



**INSTITUTO FEDERAL**

Pernambuco

Campus Garanhuns

**IFPE – INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DE PERNAMBUCO**

**CURSO DE TECNICO EM INFORMÁTICA  
MODALIDADE INTEGRADO**

**EDGAR VINICIUS CARVALHO VITAL**

**ERWESON QUIRINO DANTAS**

**IGOR SILVA DE SOUZA**

**SISTEMA EMBARCADO PARA CONTROLE E  
MONITORAMENTO DA GERAÇÃO DE BIOGÁS PRODUZIDO  
POR UM BIODIGESTOR**

GARANHUS/PE

Dezembro de 2016

EDGAR VINICIUS CARVALHO VITAL  
ERWESON QUIRINO DANTAS  
IGOR SILVA DE SOUZA

**SISTEMA EMBARCADO PARA CONTROLE E  
MONITORAMENTO DA GERAÇÃO DE BIOGÁS PRODUZIDO  
POR UM BIODIGESTOR**

Trabalho apresentado ao Instituto  
Federal de Pernambuco - Campus  
Garanhuns como pré-requisito para a  
obtenção de Certificado de Conclusão  
do Curso de Técnico em Informática

Orientador: David Alain do Nascimento

GARANHUNS / PE  
Dezembro de 2016

# Resumo

Este projeto foi desenvolvido no Instituto Federal de Pernambuco - Campus Garanhuns em parceria com programa FABIN (Fabrica de Ideias Inovadoras) do Instituto Federal de Brasília (IFB), cujo tema foi Energia Alternativa para o Brasil, após o desenvolvimento de um Biodigestor para produção de Biogás foi implementada uma solução para seu controle e monitoramento através de um sistema embarcado.

O uso de alguns sensores e componentes interligados à uma plataforma de sistema embarcado *open hardware*, responsável por monitorar a produção do biogás, se mostrou ser uma solução de baixo custo e muito eficaz, trazendo mais uma proposta de energia alternativa para o Brasil.

Palavras-chave: Biodigestor, energia limpa, renovável, microcontrolador.

*Este projeto é dedicado  
especialmente, aos nossos pais,  
professores e contribuintes com  
nossa formação e ao nosso  
orientador, David Alain do  
Nascimento e a todos aqueles  
que que nos incentivaram  
durante todo o processo de  
elaboração deste projeto.*

# Sumário

Resumo .....	1
Lista de Tabelas .....	6
Capítulo 1 – Introdução.....	6
1.1 Apresentação do Problema .....	6
1.2 Motivação .....	7
1.3 Objetivos do Trabalho .....	8
1.3.1 Objetivos do Estudo .....	9
1.3.3 Descrição da Instrumentação .....	11
1.4 Resultados Esperados .....	11
Capítulo 2 – Referencial Tecnológico .....	12
2.1 Arduino .....	12
2.1.1 Motivos para Escolha do Microcontrolador .....	12
2.1.2 O Microcontrolador Arduino UNO .....	13
2.3 Demais Componentes Físicos.....	13
2.3.1 Sensor de Temperatura DS18B20 .....	13
2.3.2 Sensor de Temperatura LM35.....	14
2.3.3 Módulo RTC ( <i>Real Time Clock</i> ) DS1307 .....	15
2.3.4 Módulo Cartão micro SD .....	16
2.3.5 Sensor Fluxo de Água ½ YF-S201 .....	18
2.3.6 Válvula Solenoide .....	18
2.3.7 Módulo Relé .....	20
2.3.8 Sensor Ultrassônico HC-SR04 .....	21
Bomba .....	22
2.4 Tabela de Pinos.....	22
2.5 Ambiente de Desenvolvimento para o Arduino .....	23
2.5.1 Estrutura do <i>Sketch</i> .....	25

3.0 Sensor de Temperatura DS18B20 .....	25
3.1 Sensor de Temperatura LM35.....	25
3.1 Real Time Clock RTC DS1307 .....	26
3.2 Módulo Cartão Micro SD .....	26
3.3 Sensor Fluxo de Água YF-S201 .....	26
3.4 Válvula Solenoide.....	27
3.5 Módulo Relé .....	27
3.6 Sensor Ultrassônico .....	27
3.7 Bomba .....	27
Capítulo 4 – Aplicação da Solução com Resultados .....	28
Capítulo 5 – Considerações finais .....	28
5.1 Conclusões.....	28
5.2 Sugestões para Trabalhos Futuros .....	29
6.0 Bibliografia .....	29
7.0 Programa Inserido no Arduino .....	30

# Lista de Figuras

Figura 1 - Topologia do Projeto .....	9
Figura 2 - Arduino Uno .....	13
Figura 3 - Sensor de Temperatura DS18B20 .....	14
Figura 4 - Sensor LM35.....	15
Figura 5- Módulo RTC DS1307 .....	16
Figura 6 - Módulo Cartão Micro SD .....	17
Figura 7 - Sensor Fluxo de Água ½ YF-S201 .....	18
Figura 8 - Válvula Solenoide .....	19
Figura 9 - Módulo Relé.....	20
Figura 10 - Sensor Ultrassônico HC-SR04.....	21
Figura 11 - Ambiente de Desenvolvimento Arduino 1.6.11.....	24
Figura 12 - Legenda de Ícones do Arduino 1.6.11 .....	24
Figura 13 - Ciclo de Desenvolvimento para Plataforma Arduino	Erro! Indicador não definido.
Figura 14 - Estrutura do Sketch .....	25

## Lista de Tabelas

Tabela 1- Pinos.....	23
----------------------	----

# Capítulo 1 – Introdução

## 1.1 Apresentação do Problema

O Brasil é um país que aposta nos biodigestores como fonte de energia renovável e limpa, mas por ser uma tecnologia relativamente nova, necessita de um estudo aprofundado para esclarecer as dúvidas encontradas na produção dessa energia, a ideia é atender a expectativa de ser um sistema de baixo custo para obter melhores resultados com a proposta de maximizar os recursos disponíveis.

Surgiu na década de 60 a plataforma Arduino proporcionando funcionalidade e facilidade para criar projetos, executa funções específicas para as quais ele foi programado, na atualidade é comum ver essas aplicações interagindo com o meio ambiente através de equipamentos eletrônicos e



periféricos. No Biodigestor, os dados manipulados servem de parâmetro, não só para garantir a integridade do sistema, mas também melhorar o desempenho através de testes realizados em diferentes protótipos para obter o melhor resultado.

## 1.2 Motivação

O mundo atual formula diversas opções para solucionar o problema energético que se adapte à nossa realidade, o Brasil vem investindo em biodigestores utilizando sua maior característica climática baseada na maior fonte de energia renovável e limpa que é o sol. A partir da criação de um biodigestor muitas incertezas sobre suas especificações e potencial surgiram, por ser uma tecnologia nova necessita de um estudo mais aprofundado, baseado no baixo custo e bom desempenho, uma solução encontrada para suprir a necessidade de captura desses dados foi a utilização do sistema embarcado nessa aplicação, conseguimos através de um microcontrolador controlar e monitorar o comportamento do biodigestor dando margens para melhorar o desempenho a partir de algumas variáveis como pressão, temperatura e vazão do sistema de aquecimento.

O curso de Informática possui disciplinas que tratam do assunto como a linguagem de programação, Tópicos Avançados que faz com que o aluno se sinta à vontade para desenvolver e buscar inovações na área. Dentre as inovações tecnológicas mais recentes nos últimos 10 anos, é possível afirmar que a plataforma de computação física microcontrolada Arduino certamente esteja entre as mais revolucionárias, devido ao seu alto grau de facilidade de aprendizado, implementação e disponibilidade de recursos, que permite que leigos e profissionais desenvolvam os mais variados dispositivos de automação.

O biodigestor tem a capacidade de substituir o gás GLP (gás de cozinha) a base de matéria orgânica de excremento de animais e resto de comida, essa mistura gasosa de dióxido de carbono e metano através da fermentação das

bactérias tem a capacidade de produzir o biogás utilizando apenas resíduos agrícolas trazendo muitas vantagens as famílias.

## 1.3 Objetivos do Trabalho

Através do programa FABIN (Fabrica de Ideias Inovadoras) foi desenvolvido um biodigestor. O objetivo deste trabalho é esclarecer algumas lacunas no estudo dessa energia renovável tal como a quantidade de gás que está sendo produzida e sua vazão, o sistema desenvolvido monitora e controla a produção do biogás ao exportar os dados para análise posterior através de um cartão de memória micro SD, o qual podem ser feitas análises gráficas e estatística sobre os dados do período de monitoramento. O sistema conta com um biodigestor, sensores de temperatura, fluxo, além de módulos como RTC DS1307, micro cartão SD, relé e válvulas de solenoide todos conectados eletronicamente a uma plataforma microcontrolada abastecida por um painel solar, coletando os dados via cartão micro SD.

Afim de ilustrar o funcionamento do sistema, foi elaborado um diagrama conforme é exibido na *Figura 1*. Através dele é possível compreender o funcionamento do aquecimento da água, o Arduino interpreta os comandos a ele enviado com auxílio de sensores para executar a ação demandada.

## PLANTA: PROJETO FABIN

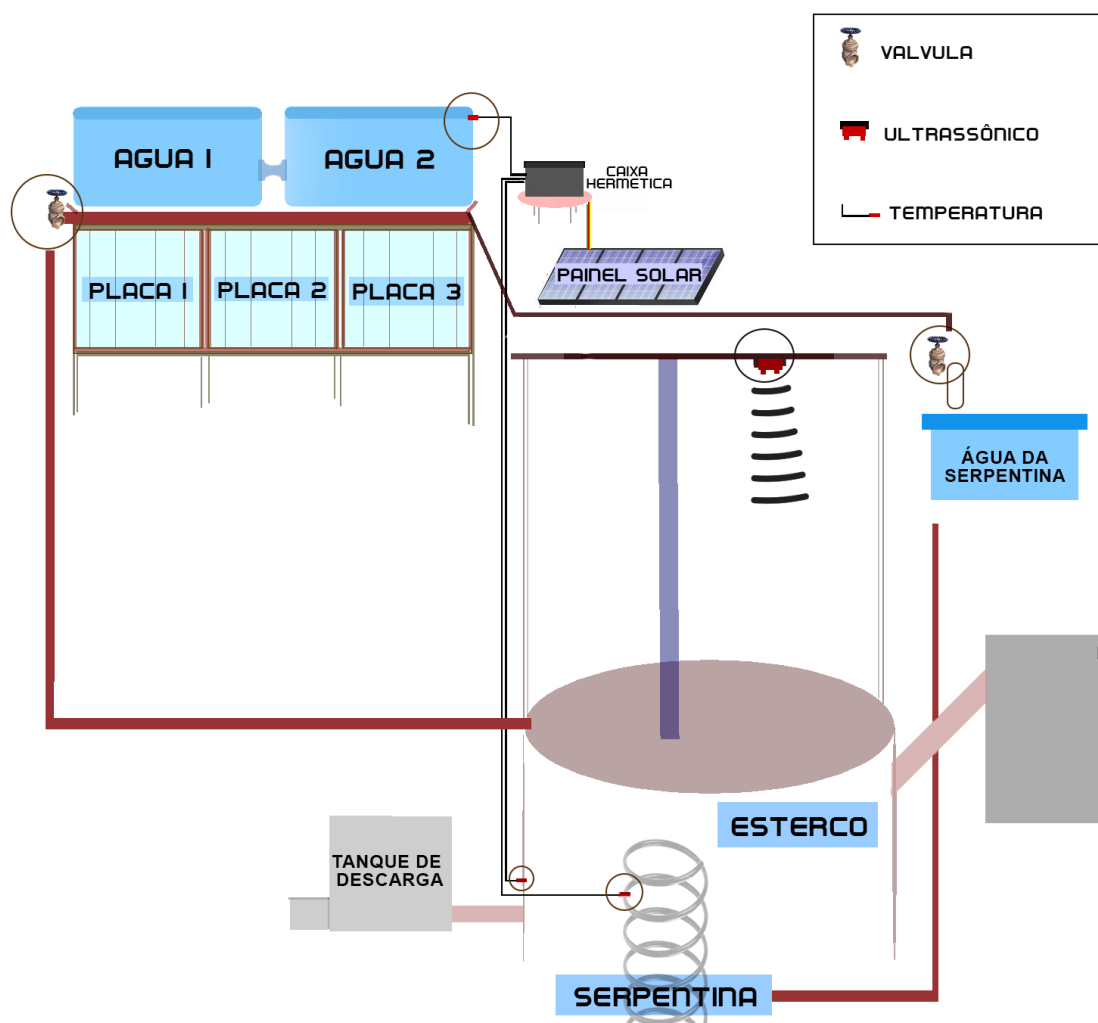


Figura 1 - Topologia do Projeto

### 1.3.1 Objetivos do Estudo

Monitorar a temperatura e controlar o sistema de produção de biogás à partir de sensores conectados a plataforma Arduino e da linguagem de programação C++ com o propósito de controlar os componentes interligados a ele, ao retirar os dados no formato de arquivo CSV (*Comma Separated Values*), os dados são escritos em arquivo na seguinte ordem: "dia da semana, data, hora, temperatura da água, temperatura do esterco, temperatura da serpentina, temperatura ambiente, vazão do biogás", e dessa maneira os dados podem ser

facilmente visualizados nos gráficos com mais detalhes, para chegar a uma conclusão se a região e o clima favorecem ou não na implantação do biodigestor na residência.

A análise financeira parte do princípio que um botijão de gás GLP custa em média R\$45.00 e uma família gasta o esse produto em um mês de uso uma casa pode economizar não apenas esse valor mensal, mas também comercializar o seu próprio fertilizante natural ou usar na sua própria plantação e ainda contribuir com o meio ambiente, claro e tudo isso só é possível se as características climáticas da região for propícia, este sistema disponibiliza dados que aprovam a veracidade do uso do biodigestor na região além de desenvolvimento de novos protótipos buscando sempre maior eficiência.

### **1.3.2 Questão**

Além da substituição do gás GLP, o biodigestor pode ser utilizado para saneamento rural e é possível reaproveitar o biofertilizante que sai do tanque de descarga, rico em nutrientes e com menor qualidade de gás para adubar a lavoura do produtor rural. O gás que está sendo utilizado não necessita de uma purificação, e inclusive as famílias que possuem o biodigestor em suas residências têm se mostrado satisfeitas com a mudança, pois a chama é mais limpa não deixando resíduos de fuligem nas panelas.

Mas como toda nova tecnologia as dúvidas são inevitáveis e até mesmo alguns produtores biodigestor não tem uma precisão dos dados do biodigestor e tem questões em aberto como: quanto de biogás está sendo produzido pelo biodigestor? Qual é a melhor temperatura para a fermentação? A produção de biogás é mais eficiente utilizando água aquecida ou fria? O quanto é melhor?

O sistema desenvolvido ajuda a esclarecer tais dúvidas. Com análises dos dados coletados, a plataforma Arduino nos proporciona realizar o monitoramento e controle da produção de gás no biodigestor, são usados sensores e periféricos integrados a este à plataforma Arduino, afim de aprovar numericamente a eficiência desta aplicação simples, porém bastante eficiente e que pode

contribuir com a economia nas residências e contribuir com diminuição da poluição.

### **1.3.3 Descrição da Instrumentação**

Como instrumentos deste experimento são utilizados:

- Placa Arduino Uno;
- Sensor de temperatura DS18B20;
- Módulo RTC DS1307;
- Módulo cartão micro SD;
- Sensor fluxo de água de conexão ½;
- Válvula de solenoide;
- Módulo relé;
- Sensor ultrassônico;
- Bomba de para-brisa;

## **1.4 Resultados Esperados**

Para este projeto construiu-se um circuito para integração dos componentes eletrônicos a fim realizar a receptação e armazenamento de variáveis enviadas pelos sensores durante o processo de fermentação anaeróbica e além de utilizar duas válvulas para liberar água aquecida para o biodigestor substituindo a água fria, e com o auxílio de uma bomba transferir água fria do recipiente de saída para o recipiente do sistema de aquecimento, afim de aumentar a produtividade do biodigestor, e a partir da análise de dados é esperada a confirmação de que o sistema de aquecimento de água acarretará no aumento da produção de biogás.

# Capítulo 2 – Referencial Tecnológico

## 2.1 Arduino

O Arduino é uma plataforma física de código aberto e ter benefícios como custo benefício, facilidade de uso, baseado em uma simples placa com microcontrolador e uma IDE prática para desenvolver os códigos para a plataforma em uma linguagem baseada em C/C++. O Arduino é responsável por capturar as informações do meio através de uma série de sensores (LM35, DS18B20, Fluxo de água) além dos módulos de micro SD, RTC e DS1307, estes monitoram a temperatura do esterco, água e estipula a produção de gases produzidas no biodigestor, e salvam no módulo de cartão micro SD com especificações de data e hora para auxiliar o monitoramento com o máximo de detalhes possíveis. O Arduino é um *hardware* de código aberto além de ter um baixo custo, o número de bibliotecas disponíveis facilita muito no desenvolvimento.

### 2.1.1 Motivos para Escolha do Microcontrolador

Existem diversas plataformas microcontroladas disponíveis no mercado como, por exemplo ARM mbed, TI LaunchPad, Arduino, etc. A plataforma Arduino (ilustrada na *Figura 2*) utilizada nesse sistema por diversos fatores entre eles o custo benefício, hardware e software de código aberto e multiplataforma e ter um ambiente de programação funcional.



Figura 2 - Arduino Uno

## 2.1.2 O Microcontrolador Arduino UNO

O microcontrolador é um conjunto de circuitos integrados programável capaz de executar as ordens gravadas em sua memória, utilizou-se no biodigestor um microcontrolador Arduino Uno que é um dos principais componentes deste projeto, este é praticamente um computador em um *chip*, contém inclusive processador, memória e periféricos de entrada e saída de dados além de utilizar memória *flash* para armazenar a programação que controlará os periféricos (sensores).

## 2.3 Demais Componentes Físicos

### 2.3.1 Sensor de Temperatura DS18B20

Utilizamos o sensor de temperatura DS18B20 neste projeto como termômetro digital, além de sua precisão ele usa apenas um fio através do protocolo 1-Wire para transmitir as leituras através de uma conexão de dados, além de ser à prova d'água. Exibido na *Figura 3*.



**Figura 3 - Sensor de Temperatura DS18B20**

Escolhemos este modelo levando em conta a disponibilidade, tempo de entrega, faixa de preço e precisão, a medição da temperatura é extremamente importante para o desenvolvimento desse projeto, além da precisão seu modelo tem a ponta revestido de um material metálico resistente (aço inoxidável) impedindo a corrosão por parte do material orgânico do biodigestor.

O sensor DS18B20 possui as seguintes características técnicas e físicas:

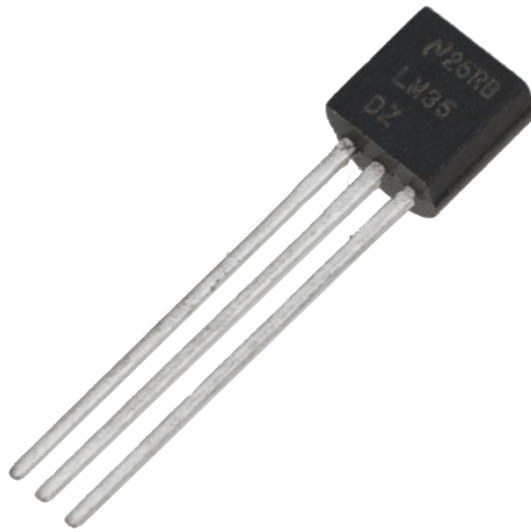
- Faixa de temperatura:  $-55^{\circ}\text{C}$  a  $+125^{\circ}\text{C}$ ;
- Tensão de alimentação: 3 ~ 5.5V DC;
- Precisão:  $\pm 0.5^{\circ}\text{C}$  entre  $-10^{\circ}\text{C}$  e  $+85^{\circ}\text{C}$ ;
- Dimensão da ponta de aço inoxidável: 6 x 50mm;
- Dimensão do cabo: 100cm;

O sensor de temperatura DS18B20 foi utilizado para medir a temperatura da água, por ter uma leitura precisa dos dados e ser à prova d'água.

### **2.3.2 Sensor de Temperatura LM35**

Este projeto conta com este sensor de temperatura LM35 de alta precisão e fácil comunicação com o microcontrolador. Exibido na *Figura 4*.





**Figura 4 - Sensor LM35**

O sensor LM35 possui as seguintes características técnicas e físicas:

- Precisão de 0,5°C;
- Faixa de temperatura: -55°C ~+150°C;
- Tensão de alimentação: 4~30V DC;
- Dimensões (CxLxE): 5x5x4mm;

### **2.3.3 Módulo RTC (*Real Time Clock*) DS1307**

O módulo RTC (*Real Time Clock*) é um relógio digital com bateria autônoma e memória interna que armazena a hora configurada que será utilizada para armazenar e visualizar os dados através dos endereços e informações transferidas comumente via protocolo de comunicação I<sup>2</sup>C.

Este projeto conta RTC DS 1307 modelo popular de RTC (*Real Time Clock*) baseado no chip DS 1307 alimentado por uma bateria LIR2032. Exibido na *Figura 5*.



Figura 5- Módulo RTC DS1307

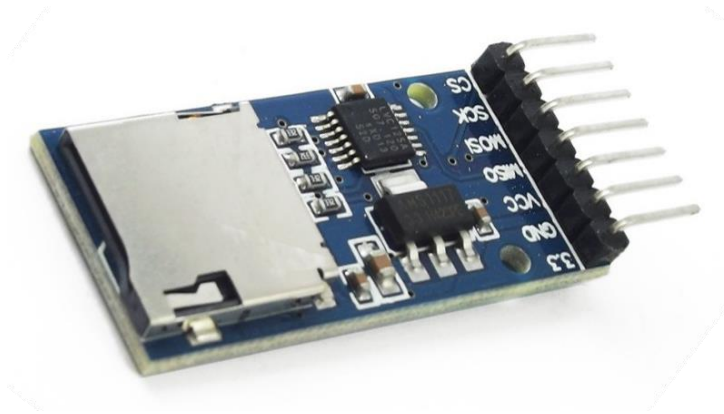
O RTC DS1307 possui as seguintes características técnicas e físicas:

- Calcula os segundos, minutos, horas, dias da semana, dias do mês, meses e anos (de 2000 a 2099);
- Tempo de estabilização: 60 segundos;
- Faixa de temperatura: -40°C a +85°C;
- Tensão de alimentação: 2 ~ 5.5V DC;
- Consumo: 500 mA;
- Dimensões: 27 mm x 28 mm x 8,4 mm (Largura x Altura x Profundidade);

O módulo RTC é um componente de extrema importância, pois é um relógio que nos informa o tempo real com calendário completo e mais de 56 bytes de SRAM, sendo capaz de fornecer informações como segundo, minutos, dia, data, mês e ano, essencial no registro dos dados.

### 2.3.4 Módulo Cartão micro SD

O Módulo de Cartão micro SD (*Secure Digital*) é utilizado no projeto por ser eficiente e compacto nos ajudar a otimizar a aplicação seu uso consiste em salvar os dados fornecido pelos sensores além de armazenar o histórico de ações durante o seu uso podendo auxiliar em eventuais erros durante a execução do Arduino. Exibido na *Figura 6*.



**Figura 6 - Módulo Cartão Micro SD**

Este dispositivo possui as seguintes características técnicas e físicas:

- Comunicação SPI;
- Lê e grava dados em arquivos utilizando o sistema de arquivos FAT32;
- Tensão de alimentação: 4,5 ~ 5V;
- Dimensões: 42 mm x 42 mm x 3,5 mm (Largura x Altura x Profundidade);

O armazenamento de dados é recolhido através de um arquivo de texto único chamado de ARQUIVO.TXT que é gravado em um módulo de cartão micro SD, este é utilizado através do sistema de comunicação SPI e SCM de modo para completar o arquivo para que possa ler e escrever no cartão.

Este módulo usa um cartão micro SD é um formato de cartão de memória não-volátil criado pela Panasonic em 1999, caracterizado por ser de pequena dimensão que permite um grande armazenamento de vários tipos de dados como por exemplo em câmeras fotográficas, telefone móveis etc.

### 2.3.5 Sensor Fluxo de Água ½ YF-S201

O sensor fluxo de água de conexão ½ de polegadas utilizado para medir a vazão de gás do biodigestor.



Figura 7 - Sensor Fluxo de Água 1/2

O sensor fluxo de água ½ possui as seguintes características técnicas e físicas:

- Faixa de tensão: 5 ~ 24V DC;
- Corrente máxima: 15mA (DC 5A);
- Temperatura máxima: Até 80°C;
- Precisão: 0,5 ~ 60 L/min;
- Pressão máxima 1.75Mpa;
- Faixa de umidade de operação: 35% ~ 90° RH;
- Roscas externas: ½;
- Material: PVC;
- Peso 51g;
- Dimensões: 6,2 cm x 3,6 cm x 3,5 cm;

### 2.3.6 Válvula Solenoide

Neste projeto foi utilizado a válvula solenoide para controlar fluxo da água. Ela é composta de duas partes principais, que são: corpo e bobina solenoide.

A bobina é formada por um fio enrolado através de um cilindro, quando a corrente elétrica passa por este fio é gerado uma força no centro da bobina solenoide, fazendo com que êmbolo da válvula seja acionado, criando assim o sistema de abertura e fechamento.

O corpo é responsável pela passagem do fluído ou não, quando sua haste é acionada pela força da bobina um pino é puxado para o centro da bobina permitindo a passagem da água. Esta válvula é do tipo normalmente fechada, ou seja, quando não está acionada, bloqueia a passagem do fluido.



**Figura 8 - Válvula Solenoide**

A válvula solenoide possui as seguintes características técnicas e físicas:

- Tensão de operação: 5-24V
- Pressão máxima: 2,0 Mpa;
- Frequência (Hz): 7,5\* fluxo (L/min)
- Rosca de entrada/saída: ½;
- Pulsos por litro: 450
- Dimensão: 2,5" X 1,4" X 1,4"

## 2.3.7 Módulo Relé

O relé usa os princípios eletromagnéticos para operar corretamente, ele é composto de duas partes, onde no seu interior é composto de uma bobina de cobre que quando é energizada gera um campo magnético e a outra parte é composta de braços metálicos que fazem contato físico para comutação baseado nesse campo magnético.

Os módulos relés mais comuns possuem três terminais de saída: o terminal comum, o terminal NC (do inglês, normalmente fechado) e o terminal NO (do inglês, normalmente aberto). Se o relé não está energizado em seu controle então ele interliga o terminal comum com o terminal NC, e quando é energizado faz a comutação da conexão do terminal comum para o terminal NO, dessa maneira o relé comuta o terminal comum com o terminal NC ou com o terminal NO.



Figura 9 - Módulo Relé

Este módulo relé possui as seguintes características técnicas e físicas:

- Tipo: Digital;
- Sinal de controle: Nível TTL;
- Bobina: 5V DC 75mA;

- Carga nominal do relé: 12A 125 VAC, 7ª 250 VAC;
- Tempo de acionamento de contato: 10ms;

### 2.3.8 Sensor Ultrassônico HC-SR04

Este sensor tem um funcionamento baseado na emissão de ondas sonoras de alta frequência. Este dispositivo calcula a distância entre objetos emitindo uma onda e calculando o tempo que a onda leva até o objeto e refletir de volta para o sensor, utilizando-se da velocidade do som para o cálculo da distância.



Figura 10 - Sensor Ultrassônico HC-SR04

O ultrassônico é uma opção de baixo custo capaz de determinar a distância de um objeto através do sonar, sua capacidade de transmitir, receber e controlar circuitos, o torna de fácil utilização para este tipo de aplicação.

O sensor HC-SR04 possui as seguintes características técnicas e físicas:

- Alimentação: 5V DC;
- Corrente de operação: 2mA;
- Ângulo de efeito: 15°;
- Alcance: 2 cm ~4 m;
- Precisão 3mm;

## Bomba

A bomba d'água adaptada nesse sistema é um modelo utilizado em para-brisa de automóveis. Exibido na *Figura 11*.



**Figura 11 - Bomba de Para-Brisa**

Esta bomba possui as seguintes características técnicas e físicas:

- Tensão de 12V DC
- Vazão: 85ml/seg

## 2.4 Tabela de Pinos

Esta tabela exemplifica as conexões entre o Arduino e demais componentes para

UTILIZAÇÃO	PINOS
Sensor LM35	A0
Sensor DS18B20	3
Sensor Ultrassônico	Echo (5) Trig ( 4)
RTC ( <i>Real Time Clock</i> )	DAS (A4) SCL (A5)



Módulo Cartão Micro SD	MISO (12) MOSI (11) SCK (13) CS (6)
Sensor de Fluxo de Água ½ 2	Sn (2) Gnd (gnd) VCC (5V)
Válvula Solenoide	Sn(7) Gnd (gnd) VCC (5V)
Válvula Solenoide (Bomba)	Sn (8) Gnd (gnd) VCC(5V)
Bomba	Sn (9) Gnd (gnd) VCC (5V)

Tabela 1- Pinos

## 2.5 Ambiente de Desenvolvimento para o Arduino

A linguagem Arduino é baseada em C/C++ chamada de *Processing*, seus programas em Arduino podem ser divididos em três partes principais: estrutura, valores que podem ser variáveis e constantes e funções como visto na *Figura 12*.

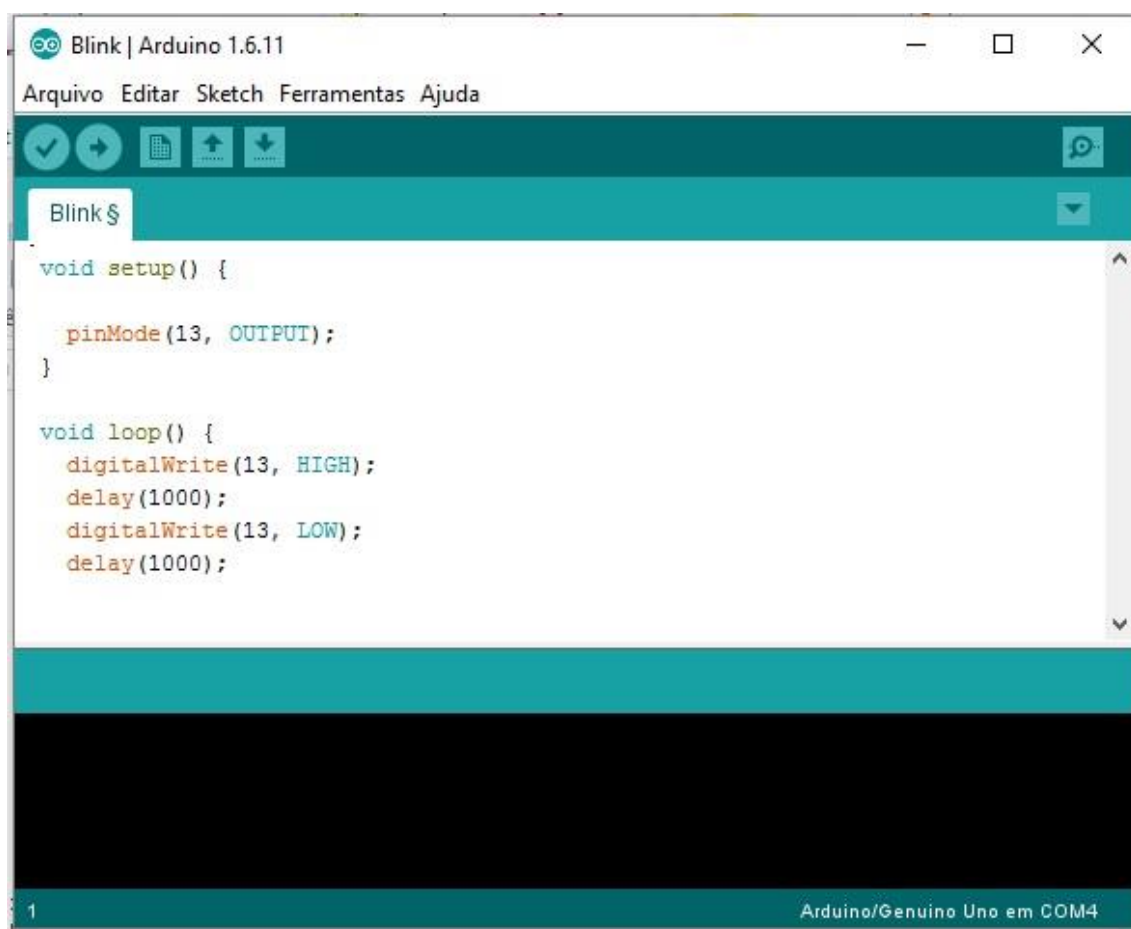


Figura 12 - Ambiente de Desenvolvimento Arduino 1.6.11

Legenda dos ícones (na ordem da esquerda para a direita):



Figura 13 - Legenda de Ícones do Arduino 1.6.11

## 2.5.1 Estrutura do *Sketch*

A estrutura do código para desenvolver um programa para o Arduino é semelhante à estrutura de qualquer código em C, começamos com a declaração de uma biblioteca, seguida das variáveis globais e as funções setup e loop que são obrigatórias do tipo void, assim como mostra a imagem abaixo:



Figura 14 - Estrutura do Sketch

## 3.0 Sensor de Temperatura DS18B20

O sensor de temperatura DS18B20 atuará como dispositivo de leitura que capta a temperatura atual da água e esterco logo após será utilizada pelo Arduino para execução de comandos, este está ligado no pino digital do Arduino, de números 3 os sensores de temperatura estão ligados à fonte de energia de 12V do Arduino alimentado pelo painel solar.

## 3.1 Sensor de Temperatura LM35

O sensor de temperatura LM15 atuará apenas como dispositivo de leitura que capta a temperatura ambiente do solenoide que depois será utilizada pelo

Arduíno para execução de comandos, este está ligado no pino analógico de números A0.

### **3.1 Real Time Clock RTC DS1307**

O RTC DS1307 é responsável por registrar a hora e a data exatas que os dados estão sendo coletados este utiliza-se dos pinos analógicos do Arduino de número DAS (A4) e SCL (A5) conectados à fonte de energia de 12V do microcontrolador.

### **3.2 Módulo Cartão Micro SD**

O módulo cartão SD está desempenhando a função de em conjunto do Arduino que ler e escreve no cartão os dados coletados dos sensores ao escrever ele organiza os dados em uma String separadas por vírgulas que identifica na seguinte ordem os dados transferidos para o cartão: dia da semana, data, hora, temperatura da água, temperatura do esterco, temperatura da serpentina, distancia, vazão do gás, o que facilita todo trabalho ao exportar para uma planilha Excel podemos gerar gráficos e melhorar a análise, este está usando o pino digital de número 6.

### **3.3 Sensor Fluxo de Água YF-S201**

Para adaptar este sensor é importante conhecer sua estrutura ele tem um sentido horário para operar na parte interna ele tem uma hélice ligado a um eixo que está conectada na parte superior a um sensor de efeito hall, que trabalha em conjunto para enviar um sinal PWM (Pulse Width Modulation) quando o eixo gira ele gera um pulso e através dele é feito o cálculo para obter a vazão de gás, um dos dados essenciais para o cálculo é a vazão de litros por minutos podendo ser encontrado nas especificações do datasheet este sensor usa a modulação de largura de pulso como base para funcionar corretamente.

Este componente estará ligado na mangueira que leva o gás até a residência cujo o sensor estará ligado ao arduino pelos pinos de VCC terra e de impulso, nesse caso utilizaremos um relé.

### **3.4 Válvula Solenoide**

Utilizamos a válvula solenoide para automatizar o controle de fluxo da água aquecida satisfazendo o quesito de funcionalidade ao projeto, adaptada para trabalhar com os fluídos de matéria líquida, esta auxilia na troca da água aquecida com a água fria, este tipo de válvula é muito prático de se trabalhar com Arduino especialmente para a dosagem que água, utilizamos os pinos digitais 8 e 9.

### **3.5 Módulo Relé**

Este dispositivo comutador é importante para possamos acionar todos os dispositivos conectados ao Arduino com uma potência muito maior desde que não ultrapasse 10A. Nessa aplicação ele foi útil para acionarmos o sensor de fluxo de água e a válvula solenoide. Este está conectado ao pino 9.

### **3.6 Sensor Ultrassônico**

O sensor ultrassônico é utilizado neste projeto para medir a altura da caixa d'água em relação ao solo, com ele podemos fazer uma estimativa de quanto esterco ainda resta para ser fermentado no biodigestor, o HC-SR04 usa os pinos Echo (5) e Trig (4) para se comunicar com o microcontrolador.

### **3.7 Bomba**

A bomba d'água de para-brisa foi uma alternativa de baixo custo utilizada com o microcontrolador, reaproveitando a água da serpentina transferindo-a para ser novamente aquecida. Está conectada ao relé.

## **Capítulo 4 – Aplicação da Solução com Resultados**

Os sensores integrados fazem a função de monitorar a os resultados, tal como a temperatura da água, esterco, vazão e serpentina, com o auxílio de módulos podem complementar a aplicação como por exemplo RTC DS1307 que configura a hora e o módulo de cartão micro SD que tem a função de salvar os dados coletados. Uma pequena automação pode fazer a troca da água fria pela quente através de válvulas solenoides que controlam os fluídos de água auxiliado por uma bomba, ambos acionados pelo microcontrolador Arduino Uno.

Os resultados comprovam que ao utilizar a água aquecida no esterco a atividade de fermentação das bactérias aumenta consideravelmente, através deste sistema é possível estipular o quanto de gás está sendo produzido pelo biodigestor e qual é vazão aproximada desse biogás.

A solução proposta mostra que o clima das cidades analisadas no Agreste Meridional de Pernambuco é propícia a utilizar o biodigestor as famílias economizariam e contribuiriam para o meio ambiente.

## **Capítulo 5 – Considerações finais**

### **5.1 Conclusões**

Após experiências na instalação e operação de biodigestores na região do Agreste Meridional de Pernambuco é viável utilizando tecnologias simples baseada no sistema embarcado é possível monitorar o desempenho e controlar a produção de Biogás nessas unidades de Biodigestores que usam como matéria orgânica o esterco dos bovinos, em uma região conhecida como bacia leiteira apostar na implantação dessa energia alternativa seria uma opção principalmente no aspecto financeiro e contribuindo para diminuição da poluição

no meio rural com uma solução barata e eficiente para que os moradores da região.

## 5.2 Sugestões para Trabalhos Futuros

À medida que o projeto foi sendo desenvolvido, surgiram várias ideias para agregar ao projeto, uma das ideias que surgiu foi de aprimorar o Biodigestor para implantá-lo nas escolas públicas da zona rural aproveitando o esterco, ou até mesmo fechar uma parceria da comunidade com os órgãos públicos do município para manter o sistema funcionando e em troca teria uma fonte de energia para cozer alimentos para uma enorme quantidade de alunos e funcionários. Outra alternativa seria o município financiar o biodigestor através de um projeto social ajudando assim milhares de famílias, colaborando diretamente com a redução das despesas deixando de usar o gás GLP, além de gerar adubo para suas plantações eliminando qualquer tipo de desperdício de material orgânico utilizado no biodigestor.

Porque não aumentar a eficiência do Biodigestor? Com o monitoramento pode-se fazer vários testes com materiais diferentes afim de obter o melhor desempenho fica claro que esta energia alternativa é uma tecnologia poderosa que precisa ser dominada.

## 6.0 Bibliografia

ARDUINO Board Uno. **Arduino**. Disponível em: <<https://www.arduino.cc/en/Main/arduinoBoardUno>>. Acesso em: 25 Outubro 2016.

CAMPOS, A. Criando uma biblioteca para o Arduino. **BR-Arduino.org**, 2016. Disponível em: <<http://br-arduino.org/2014/12/criando-uma-biblioteca-para-o-arduino-para-facilitar-o-reuso-de-suas-funcoes.html>>. Acesso em: 26 Outubro 2016.

CONTROL DE PROCESO EN BIODIGESTORES. AquaLimpia Engineering e.K.. Uelzen, p. 10. 2013.

FONSECA, E. G. P. Apostila Arduino, Rio de Janeiro, Outubro 2010.

GARAGEM, L. Tutorial: como utilizar o Sensor de temperatura DS18B20. **Lab de Garagem**, 2012. Disponível em: <<http://labdegaragem.com/profiles/blogs/tutorial-como-utilizar-o-sensor-de-temperatura-ds18b20-prova-de>>. Acesso em: 29 Outubro 2016.

GARAGEM, L. Tutorial: Como utilizar o Sensor de Fluxo de Água. **Lab de Garagem**, 2014. Disponível em: <<http://labdegaragem.com/profiles/blogs/tutorial-como-utilizar-o-sensor-de-fluxo-de-agua>>. Acesso em: 27 Outubro 2016.

SAVITCH, W. **C++ Absolute**. 6ª. ed. [S.l.]: PEARSON EDUCATION BRASIL, 2015.

## 7.0 Programa Inserido no Arduino

```
#include <DS1307.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Ultrasonic.h>
#include <SD.h>
#include <String.h>
#include "LiquidCrystal.h"

//PINOS DO SENSOR ULTRASSONICO
#define pino_trigger 4
#define pino_echo 5

//PINO DO SENSOR DE TEMPERATURA INTERNA
#define ONE_WIRE_BUS 3
```



```

//PINO DO SENSOR DE VAZÃO
#define FLOWSENSORPIN 2

//DIFERENÇA DE TEMPERATURA
#define diferencaTempMin 10
float diferencaTemp;

//PINO DA VALVULA DO RESERVATORIO
int pino_valvula_reservatorio = 7;

//PINO DA VALVULA DA BOMBA
int pino_valvula_bomba = 8;

//PINO DA BOMBA
int pino_bomba = 9;

//PINO DO SENSOR DE TEMPERATURA AMBIENTE
const int LM35 = A0;

/**Sensor Ultrassonico**
Ultrasonic ultrasonic(pino_trigger, pino_echo);

/**Sensor de Temperatura**
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature temperature(&oneWire);
DeviceAddress address;

/**Modulo RTC**
DS1307 rtc(A4, A5);

/**Cartão SD**
Sd2Card cartaoSD;
const int pinoCartaoSD = 6;
//Serve para printar as informações no SD pela primeira vez.
int count = 0;

/**Sensor de Fluxo**
//Conta os pulsos do sensor de fluxo
volatile uint16_t pulses = 0;
//Controla o estado do pino de pulso
volatile uint8_t lastflowpinstate;
//Registra o tempo entre os pulsos
volatile uint32_t lastflowratetimer = 0;
//Usa o lastflowratetimer para calcular o flowrate
volatile float flowrate;

```

```

void setup(void)
{
    Serial.begin(9600);

    Serial.print("DiaDaSemana;Data;Hora;TemperaturaReservatorioAquecido;TemperaturaReservatori
o;Distancia;TemperaturaAmbiente;Vazao;\n");
    temperature.begin();
    iniciaComunicacaoSD();
    if (!temperature.getAddress(address, 0))
        mostra_endereco_sensor(address);
    SD.begin(pinoCartaoSD);
    //Define a parada do RTC
    rtc.halt(false);
    //Define o dia da semana
    //rtc.setDOW(SEXTA);
    //Define o horario
    //rtc.setTime(13, 03, 00);
    //Define o dia, mes e ano
    //rtc.setDate(11, 11, 2016);
    //Definicoes do pino SQW/Out
    rtc.setSQWRate(SQW_RATE_1);
    rtc.enableSQW(true);
    //Define o pino do sensor de fluxo
    pinMode(FLOWSENSORPIN, INPUT);

    //Define o pino do sensor de fluxo como high
    digitalWrite(FLOWSENSORPIN, HIGH);

    //colhe as informações do pino do sensor de fluxo e armazena na variavel lastflowpinstate
    lastflowpinstate = digitalRead(FLOWSENSORPIN);
    useInterrupt(true);

    //DEFINE O PINO DA VALVULA DO RESERVATORIO COMO SAIDA
    pinMode(pino_valvula_reservatorio, OUTPUT);

    //DEFINE O PINO DA VALVULA DA BOMBA COMO SAIDA
    pinMode(pino_valvula_bomba, OUTPUT);

    //DEFINE O PINO DA BOMBA
    pinMode(pino_bomba, OUTPUT);
}

void loop(){

```

```

float tempAmbiente = calcTemperaturaAmbiente();
float temperaturaReservAquecido = mostrarTemperaturaReservAquecido();
float temperaturaReserv = mostrarTemperaturaReserv();
float distancia = mostrarDistancia();
float vazao = mostrarVazaoGas();
String data = mostrarData();
String hora = mostrarHora();
String diaDaSemana = mostrarDiaDaSemana();
diferencaTemp = temperaturaReservAquecido - temperaturaReserv;

Serial.print(diaDaSemana);
Serial.print(";");
Serial.print(data);
Serial.print(";");
Serial.print(hora);
Serial.print(";");
Serial.print(temperaturaReservAquecido);
Serial.print(";");
Serial.print(temperaturaReserv);
Serial.print(";");
Serial.print(distancia);
Serial.print(";");
Serial.print(tempAmbiente);
Serial.print(";");
Serial.print(vazao);
Serial.print(";");

gravarSDs(diaDaSemana);
gravarSDs(data);
gravarSDs(hora);
gravarSDf(temperaturaReservAquecido);
gravarSDf(temperaturaReserv);
gravarSDf(distancia);
gravarSDf(tempAmbiente);
gravarSDf(vazao);
gravarSDf(-100);
delay(12000);

if(hora.toInt() > 6 && hora.toInt() < 18){
    trocaDeAgua();
}
}

/**FUNÇÃO DA AUTOMAÇÃO**
void trocaDeAgua(){
    if(diferencaTemp < diferencaTempMin){

```

```

    ligarValvulaBomba();
    ligarBomba();
    ligarValvulaReservatorio();
    delay(234000);
    desligarBomba();
    desligarValvulaBomba();
    delay(60000);
    desligarValvulaReservatorio();
}
}

/**FUNÇÕES DO SENSOR DE TEMPERATURA INTERNA**
float mostrarTemperaturaReservAquecido(){
    temperature.requestTemperatures();
    float tempC = temperature.getTempC(address);
    return tempC;
}

float mostrarTemperaturaReserv(){
    temperature.requestTemperatures();
    float tempC = temperature.getTempC(address);
    return tempC;
}

void mostra_endereco_sensor(DeviceAddress deviceAddress)
{
    for (uint8_t i = 0; i < 8; i++)
    {
        // Adiciona zeros se necessário
        if (deviceAddress[i] < 16) Serial.print("0");
        Serial.print(deviceAddress[i], HEX);
    }
}

/**FUNÇÃO DO SENSOR DE TEMPERATURA AMBIENTE**
float calcTemperaturaAmbiente(){
    float temperaturaAmbiente;
    temperaturaAmbiente = (float(analogRead(LM35))*5/(1023))/0.01;
    return temperaturaAmbiente;
}

/**FUNÇÕES DO SENSOR DE DISTANCIA**
float mostrarDistancia(){
    float cmMsec;
    long microsec = ultrasonic.timing();
    cmMsec = ultrasonic.convert(microsec, Ultrasonic::CM);

```

```

    return cmMsec;
}

/**FUNÇÕES DO CARTÃO SD**
void iniciaComunicacaoSD(){
    if (!SD.begin(pinoCartaoSD))
    {
        Serial.println("Falha ao acessar o cartao !");
        Serial.println("Verifique o cartao/conexoes e reinicie o Arduino...");
        return;
    }
    Serial.println("Cartao iniciado corretamente !");
}

void gravarSDf(float sensor){
    SD.begin(pinoCartaoSD);
    File dataFile = SD.open("arquivo.txt", FILE_WRITE);
    if(count == 0){
        dataFile.println("Temperatura;Distancia;DiaDaSemana;Data;Hora;Vazão;");
        count++;
    }
    // Grava os dados no arquivo
    if (dataFile)
    {
        if(sensor == -100){
            dataFile.println();
        }else{
            dataFile.print(sensor);
            dataFile.print(";");
        }
        dataFile.close();
    }
    else
    {
        Serial.println("Erro ao abrir arquivo.txt !");
    }
}

void gravarSDs(String sensor){
    SD.begin(pinoCartaoSD);
    File dataFile = SD.open("arquivo.txt", FILE_WRITE);
    if (dataFile)
    {
        dataFile.print(sensor);
        dataFile.print(";");
        dataFile.close();
    }
}

```

```

    }
    else
    {
        Serial.println("Erro ao abrir arquivo.txt !");
    }
}

```

```

/**FUNÇÕES DO MODULO RTC**

```

```

String mostrarData(){
    String data = "";
    data = rtc.getDateStr(FORMAT_SHORT);
    return data;
}

```

```

String mostrarHora(){
    String hora = "";
    hora = rtc.getTimeStr();
    return hora;
}

```

```

String mostrarDiaDaSemana(){
    String dia = "";
    dia = rtc.getDOWStr(FORMAT_SHORT);
    return dia;
}

```

```

/**FUNÇÕES DO SENSOR DE VAZÃO DE GÁS**

```

```

float mostrarVazaoGas(){
    float liters = pulses;
    liters /= 7.5;
    liters /= 60;
    return liters *= 1000;
}

```

```

//A INTERRUPTÃO É CHAMADA UMA VEZ POR MILISSEGUNDO, PARA PROCURAR POR
PULSOS DO SENSOR

```

```

SIGNAL(TIMER0_COMPA_vect) {
    uint8_t x = digitalRead(FLOWSSENSORPIN);

```

```

    if (x == lastflowpinstate) {
        lastflowratetimer++;
        return;
    }

```

```

    if (x == HIGH) {
        //TRANSIÇÃO DO LOW PARA HIGH

```

```

    pulses++;
}
lastflowpinstate = x;
flowrate = 1000.0;
flowrate /= lastflowratetimer; // EM HERTZ
lastflowratetimer = 0;
}

void useInterrupt(boolean v) {
    if (v) {

        OCR0A = 0xAF;
        TIMSK0 |= _BV(OCIE0A);
    } else {

        TIMSK0 &= ~_BV(OCIE0A);
    }
}

/**FUNÇÕES DA VALVULA DO RESERVATORIO**
void ligarValvulaReservatorio(){
    digitalWrite(pino_valvula_reservatorio, HIGH);
}

void desligarValvulaReservatorio(){
    digitalWrite(pino_valvula_reservatorio, LOW);
}

/**FUNÇÕES DA VALVULA DA BOMBA**
void ligarValvulaBomba(){
    digitalWrite(pino_valvula_bomba, HIGH);
}

void desligarValvulaBomba(){
    digitalWrite(pino_valvula_bomba, LOW);
}

/**FUNÇÕES DA BOMBA**
void ligarBomba(){
    digitalWrite(pino_bomba, HIGH);
}

void desligarBomba(){
}

```















