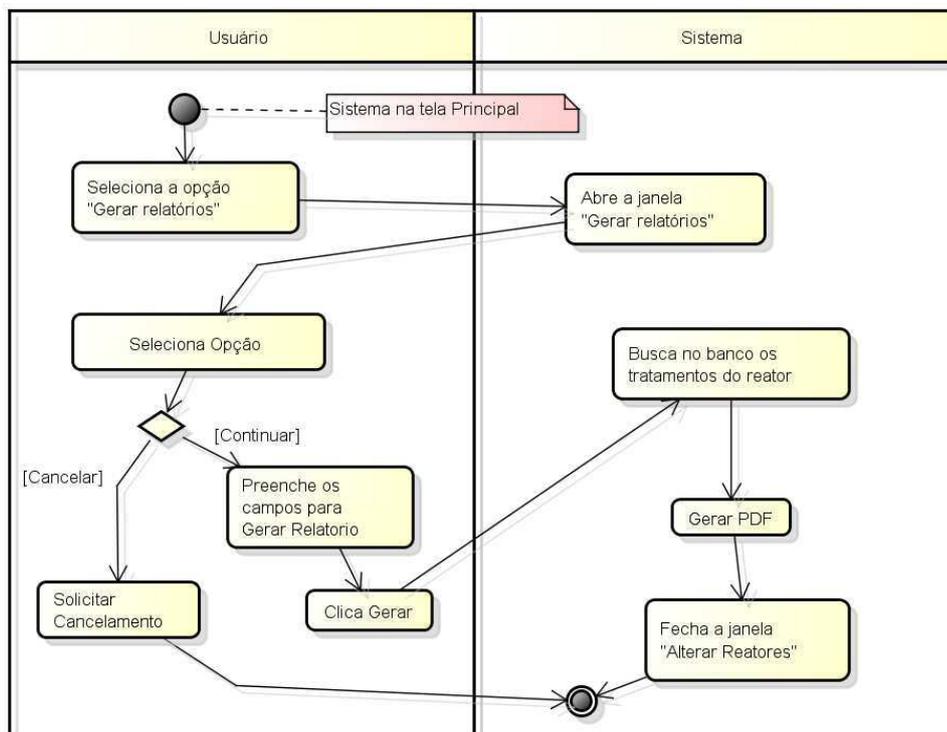


Figura 73 Diagrama de atividades Gerar PDF.

## Apêndice B– Diagramas de Classes

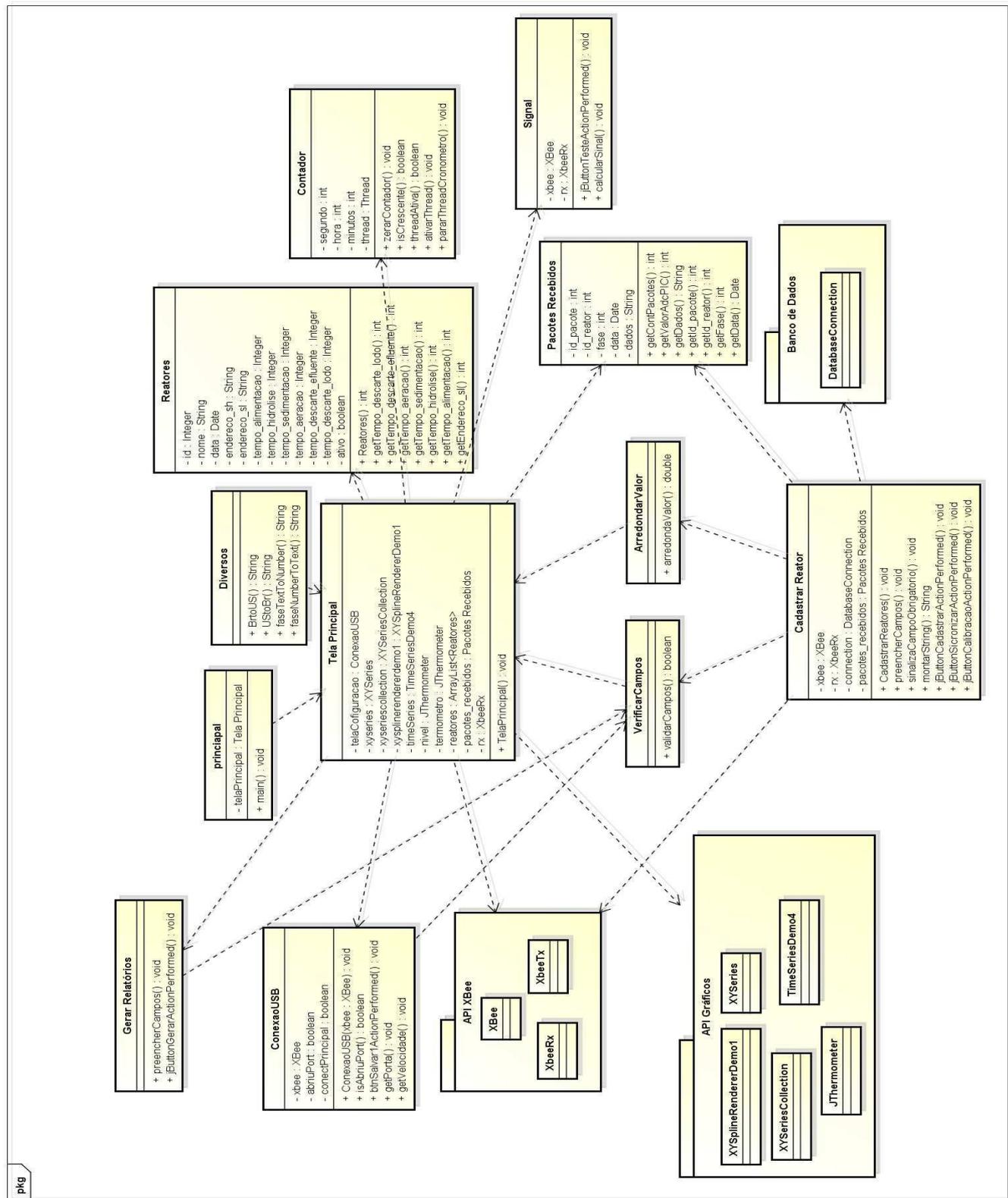


Figura 74 Diagrama de Classes do sistema computacional.

## Apêndice C– Diagrama elétrico da fonte de alimentação

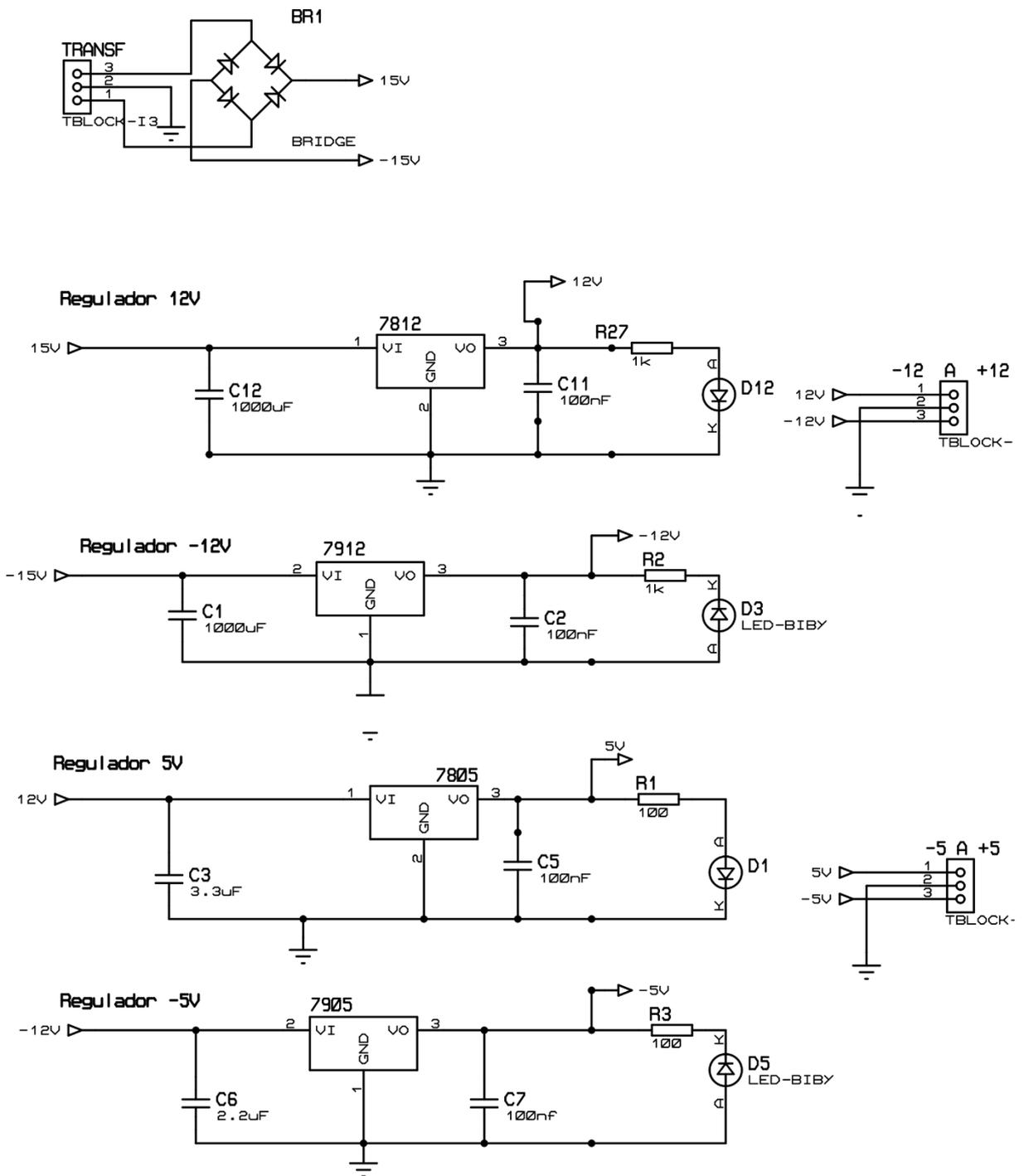


Figura 75 Diagrama elétrico da placa de alimentação parte 1.

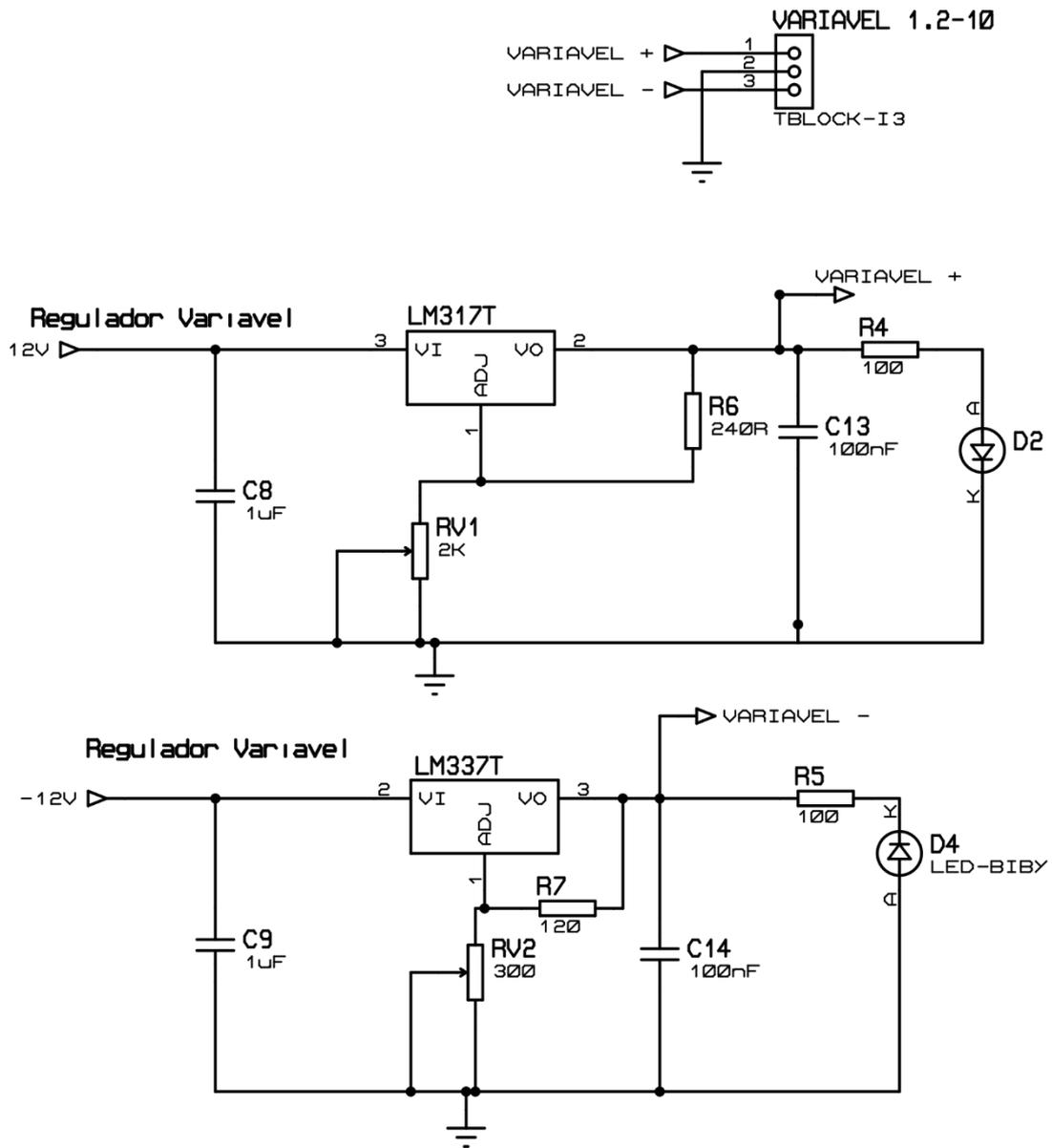


Figura 76 Diagrama elétrico da placa de alimentação parte 2.

## Apêndice D – Diagrama elétrico da placa de comunicação com XBee.

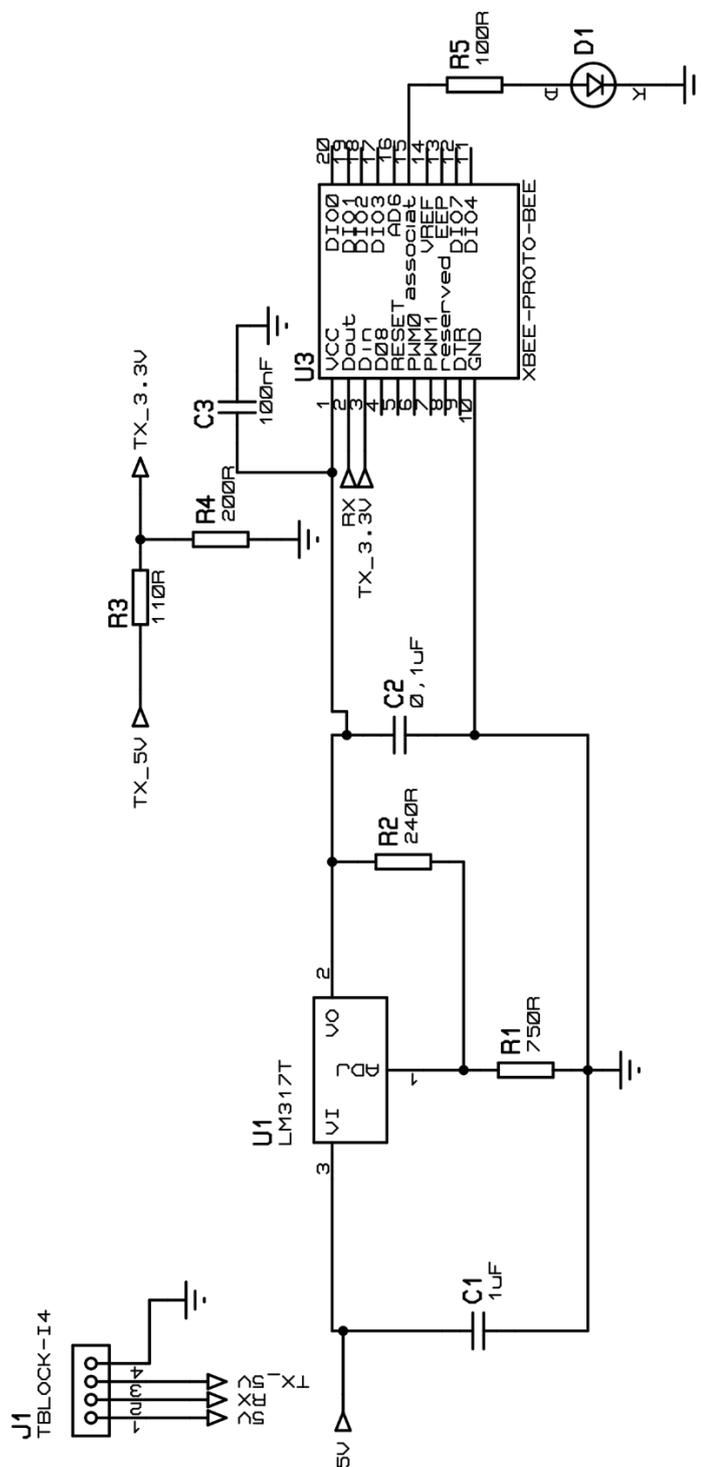


Figura 77 Diagrama elétrico do condicionamento do XBee.

## Apêndice E – Diagrama elétrico da placa central do sistema embarcado.

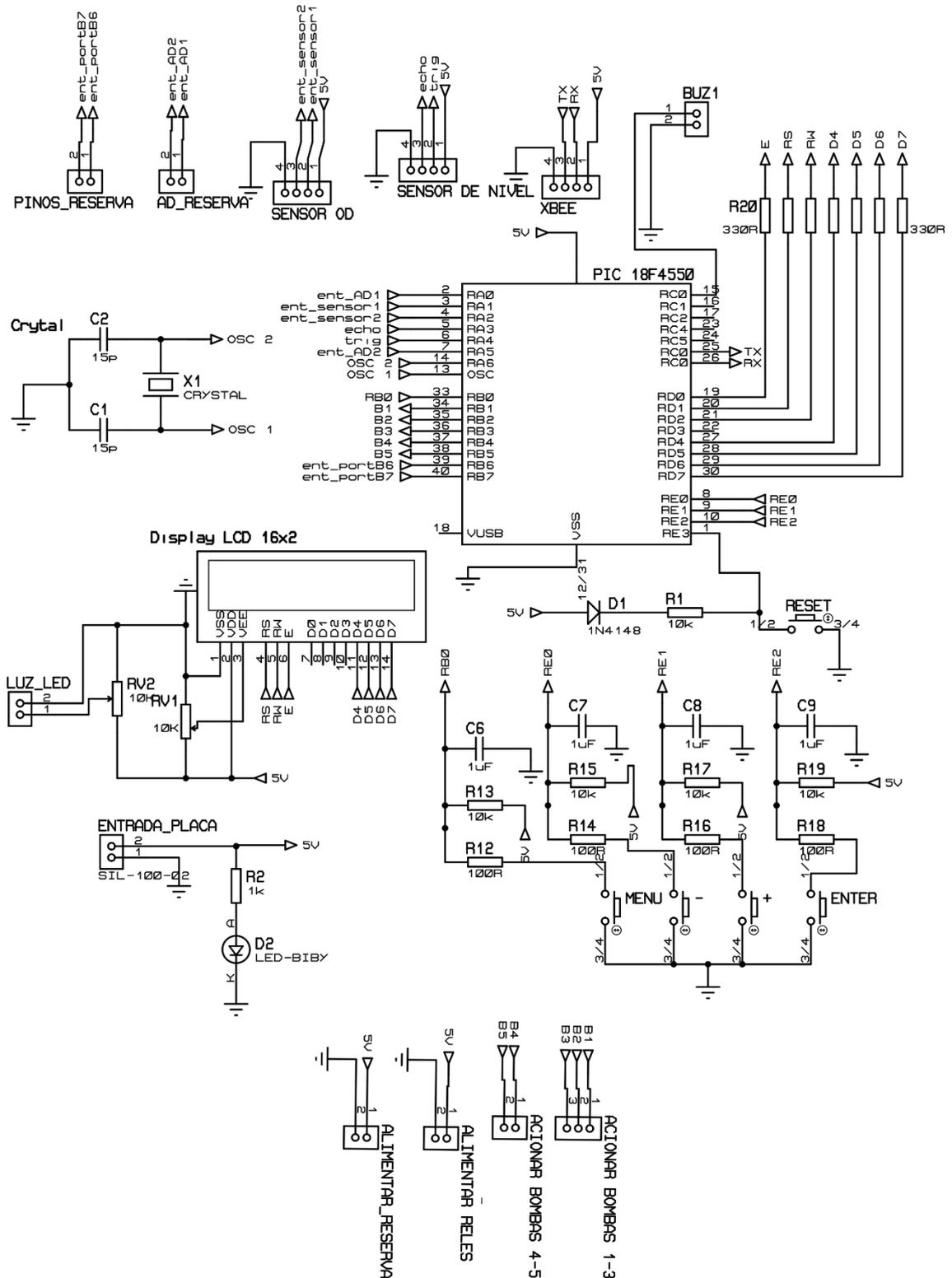


Figura 78 Diagrama da placa Central do sistema embarcado.

## Apêndice F – Diagrama elétrico da placa de acionamento dos relés.

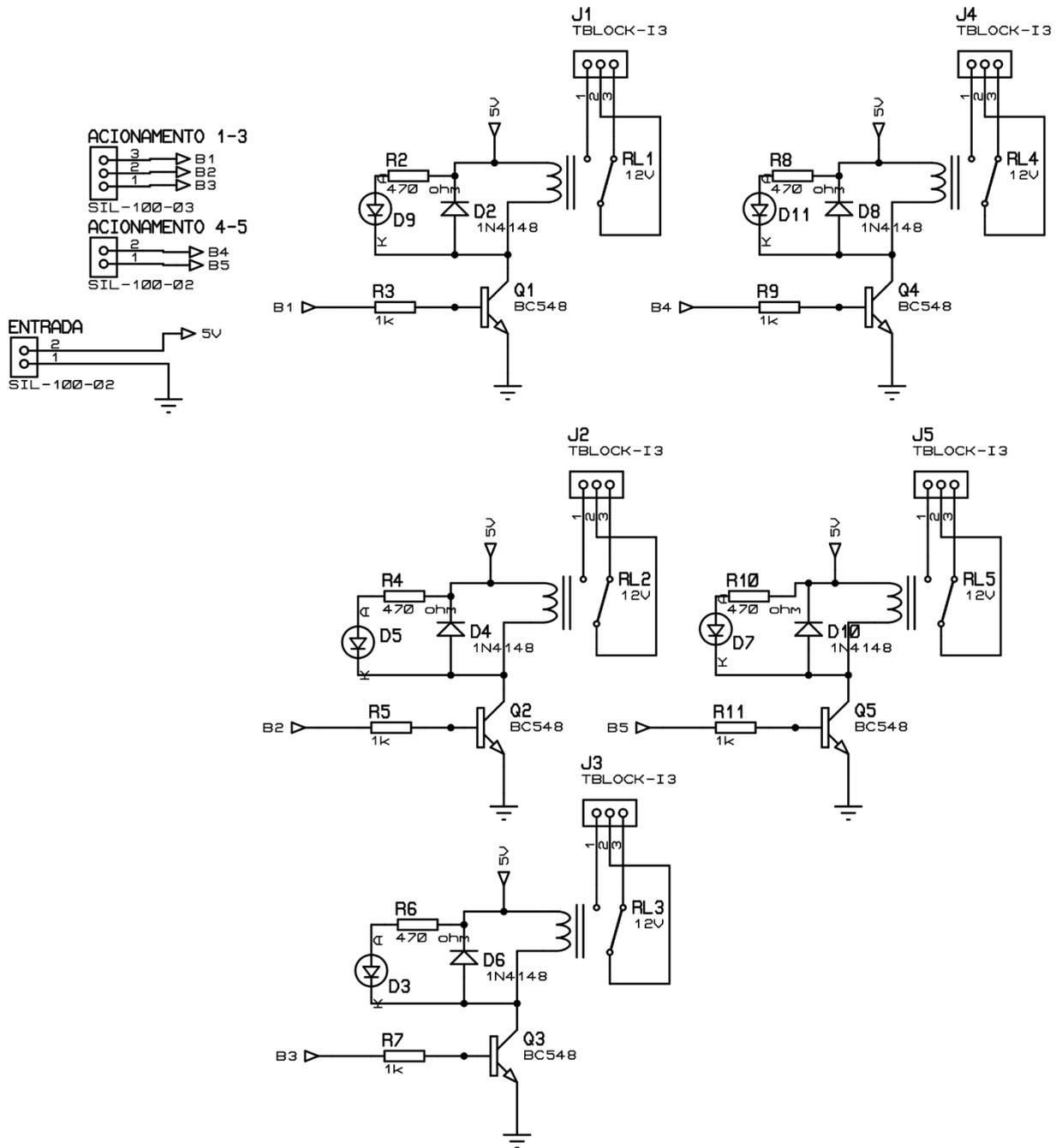


Figura 79 Diagrama da placa de acionamento.

## Apêndice G – Diagrama elétrico da placa de condicionamento dos sinais dos sensores de OD e temperatura.

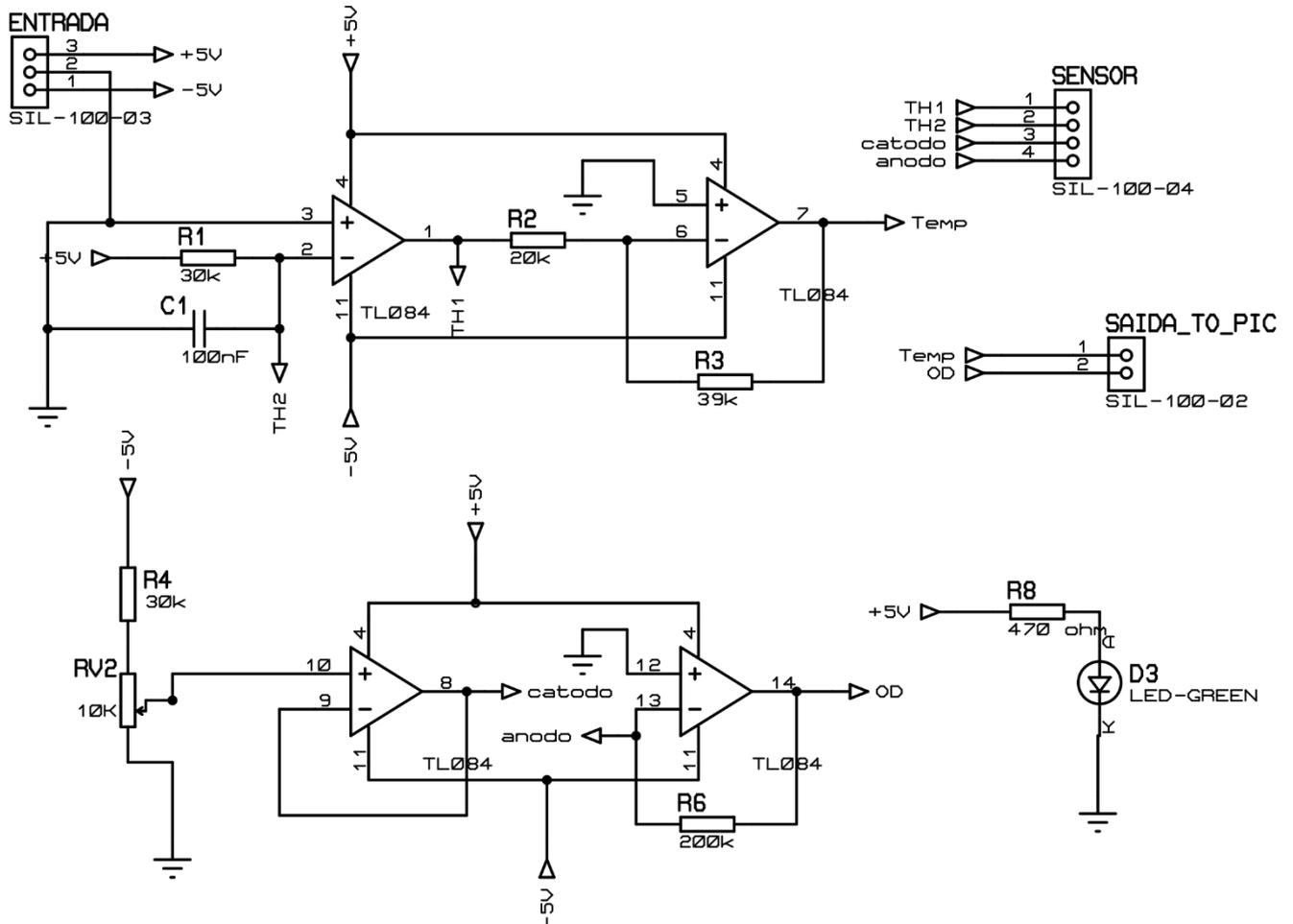


Figura 80 Diagrama da placa de condicionamento dos sensores.