

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
COORDENAÇÃO DE AUTOMAÇÃO INDUSTRIAL
CURSO SUPERIOR DE TÉCNOLOGIA EM AUTOMAÇÃO DE PROCESSOS INDUSTRIAIS

MAURICIO DALLA COSTA

CENTRAL METEOROLÓGICA MICROCONTROLADA DE BAIXO CUSTO

TRABALHO DE CONCLUSÃO DE CURSO

PATO BRANCO

2013

MAURICIO DALLA COSTA

**CENTRAL METEOROLÓGICA MICROCONTROLADA DE BAIXO
CUSTO**

Trabalho de Conclusão de Curso, apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Tecnologia em Automação de Processos Industriais, da Coordenação de Automação Industrial da Universidade Tecnológica Federal do Paraná (UTFPR), Câmpus Pato Branco, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Dr. Fabio Luiz Bertotti

PATO BRANCO

2013

TERMO DE APROVAÇÃO

O Trabalho de Conclusão de Curso intitulado “**CENTRAL METEOROLÓGICA MICROCONTROLADA DE BAIXO CUSTO**”, do acadêmico **MAURICIO DALLA COSTA** foi considerado **APROVADO** de acordo com ata de defesa nº **142** de **2013**, a qual encontra-se na coordenação do curso.

Fizeram parte da Banca os Professores

FÁBIO LUIZ BERTOTTI (ORIENTADOR)

ANDRÉ MACARIO BARROS

EMERSON GIOVANI CARATI

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer a meus pais e minha família, por todo o apoio prestado nos momentos em que precisei de suporte durante o curso e na vida. Devo primeiramente a eles essa conquista.

Agradeço ao meu professor orientador Dr. Fabio Luiz Bertotti pelas orientações, pela ajuda e apoio nos momentos difíceis durante o desenvolvimento do trabalho de conclusão de curso.

Agradeço também aos demais professores do curso que ministraram aulas, e a instituição UTFPR.

E também agradeço a todos os meus amigos, que estiveram sempre junto comigo em tudo que precisei.

RESUMO

DALLA COSTA, Mauricio. Estação Meteorológica Microcontrolada de baixo custo. 2013. 57.f. Monografia (Trabalho de Conclusão de Curso) – Graduação em Tecnologia em Automação de Processos Industriais, Universidade Tecnológica Federal do Paraná. Pato Branco, 2013.

Este trabalho apresenta o desenvolvimento de uma estação meteorológica, para monitoramento de temperatura, pressão atmosférica, umidade relativa do ar, velocidade e posição do vento. Os sensores utilizados fornecem dados meteorológicos em determinado intervalo de tempo para um sistema microcontrolado que processa tais informações e as envia para um computador, através de uma interface de comunicação serial no protocolo RS-232, permitindo que o usuário armazene e visualize estas informações em um terminal de comunicação. Os resultados são apresentados e discutidos no final deste trabalho, mostrando a tela do programa de leitura no computador e gráficos com os dados meteorológicos medidos com a estação.

Palavras-Chave: Microcontrolador. Anemômetro. Biruta. Sensores. Comunicação Serial.

ABSTRACT

DALLA COSTA, Mauricio. Low cost Microcontrolled Meteorological Station. 2013. 56f. Graduation Conclusion Work (Graduation in Industrial Processes' Automation Technology) – Technology Graduation Program, Federal University of Technology – Paraná. Pato Branco, 2013.

This work presents the development of a meteorological station for temperature, atmospheric pressure, relative humidity, wind speed and direction monitoring. Sensors provide meteorological data in a given time interval for a microcontrolled system that processes such data and sends them to a computer through a serial communication interface based on RS-232 protocol, allowing the user to visualize and store the information in a communication terminal software. The results are presented and discussed at the end of this work, showing a screen reading program in computer graphics and with the meteorological data measured at the station.

Key-words: Microcontroller. Anemometer. Windsock. Sensors. Serial Communication.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 - Curvas das razões de resistência como função da temperatura para termistores do tipo NTC e PTC. | 12 |
| Figura 2 - Tensões normalizadas de saída dos Termopares | 14 |
| Figura 3 - Sensor de pressão do tipo MEMS..... | 15 |
| Figura 4 - Anemômetro do tipo concha de Robinson. | 16 |
| Figura 5 - Anemômetro do "Aviãozinho"..... | 17 |
| Figura 6 – Formas dos sinais em um barramento SPI. | 20 |
| Figura 7 - Diagrama em Blocos da Estação Meteorológica. | 23 |
| Figura 8 - Ligação do sensor SHT11 com Microcontrolador. | 25 |
| Figura 9 – Placa de circuito impresso com o sensor BMP085. | 26 |
| Figura 10 – Anemômetro..... | 26 |
| Figura 11 – Biruta..... | 27 |
| Figura 12 - Tela principal do <i>Code Composer Studio</i> | 29 |
| Figura 13 - Circuito Anemômetro. | 30 |
| Figura 14 - Sequência da comunicação I ² C. | 32 |
| Figura 15 - Registradores de Calibração do BMP085..... | 33 |
| Figura 16 – Procedimento para obter a temperatura e pressão atmosférica do sensor BMP085..... | 34 |
| Figura 17 – Sequência de bits para comunicação com SHT11..... | 35 |
| Figura 18 - Coeficientes de conversão de temperatura para o sensor SHT11..... | 36 |
| Figura 19 - Coeficientes para conversão de umidade relativa do ar do sensor SHT11..... | 37 |
| Figura 20 – Circuito de interface serial com o computador. | 37 |
| Figura 21 - Circuito Fonte de Alimentação. | 38 |
| Figura 22 - Estação meteorológica desenvolvida..... | 40 |
| Figura 23 - Placa Indicadora de Posição do Vento. | 41 |
| Figura 24 - Placa de conexão dos sensores SHT11 e BMP085..... | 42 |
| Figura 25 - Placa do sistema microcontrolado. | 42 |
| Figura 26 - Tela de exposição dos dados ao usuário..... | 43 |
| Figura 27 - Gráfico da temperatura do sensor BMP085..... | 44 |
| Figura 28 - Gráfico da pressão atmosférica. | 44 |
| Figura 29 - Gráfico de umidade relativa do ar. | 44 |

| | |
|---|----|
| Figura 30 - Gráfico de temperatura do sensor SHT11. | 45 |
| Figura 31 - Gráfico de velocidade do vento..... | 45 |
| Figura 32 - Circuito de medição de direção do vento. | 51 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Modelos de Termopares Normalizados. | 14 |
| Tabela 2 - Tabela verdade do multiplexador | 31 |
| Tabela 3 - Tabela de custos da estação. | 46 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|---------------------|--|
| ANSI | <i>American National Standards Institute</i> |
| CI | <i>Circuito Integrado</i> |
| CPU | <i>Central Processor unit</i> |
| E ² PROM | <i>Electrically-Erasable Programmable Read-Only Memory</i> |
| GPS | <i>Global Positioning System</i> |
| I ² C | <i>Inter-Integrated Circuit</i> |
| LED | <i>Light Emitting Diode</i> |
| LSB | <i>least Significant Byte</i> |
| MISO | <i>Master In Slave Out</i> |
| MEMS | <i>Sistemas Micro Eletro-Mecânico</i> |
| MOSI | <i>Master Out Slave In</i> |
| MSB | <i>Most Significant Byte</i> |
| NTC | <i>Negative Temperature Coefficient</i> |
| NTD | <i>Resistive Temperature Detectors</i> |
| PTC | <i>Positive Temperature Coefficient</i> |
| PVC | <i>Polyvinyl Chloride</i> |
| RAM | <i>Random Access Memory</i> |
| RISC | <i>Reduced Instruction Set Computer</i> |
| SCLK | <i>Serial de Clock</i> |
| SCL | <i>Serial Clock</i> |
| SDA | <i>Serial Data</i> |
| SPI | <i>Serial Peripheral Interface</i> |
| TI | <i>Texas Instruments</i> |
| UART | <i>Acrônimo de Universal Asynchronous Receiver-Transmitter</i> |
| USB | <i>Universal Serial Bus</i> |

SUMÁRIO

| | |
|---|----|
| 1. INTRODUÇÃO | 8 |
| 1.1 OBJETIVOS DO PROJETO | 9 |
| 1.1.1 Objetivo Geral | 9 |
| 1.1.2 Objetivos Específicos | 9 |
| 1.2 ORGANIZAÇÃO DO TRABALHO | 9 |
| 2. FUNDAMENTAÇÃO TEORICA | 11 |
| 2.1 SENSORES | 11 |
| 2.1.1 Sensores de Temperatura | 12 |
| 2.1.2 Transdutores de umidade capacitivos | 14 |
| 2.1.3 Transdutores Barométricos | 15 |
| 2.1.4 Sensores de Velocidade do Vento | 16 |
| 2.1.5 Sensores de direção de vento | 17 |
| 2.2 MICROCONTROLADORES | 17 |
| 2.3 PROGRAMAÇÃO DE MICROCONTROLADORES | 18 |
| 2.3.1 Linguagem <i>Assembly</i> | 18 |
| 2.3.1 Linguagem C | 18 |
| 2.4 COMUNICAÇÃO SERIAL | 19 |
| 2.4.1 Protocolo SPI | 20 |
| 2.4.2 Protocolo I ² C | 20 |
| 2.4.3 Padrão RS-232 | 21 |
| 3. MATERIAIS E MÉTODOS | 23 |
| 3.1 MATERIAIS | 24 |
| 3.1.1 Sensor SHT11 | 24 |
| 3.1.2 Sensor BMP085 | 25 |
| 3.1.3 Anemômetro e Biruta | 26 |
| 3.1.4 Kit MSP430 <i>Launchpad</i> | 27 |
| 3.1.5 Microcontrolador MSP430G2553 | 27 |
| 3.1.6 Ferramenta de Desenvolvimento de Código <i>Code Composer Studio</i> | 28 |
| 3.1.7 Aplicativo <i>AcessPort</i> | 29 |
| 3.2 METODOLOGIA | 30 |
| 3.2.1 Circuitos de leitura do Anemômetro e da Biruta | 30 |
| 3.2.2 Sistema Microcontrolado | 32 |
| 4. RESULTADOS | 40 |
| 4.1 PLACA DO CIRCUITO DE DETECÇÃO DA POSIÇÃO DO VENTO | 41 |
| 4.2 PLACA DE INTERCONEXÃO DOS SENSORES | 41 |
| 4.3 PLACA DO SISTEMA MICROCONTROLADO | 42 |
| 4.4 MEDIÇÕES DOS PARAMETROS METEOROLOGICOS | 43 |
| 4.5 CUSTOS DOS MATERIAS UTILIZADOS NA ESTAÇÃO | 44 |
| 5. CONCLUSÕES | 47 |
| APÊNDICE 1 | 51 |
| APÊNDICE 2 | 52 |

| | |
|-----------------|----|
| APÊNDICE 3..... | 55 |
| APÊNDICE 4..... | 56 |
| APÊNDICE 5..... | 57 |

1. INTRODUÇÃO

Atualmente, as informações climáticas têm grande importância para a sociedade moderna, nas mais diversas áreas. A aquisição de variáveis climáticas, o armazenamento e análise destas permite prever com um nível significativo de segurança quando alguns estados climáticos poderão se repetir. Sistemas meteorológicos auxiliam também na prevenção de catástrofes naturais, tão presentes nos dias atuais. Ainda na área da agricultura, torna-se importante desde o processo do plantio até a colheita, pois estes processos dependem de vários fatores climáticos (TORRES, 2013).

A principal forma de saber as condições climáticas hoje é através de jornais e internet, mas a previsão pode se tornar errônea para alguns pontos de uma cidade, como bairros afastados e comunidades do interior, uma vez que a abrangência da previsão é feita para uma localidade ou região. Portanto, torna-se importante o desenvolvimento de equipamentos que façam medições locais, com custo que seja acessível para o usuário, levando a uma previsão do tempo mais precisa.

A partir de sensores de temperatura, umidade, pressão, entre outros, e de um sistema microcontrolado, selecionados visando reduzir os custos, é possível desenvolver uma pequena estação meteorológica, que seja acessível à população, principalmente a produtores rurais, onde os fatores climáticos são de suma importância. As variáveis medidas pela estação podem ser enviadas para um computador, através de uma interface padrão RS-232, para armazenamento e apresentação dos dados ao usuário, resultando em um produto de fácil operação, pois qualquer pessoa que saiba manusear um computador pode facilmente operar a estação.

1.1 OBJETIVOS DO PROJETO

1.1.1 Objetivo Geral

O objetivo geral consiste em desenvolver e implementar uma pequena estação meteorológica microcontrolada de baixo custo, visando obter medidas de temperatura, umidade relativa do ar, pressão atmosférica, velocidade do vento e direção, cujos resultados são apresentados em um computador.

1.1.2 Objetivos Específicos

Para que se atinja o objetivo geral do trabalho, é preciso que alguns objetivos específicos sejam atingidos, tais como:

- Desenvolvimento e implementação de um sistema microcontrolado (*hardware* e *firmware*) para aquisição de dados dos sensores e envio destes para um computador.
- Estudo do funcionamento e interfaceamento com o sistema microcontrolado para os seguintes sensores:
 - Sensor de temperatura e umidade relativa do ar STH11;
 - Sensor de pressão atmosférica BMP085.
- Avaliação e aquisição da velocidade do vento de um anemômetro comercial;
- Desenvolvimento de um sistema de medição de posição para uma biruta pré-concebida;

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho apresenta, no Capítulo 1, uma introdução sobre o tema proposto e os objetivos a serem alcançados. No Capítulo 2, é abordada a fundamentação teórica dos assuntos que são relevantes ao presente trabalho. No Capítulo 3, é apresentado o desenvolvimento do trabalho, onde são revelados os materiais e os métodos empregados no desenvolvimento do projeto. Os resultados

obtidos serão apresentados e comentados no Capítulo 4. Por fim, o Capítulo 5 apresenta a conclusão deste trabalho e sugestões para trabalhos futuros.

2. FUNDAMENTAÇÃO TEORICA

Neste capítulo serão abordados alguns conceitos sobre assuntos que são relevantes para o projeto e desenvolvimento de uma estação de meteorológica, como tipos de sensores existentes, microcontrolador e linguagens de programação, protocolos de comunicação.

2.1 SENSORES

Sensores são dispositivos usados para mensurar fenômenos físicos e, portanto, são sensíveis a alguma forma de energia do ambiente, tais como temperatura, pressão atmosférica, umidade etc. Tipicamente, os sensores possuem um elemento transdutor e fornecem um sinal elétrico que corresponde à grandeza física. Por outro lado, um dispositivo transdutor converte uma forma de energia em outra ou altera uma determinada grandeza elétrica, como resistência, capacitância ou indutância (SEIÇA, 2011).

O sinal de saída de um sensor, normalmente, deve ser manipulado antes de ser utilizado por sistema de controle, pois nem sempre está adequado aos níveis de tensão ou corrente requerido por este sistema. Para isso, é necessário utilizar um circuito de condicionamento de sinal.

Os sensores podem ser analógicos ou digitais. Os sensores analógicos podem assumir em seu sinal de saída ao qualquer valor de tensão ao longo do tempo, dentro de sua faixa de operação. Por outro lado, os sensores digitais podem apenas assumir dois valores distintos de tensão, que são interpretados como zero ou um lógico (WENDDLING, 2010).

2.1.1 Sensores de Temperatura

2.1.1.1 Termistores

Os termistores tem sua resistência elétrica alterada de acordo com a variação de temperatura e são construídos, tipicamente, a partir de óxidos semicondutores que são sintetizados e prensados de varias formas, e são encapsulados em vidro ou epóxi. A principal característica destes dispositivos é a alta sensibilidade devido ao seu elevado coeficiente de temperatura (SHINGUE, 2008).

A maioria dos termistores presentes no mercado é do tipo NTC (*Negative Temperature Coeficient*), que possuem o coeficiente de temperatura negativo, ou seja, sua resistência diminui conforme a temperatura aumenta. Os termistores do tipo PTC (*Positive Temperature Coeficient*) possuem coeficiente de temperatura positivo, que ao contrario dos NTC sua resistência aumenta em função do aumento de temperatura. A Figura 1 a curva em razão das resistências do PTC e NTC.

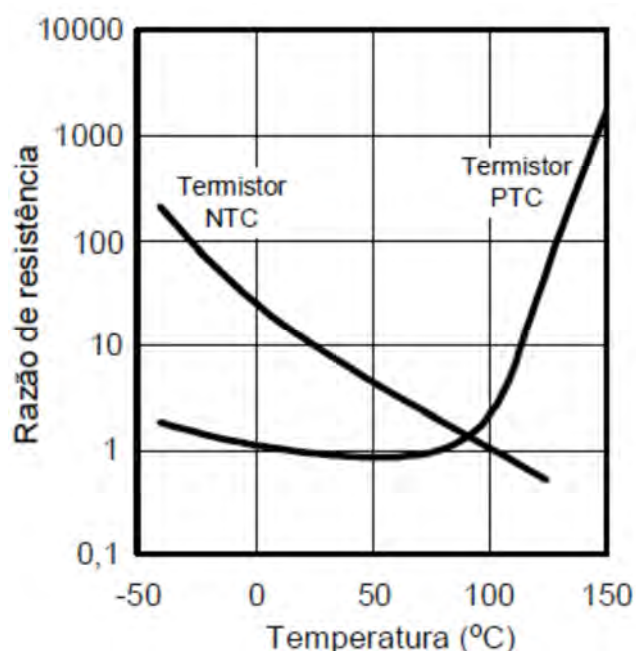


Figura 1 - Curvas das razões de resistência como função da temperatura para termistores do tipo NTC e PTC.

Fonte: Shingue (2008).

Os termistores não são dispositivos lineares e apresentam sensibilidade elevada, onde a resistência varia na ordem de 3% a 5% por °C. A faixa de operação dos termistores varia entre -100°C a +300°C (BRUSAMARELLO, 2013).

2.1.1.2 Termoresistores

Os transdutores de temperatura do tipo termoresistores são conhecidos como RTD (*Resistive Temperature Detectors*) e são compostos por metais de resistência elétrica com elevado coeficiente de temperatura, como a platina, por exemplo. O termoresistor mais utilizado é o PT-100, cuja denominação se deve ao fato de uma resistência de, aproximadamente, 100Ω a 0 °C, e sua faixa de operação de temperatura vai de -200 a + 650°C (BRUSAMARELLO, 2013).

2.1.1.3 Termopares

Os termopares são sensores de medição de temperatura, formados por dois condutores metálicos distintos, puros ou homogêneos. Em uma das extremidades são unidos e soldados, na outra extremidade em que os metais ficam abertos surge uma tensão de circuito aberto V_s proporcional à temperatura T_{tc} na junção do termopar (FRANÇA, 1997). Tensão essa que é chamada de tensão de *seebeck*. A relação entre T_{tc} e V_s é dada por (FRANÇA, 1997):

$$\Delta V_s = S(T_{tc})T_{tc}. \quad (1)$$

Existem vários tipos de termopares disponíveis no mercado, e são classificados de acordo com metal utilizado e designados por uma letra maiúscula de acordo com o ANSI (*American National Standards Institute*). A Tabela 1 mostra os tipos de termopares normalizados, e Figura 2 mostra o gráfico de tensão de saída.

Tabela 1 - Modelos de Termopares Normalizados.

| Tipo | Constituição | Faixa de Temperatura (°C) |
|------|-----------------------------|---------------------------|
| B | Platina/ 30% Ródio-Platina | 0–1800 °C |
| C | Tung-5%Rênio/Tung-26% Rênio | 0–2320 °C |
| E | Cromel / Constantan | -270–1000 °C |
| G | Tungstênio/ Tung-26% Rênio | 0–2300 °C |
| J | Ferro / Constantan | -210–750 °C |
| K | Cromel / Alumel | -270–1370 °C |
| N | Nicrosil / Nisil | -270–1300 °C |
| R | Platina / 13%Ródio-Platina | -50–1750 °C |
| S | Platina / 10%Ródio-Platina | -50–1750 °C |
| T | Cobre / Constantan | -270–400 °C |

Fonte: Silva (2012).

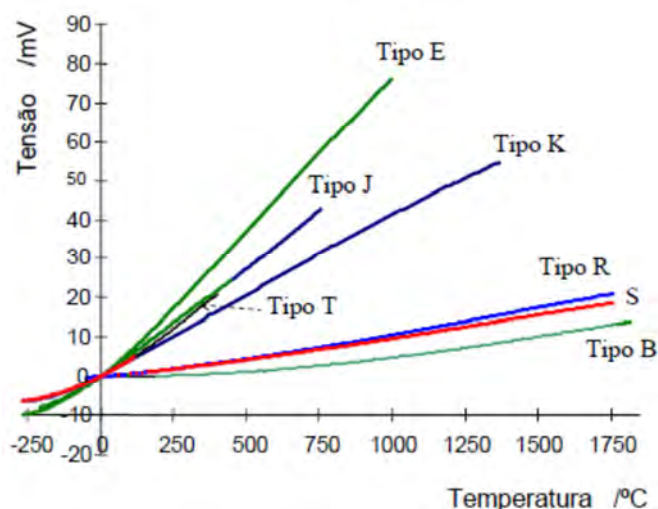


Figura 2 - Tensões normalizadas de saída dos Termopares

Fonte: Silva (2012).

2.1.2 Transdutores de umidade capacitivos

Os transdutores de umidade capacitivos são construídos da mesma forma que um capacitor de placas paralelas, porém com o dielétrico formado por um material polímero que é capaz de absorver as moléculas de água do ar que o circunda, resultando na mudança da constante dielétrica e, conseqüentemente, na capacitância do transdutor. Esta capacitância pode ser obtida a partir de (MULLER, 2008):

$$C(RH) = \varepsilon RH \varepsilon_0 A / d \quad (2)$$

onde ϵ_0 é a permissividade elétrica no vácuo, d é a distância entre as placas paralelas, A é a área das placas e ϵ_{RH} é a permissividade dielétrica relativa, a qual é dependente da umidade.

2.1.3 Transdutores Barométricos

Os transdutores barométricos são usados para medir pressão atmosférica. Transdutores de pressão convertem deslocamento ou deformação mecânica em materiais elásticos, geradas pela pressão atuante, em sinal elétrico. Para uso de medições deve estar presente juntamente com o material elástico, um elemento transdutor elétrico, para converter deformação e sinal elétrico (TEIXEIRA, 2009).

O material mais utilizado na transdução de pressão é o diafragma. A deformação gerada no diafragma e detectada por uma ponte de *Wheatstone* com sensores do tipo *strain gauge* (Extensômetros), que é um elemento que produz uma mudança na sua resistência em função da deformação mecânica (TEIXEIRA, 2009).

Os sensores de pressão mais recentes são do tipo MEMS (Sistemas Micro Eletro-Mecânico) mostrado na Figura 3, são mecanicamente semelhantes aos sensores de pressão tradicionais, porem produzidos em escalas micrométricas. Neste sensor são usados semicondutores no elemento ativo do sensor extensométrico, a capsula transdutora é fabricada com um circuito integrado por difusão de boro num substrato de silício (COELHO, 2013).

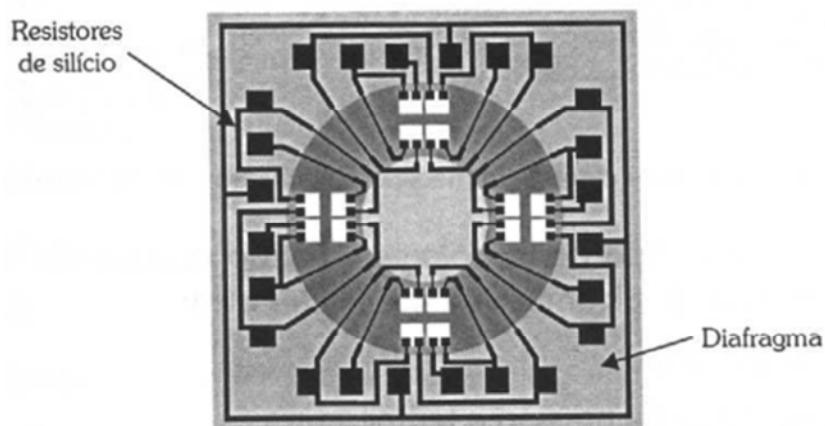


Figura 3 - Sensor de pressão do tipo MEMS
Fonte: Coelho (2013).

2.1.4 Sensores de Velocidade do Vento

Os sensores de velocidade de vento são chamados de anemômetros. Existem diversos modelos de anemômetros, porém os mais utilizados em estações meteorológicas são os rotacionais, podendo ser do tipo rotor horizontal de conchas de Robinson ou do tipo “aviãozinho” (Rotação sobre um eixo Horizontal), conforme mostram as Figura 4 e Figura 5, respectivamente (MOURA, 2000).

Os anemômetros do tipo concha de Robinson são formados por três ou mais conchas montadas simetricamente, formando ortogonais com um eixo vertical. Independente da direção onde o vento sopra o conjunto de conchas faz mover um mecanismo que aciona o dispositivo usado para contar o numero rotações, esse dispositivo pode ser *encoder*, *reed switch*, entre outros.

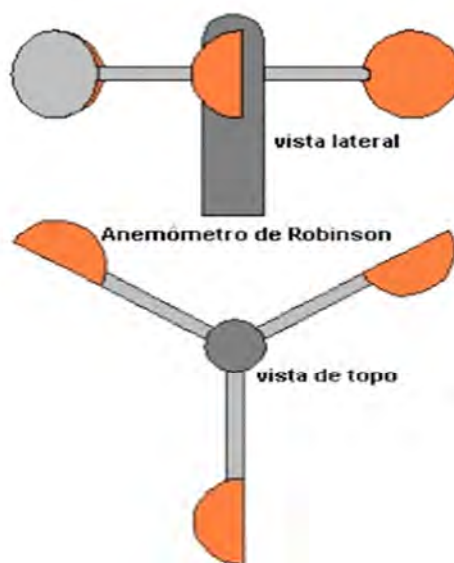
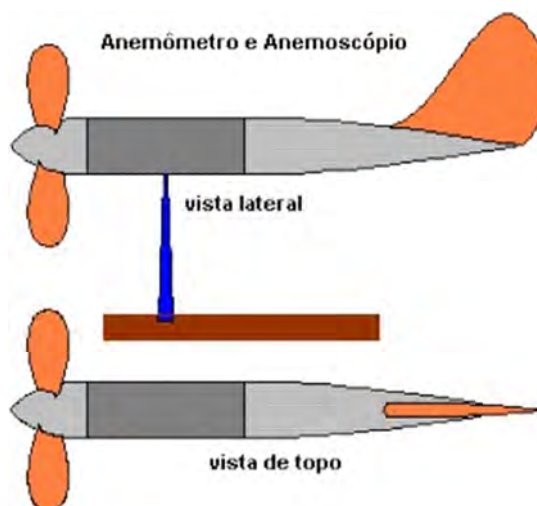


Figura 4 - Anemômetro do tipo concha de Robinson.
Fonte: Moura (2000).

Os anemômetros do tipo “aviãozinho” são dotados de uma hélice para fluxo de ar no sentido axial. Este tipo de anemômetro pode medir velocidades elevadas, tipicamente até 320 Km/h. Diferentemente do anemômetro de conchas, o qual funciona independentemente da direção do vento, o anemômetro tipo “aviãozinho” precisa de uma espécie de leme, um anemoscópio colocado na horizontal do lado oposto da hélice para direcioná-lo no sentido do vento (ARAÚJO, 2009).



**Figura 5 - Anemômetro do "Aviãozinho".
Fonte: Moura (2000).**

2.1.5 Sensores de direção de vento

O dispositivo utilizado para medir a direção de vento é denominado biruta. Este é um instrumento simples constituído de por um cone de tecido flexível aberto nas duas pontas, sendo que a abertura da frente é maior e é acoplada a um arco de metal. Quando o vento sopra através destas aberturas, a biruta é orientada na posição horizontal apontando a direção do vento. A biruta é um instrumento visual, mas pode ser facilmente acoplado a um sistema eletrônico para medir a posição do seu eixo (TÉCNICAS..., 2013).

2.2 MICROCONTROLADORES

Um microcontrolador é um pequeno dispositivo eletrônico programável capaz de executar instruções lógicas e aritméticas sequencialmente. Este dispositivo é um sistema computacional completo, onde estão incluídos internamente uma CPU (*Central Processor Unit*), memórias RAM, *flash* e E²PROM (*Electrically-Erasable Programmable Read-Only Memory*), pinos de entrada e saída, conversor A/D (Analogico para Digital), interfaces seriais, módulos de temporização, entre outros, todos integrados em um único encapsulamento (chip). Este dispositivo é utilizado em diversas áreas, dentre estas a automação industrial, a área automobilística, os eletrodomésticos, produtos eletroeletrônicos, entre outros (JUCA, 2012).

2.3 PROGRAMAÇÃO DE MICROCONTROLADORES

As linguagens de programação são vocabulários, conjunto de regras gramaticais usadas pra escrever programas, sendo que cada linguagem tem um conjunto único de palavras-chave e uma sintaxe específica para organizar as instruções do programa (SILVA P. S., 2013).

Para a programação de microcontroladores existem várias linguagens, de alto nível até linguagens de nível próximo à linguagem de máquina. As principais linguagens usadas na programação de microcontroladores são a linguagem C e *Assembly* (CORTELETTI, 2006).

2.3.1 Linguagem *Assembly*

A linguagem *Assembly* é uma forma de representação de códigos de máquina usando mnemônicos, abreviações de formas que descrevem a operação efetuada pelo comando em código de máquina (PEREIRA, 2003).

Pela facilidade de otimização do código conferida pela linguagem *assembly*, é possível aproveitar ao máximo as características particulares do *hardware*, onde o programa vai ser executado, tendo assim como principais vantagens melhores tempos de execução e menor tamanho do código gerado. Porém, como desvantagem principal, o programador necessita saber detalhes internos do microcontrolador, como conjunto de instruções e registradores (CORTELETTI, 2006).

2.3.2 Linguagem C

A linguagem C foi criada por Dennis Ritchie da Bells Labs em 1972 e é considerada uma linguagem de nível intermediário, ou seja, com nível superior à linguagem *assembly* e inferior às linguagens orientadas a objetos, proporcionando equilíbrio entre estruturação de código e eficiência na geração de código de máquina (CORTELETTI, 2006).

A linguagem C, diferentemente da linguagem *assembly*, necessita que o programa fonte gerado pelo programador seja compilado e processado por outro programa, conhecido como montador (*linker*), o qual gera o programa em linguagem de máquina para gravação na memória do microcontrolador (CORTELETTI, 2006).

As vantagens de programar em linguagem C se tornam evidentes pelo fato dos compiladores oferecerem bibliotecas, ou seja, códigos que contém definições e funções prontas para o usuário utilizar em seu código. As principais desvantagens do uso da linguagem C em microcontroladores, quando comparado com a linguagem *assembly*, consistem do maior tamanho do código de máquina gerado e do maior tempo requerido para sua execução (CORTELETTI, 2006).

Optou-se pelo uso da linguagem C na programação do microcontrolador, pelo conhecimento da linguagem, e pelo fato do compilador oferecer bibliotecas prontas, tornando mais fácil a programação.

2.4 COMUNICAÇÃO SERIAL

Na comunicação serial, os bytes que formam uma mensagem são separados em bits e enviados um por vez através de um canal de transmissão com poucas vias. Quando comparado com uma transmissão paralela de dados, a comunicação serial proporciona maior simplicidade e menor custo, apesar da taxa de transmissão de dados ser menor. A comunicação serial é empregada em aplicações que não exigem taxas elevadas transferência de dados e onde a distância do enlace de comunicação é longa (CANZIAN, 2013).

A comunicação pode ser síncrona, quando a transmissão de dados é controlada por um sinal de *clock*, ou assíncrona. Além disso, o modo de transmissão de dados nas vias de comunicação pode ser unidirecional ou bidirecional (full duplex). Os principais protocolos de comunicação serial usados em microcontroladores são UART (Acrônimo de *Universal Asynchronous Receiver-Transmitter*), SPI (*Serial Peripheral Interface*), I²C (*Inter-Integrated Circuit*) (CUNHA, 2013).

2.4.1 Protocolo SPI

O protocolo de comunicação SPI foi desenvolvido pela Motorola para interligar circuitos integrados inteligentes e, devido a sua simplicidade, vários outros fabricantes adotaram este padrão, o qual é utilizado em uma grande variedade de componentes encontrados em sistemas embarcados.

Trata-se de um protocolo serial síncrono que é capaz de operar em modo bidirecional e atingir taxas de bits superiores a 2 Mbps. O barramento SPI especifica os seguintes sinais lógicos, SCLK (*Serial Clock*) é gerado pelo mestre para sincronizar a transferência de bits, MOSI (*Master Out Slave In*) transporta os dados do mestre para o escravo e MISO (*Master In Slave Out*) transporta os dados do escravo para o mestre (MACIALI, SOARES, & AGUIAR, 2013).

Quando mais de um dispositivo é conectado ao barramento SPI, além dos sinais SCLK, MOSI e MISO, é necessário um sinal de *enable* ou *chip select* para cada dispositivo, cuja função é selecionar o dispositivo desejado para comunicação. A Figura 6 apresenta um exemplo de comportamento dos sinais do barramento SPI ao longo do tempo.

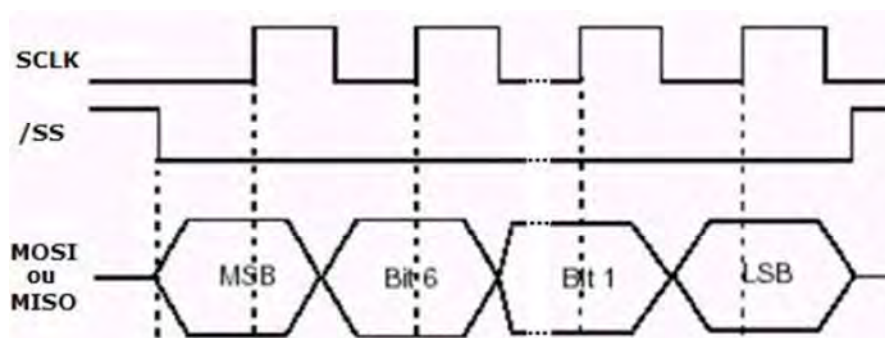


Figura 6 – Formas dos sinais em um barramento SPI.
Fonte: Espindola & Piecarz (2007).

2.4.2 Protocolo I²C

O protocolo de comunicação I²C foi desenvolvido pela Philips para conexão de seus circuitos integrados seus produtos eletrônicos. O barramento I²C é formado por duas linhas bidirecionais, sendo uma delas destinada ao envio e recebimento de dados e endereços, denominada de SDA (*Serial Data*) e a outra

linha corresponde ao sinal de *clock*, denominada SCL (*Serial Clock*) para fins de sincronismo da transmissão de dados (CAETANO, 2010).

O barramento I²C permite conectar vários dispositivos, pois o I²C é capaz de selecionar o dispositivo ao qual deseja comunicação, através de um endereço que cada escravo possui. Desta forma quando o mestre deseja se comunicar com um dispositivo escravo, este envia um *byte* com o endereço do escravo, o qual recebe este byte e reconhece que o mestre receberá ou enviará dados de ou para este dispositivo (CAETANO, 2010).

2.4.3 UART

O UART é um protocolo de comunicação entre dispositivos digitais baseado na comunicação serial assíncrona. A principal função do UART é fragmentar caracteres em bits para transmissão. A comunicação ocorre através do envio de um *start* bit para informar começo de uma transmissão, sete bits de dados se tiver em formato de texto ou oito se tiver em formato binário, um bit de paridade e um *stop* bit marcando o fim da transmissão. A velocidade de transferência de bits (*baud rate*) devem ser a mesma configurada no dispositivo que faz a transmissão e no que faz a recepção dos dados (NARCISO et al., 2013).

2.4.4 Padrão RS-232

O padrão RS-232 foi criado para comunicação de dados entre equipamentos, principalmente para comunicação de periféricos aos computadores. Mesmo perdendo grande parte do seu mercado para o protocolo USB (*Universal Serial Bus*), esta porta de comunicação ainda está presente em alguns computadores, onde apresentam o conector tipo DB9 ou DB25, ambos do tipo macho, sendo que o uso do conector tipo DB9 é mais frequente (CANZIAN, 2013).

A comunicação do tipo assíncrona é associada ao padrão RS-232, onde a conexão é feita com interfaces do tipo UART. Dos nove pinos do conector DB9, apenas três são usados para comunicação, sendo o pino 2 associado à recepção de dados (RX), pino 3 associado à transmissão de dados (TX), o pino 5 é o sinal de

terra (GND). Os sinais são transmitidos em níveis de tensão de -18 a -3V para nível lógico 1 e de 3 a 18 V para nível lógico 0 (CANZIAN, 2013).

Como não há sinal sincronismo entre o transmissor e receptor, ambos os dispositivos devem ser configurados com a mesma taxa de comunicação que é chamada de *baud rate*. O *baud rate* é a velocidade com que os bytes são enviados através de um canal e é medido em transições elétricas por segundo. Os valores mais comuns de *baud rate* são 2400, 4800 e 9600bps (CANZIAN, 2013).

3. MATERIAIS E MÉTODOS

O funcionamento da estação meteorológica é representado a partir do fluxograma mostrado na Figura 7. Segundo este diagrama, o sistema microcontrolado faz a aquisição dos dados climáticos dos sensores, periodicamente em um determinado período de tempo.

A estação possui o sensor SHT11 que mede temperatura e umidade relativa do ar, que se comunica com o microcontrolador através de um protocolo de comunicação desenvolvido, faz a medição de pressão atmosférica através do sensor BMP085 que se comunica com o microcontrolador através do protocolo de I²C. Possui uma placa de circuito que indica o ponto cardeal, mostrando posição do vento. E faz a medição da velocidade do vento através de um sensor magnético conectado a um anemômetro, o qual gera pulsos, que são identificados pelo microcontrolador. Por fim faz a comunicação do sistema com um computador para exposição dos dados ao usuário em um terminal de comunicação o *AcessPort*, através do protocolo de comunicação RS-232.

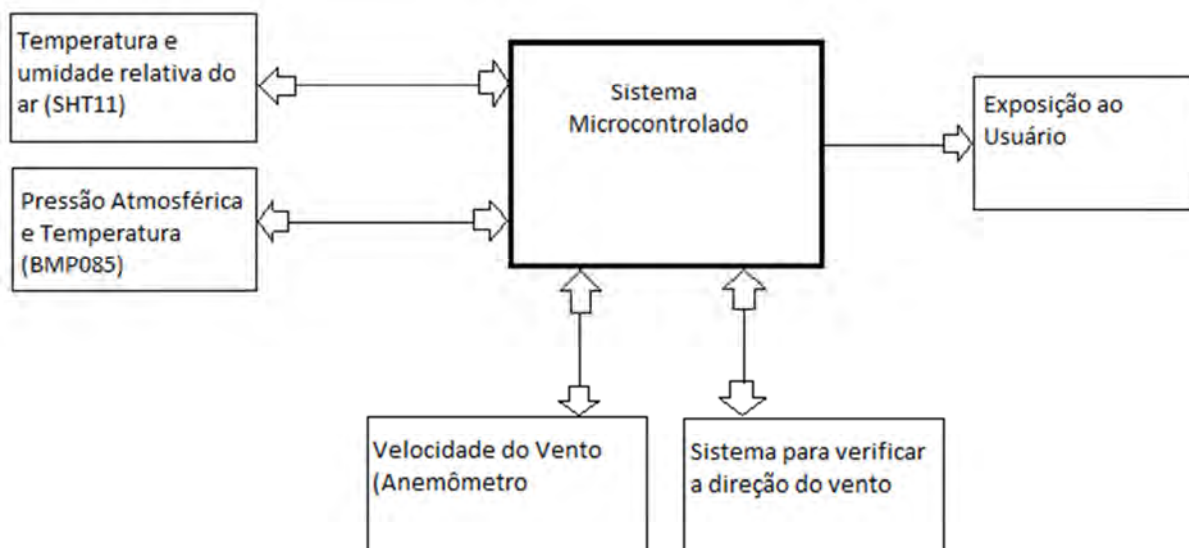


Figura 7 - Diagrama em Blocos da Estação Meteorológica.
Fonte: Autoria própria.

3.1 MATERIAIS

Os materiais utilizados para medição das condições climáticas propostas, e no sistema microcontrolado, assim como *softwares* utilizados para programação e depuração do código no microcontrolador, e o aplicativo para apresentação dos dados no computador serão abordados nesta seção.

3.1.1 Sensor SHT11

Para medir temperatura e umidade relativa do ar foi usado o sensor SHT11, desenvolvido pela empresa Sensirion, que é capaz de medir temperatura e umidade com alta precisão, possuindo transdutores e circuitos analógicos e digitais integrados em um único encapsulamento, onde a saída de dados é digital (SENSIRION, 2011).

O sensor SHT11 emprega em seu transdutor um polímero capacitivo sensível a uma faixa relativa de umidade de 0 a 100%, enquanto que para medição de temperatura o dispositivo usado é do tipo “*bandgap*”, que atua em temperaturas na faixa de -40°C a 120°C. Um conversor A/D de 14 bits efetua a conversão dos sinais analógicos proveniente dos transdutores integrados em palavras digitais que, através de um circuito lógico específico, cria um protocolo para transmissão serial destas informações (SENSIRION, 2011).

Para o funcionamento do SHT11 é necessário alimentá-lo com uma tensão entre 2,4V e 5,5V. A interface serial de comunicação possui dois sinais, sendo que o primeiro deles é o sinal de *clock* SCK para sincronização com o microcontrolador, e o segundo é o sinal de dados DATA que é bidirecional, portanto permite o envio de dados do microcontrolador para o sensor e vice-versa. A Figura 8 mostra um diagrama de interligação deste sensor com um microcontrolador.

O sensor SHT11 foi escolhido por ser um sensor integrado com comunicação de dados digital, por possuir uma ótima precisão, além de vir calibrado de fábrica e por ter um baixo consumo de energia.

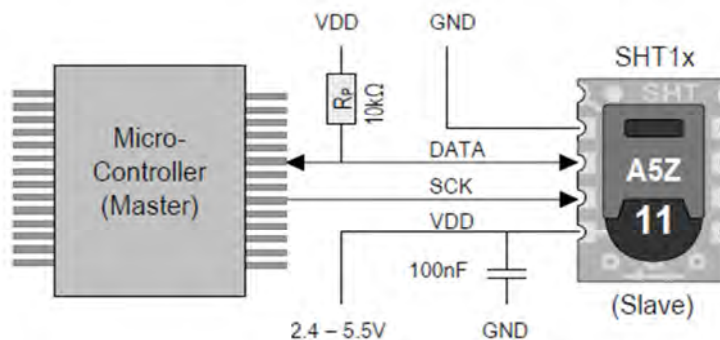


Figura 8 - Ligação do sensor SHT11 com Microcontrolador.
Fonte: Sensirion(2011).

3.1.2 Sensor BMP085

Para medir pressão atmosférica usou-se o sensor BMP085, fabricado pela empresa *Bosh*, é um sensor de pressão barométrica, e de temperatura, de alta precisão e linearidade, baseado na tecnologia piezo-resistivo, que disponibiliza uma interface I²C para comunicação com o microcontrolador. Esse dispositivo pode ser alimentado com uma tensão entre 1,8V e 3,6V, e seu consumo de energia fica na ordem de 1 μ A em *stand-by* e 12 μ A em modo de maior resolução, sendo otimizado para utilização em dispositivos móveis, telefone, GPS (*Global Positioning System*), entre outros (SENSORTEC, 2009).

O sensor BMP085 efetua medidas de pressão atmosférica na faixa 300 a 1100hpa com precisão de 0,03hpa em seu modo de resolução máxima, e medidas de temperatura na faixa -40°C a 65°C com precisão de 0.5°C. O dispositivo adquirido é montado em uma placa de circuito impresso, formando um modulo, conforme mostrado na Figura 9, possuindo um pino de sinal de dados SDA, de sinal de *clock* SCL, e de *reset* XCLR, e alimentação (Vcc e GND).

Este sensor foi escolhido para medir pressão atmosférica por sua alta precisão, de 0,03hpa, por ter um consumo muito baixo, e principalmente por sua interface de comunicação de dados se dar através de um protocolo presente no microcontrolador.



Figura 9 – Placa de circuito impresso com o sensor BMP085.

Fonte: Sensortec (2009).

3.1.3 Anemômetro e Biruta

O anemômetro usado é do modelo três conchas, conforme mostra a Figura 10. O tubo principal deste instrumento é feito com alumínio marítimo, e as conchas são feitas de PVC (*Polyvinyl chloride*) com eixos de alumínio e seu rolamento interno é selado. Ao eixo é acoplado um *reed_switch* comutado por um ímã preso à parte móvel que gira com o vento, podendo-se assim fazer a contagem de tempo entre as voltas. Foi utilizado este anemômetro para fazer as medidas de velocidade do vento por ser o mais barato encontrado no mercado.



Figura 10 – Anemômetro.

Fonte: Autoria Própria.

A biruta é do tipo cone e construída de *nylon* como mostra a Figura 11. A abertura frontal possui 16cm de diâmetro, a abertura traseira tem 8cm, e seu comprimento é de 60cm.

3.1.4 Kit MSP430 *Launchpad*

O kit MSP430G2 *LaunchPad* usado neste projeto, é uma ferramenta de fácil utilização desenvolvida pela empresa *Texas Instruments*, cujo seu principal objetivo é o interfaceamento e depuração do microcontrolador com o programa usado para programação do mesmo. Este kit possui suporte para os dispositivos MSP430G2xx e MSP430F20xx (PDIP14 e PDIP20), a comunicação serial UART é feita via interface de programação, possui dois LED's (*Light Emitting Diode*) conectados aos pinos de I/O para uso geral, e dois botões do tipo *push-botton* (TEXAS, 2010). Foi utilizado esse kit por conta de seu baixo custo.



Figura 11 – Biruta.
Fonte: Autoria Própria.

3.1.5 Microcontrolador MSP430G2553

Os microcontroladores da família MSP430, fabricados pela *Texas Instruments*, são desenvolvidos especialmente para baixo consumo de energia. A CPU é do tipo RISC (*Reduced Instruction Set Computer*) de 16 bits com arquitetura Von-Neumann. Ainda conta com um sistema de *clock* flexível e vários tipos de

periféricos analógicos e digitais. O encapsulamento possui 20 pinos, sendo que 16 são de I/O, divididos nas portas 1 e 2.

Foi escolhido para uso neste projeto por se já ter conhecimento sobre este microcontrolador, sendo as que principais características do MSP430G2553 são (TEXAS, 2011):

- Tensão de alimentação entre 1,6 e 3,6V;
- Possui 2 *timers* de captura e comparação;
- 16 KB de memória flash;
- 512 Bytes de memória RAM;
- 2 interfaces de comunicação serial:
 - Interface USCI0: UART, LIN, IrDA, SPI;
 - Interface USCI1: SPI, I²C.
- Conversor A/D de 10 bits;
- Comparador analógico.

3.1.6 Ferramenta de Desenvolvimento de Código *Code Composer Studio*

O *Code Composer Studio*, cuja tela principal é mostrada na Figura 12 é um ambiente de desenvolvimento integrado desenvolvido pela *Texas Instruments*, que é utilizado para desenvolver e depurar programas para sistemas embarcados. Possui bibliotecas para cada um dos microcontroladores da família MSP430, editor de código fonte, ambiente para construção de projetos, ambiente de depuração entre outros recursos (TEXAS, 2013).

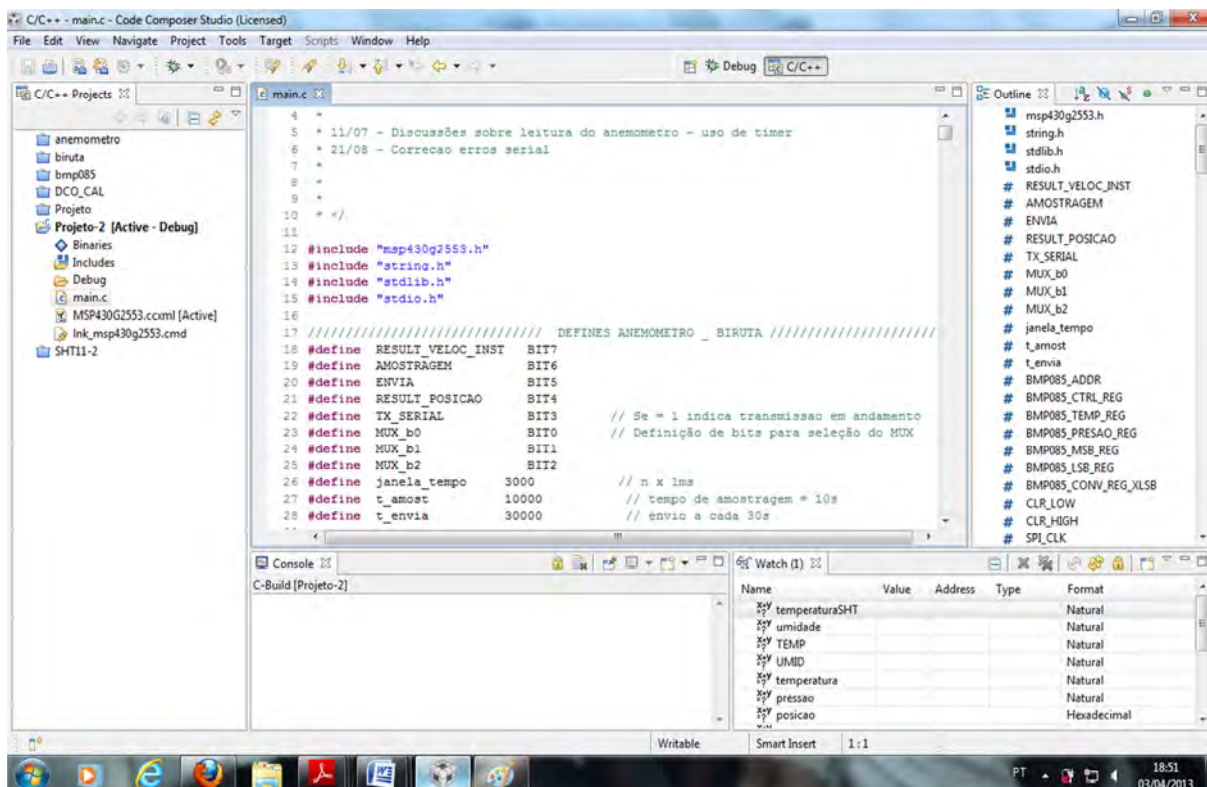


Figura 12 - Tela principal do *Code Composer Studio*.
Fonte: Autoria Própria.

3.1.7 Aplicativo AcessPort

Para apresentação dos dados ao usuário foi usado o programa *AcessPort*, que é um terminal de comunicação serial para computador. Segundo (SUDT, 2009), as principais características do *AcessPort* são:

- Suporte para todos os formatos de dados (ASCLi, HEX) e diferentes tamanhos de palavras;
- Permite comunicação Full Duplex;
- Suporta taxas de transmissão entre 110 a 256000bps;
- Pode trabalhar com as seguintes portas COM:
 - Portas COM padrão, *on-board*;
 - Placas de extensão de portas;
 - Portas COM conectadas ao computador através de USB (emuladas).

3.2 METODOLOGIA

Na metodologia serão abordados os métodos utilizados no desenvolvimento do presente trabalho, envolvendo os circuitos e protocolos de comunicação entre o microcontrolador e sensores utilizados, o desenvolvimento do circuito de medição da direção do vento, de alimentação e de comunicação do sistema com dispositivo externo.

3.2.1 Circuitos de leitura do Anemômetro e da Biruta

A velocidade do vento é medida através de um anemômetro de três conchas. Em seu eixo foi acoplado um sensor magnético do tipo *reed-switch*, conectado a uma porta do microcontrolador, a qual foi configurada como entrada, e com resistor de *pull-up* ativado. Quando o ímã está posicionado sobre o sensor, o contato deste fecha, conectando o polo de sinal no GND. Desta forma, pulsos são gerados com período que é função da velocidade de giro do anemômetro. A Figura 13 mostra a conexão do *reed switch* do anemômetro ao microcontrolador.

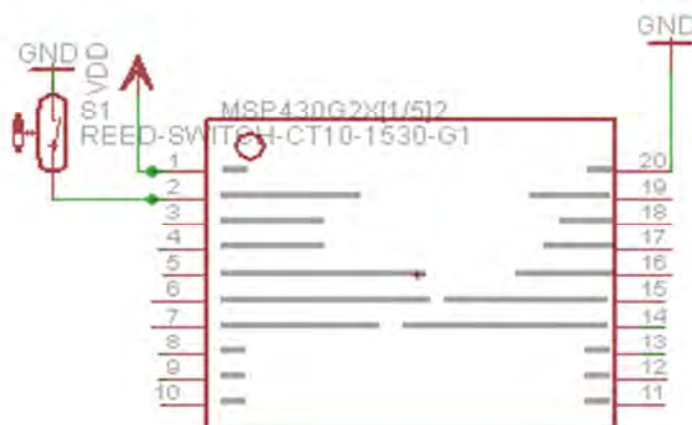


Figura 13 - Circuito Anemômetro.
Fonte: Autoria própria.

No código desenvolvido para o microcontrolador foi ativada a detecção de borda de descida para o bit da porta a qual esta ligado o *reed-switch*. Quando é gerada uma borda de descida o programa entra na rotina de tratamento de interrupção da porta, onde o Timer1 é ativado para avaliar intervalo de tempo entre

duas bordas de descida e, conseqüentemente, é possível relacionar este período de tempo com a velocidade do vento através de (SILVA D. C., 2013):

$$V=2.\pi.r^2/T \quad (3)$$

onde r é o raio e T é o tempo entre uma borda e outra.

Para a medição da direção do vento, usou-se oito sensores magnéticos do tipo *reed-switch*, um para cada ponto cardeal, comutados através de um ímã conectado à biruta. Conforme o vento muda de direção a biruta muda também, comutando o *reed-switch* correspondente ao seu ponto cardeal. Esses oito sensores são ligados a um multiplexador 74HC151 de 8 canais como mostra Apêndice 1, que controlado pelo microcontrolador faz a leitura de cada um dos sensores.

Para realizar a leitura dos sensores da biruta, o microcontrolador manda oito seqüências de 3 bits de controle para o multiplexador como mostra Tabela 2, e tem o retorno de qual *reed-switch* está ativo, podendo assim identificar a direção do vento.

Tabela 2 - Tabela verdade do multiplexador

| Entradas | | | Saída |
|----------|---|---|-------|
| C | B | A | Y |
| X | X | X | L |
| L | L | L | D0 |
| L | L | H | D1 |
| L | H | L | D2 |
| L | H | H | D3 |
| H | L | L | D4 |
| H | L | H | D5 |
| H | H | L | D6 |
| H | H | H | D7 |

Fonte: Autoria própria.

3.2.2 Sistema Microcontrolado

O sistema microcontrolado faz todo o controle da estação, alimentação dos circuitos bem como a leitura dos sensores SHT11, BMP085 e a comunicação com sistema de medição de velocidade e direção do vento. Além disso o sistema microcontrolado é responsável pelo envio de dados para um computador através da interface de comunicação serial.

3.2.2.1 Interfaceamento com sensores

A comunicação do sistema microcontrolado com o sensor BMP085 usado para medição de pressão atmosférica e temperatura ocorre através da interface serial presente no microcontrolador, a qual é configurada para trabalhar no modo I²C, atuando como mestre. Para comunicação com o sensor, os bytes enviados devem ser carregados no *buffer* de transmissão UCB0TXBUF e devem seguir a sequência mostrada na Figura 14.

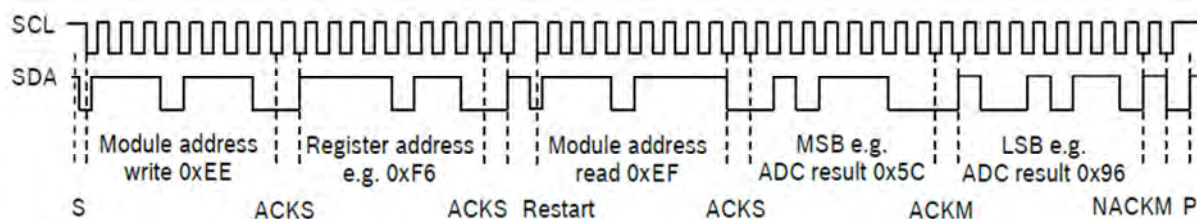


Figura 14 - Sequência da comunicação I²C.
Fonte: Sensortec (2009).

A sequência de comandos para obter temperatura e pressão atmosférica será descrita a seguir. Primeiro é enviado pelo microcontrolador o endereço do sensor, que é 0x77, para estabelecer a comunicação. Os primeiros dados adquiridos do sensor são os de calibração, enviando os endereços dos registradores e armazenado em variáveis, como mostra a Figura 15.

| | BMP085 reg adr | |
|------------------|-----------------------|------------|
| Parameter | MSB | LSB |
| AC1 | 0xAA | 0xAB |
| AC2 | 0xAC | 0xAD |
| AC3 | 0xAE | 0xAF |
| AC4 | 0xB0 | 0xB1 |
| AC5 | 0xB2 | 0xB3 |
| AC6 | 0xB4 | 0xB5 |
| B1 | 0xB6 | 0xB7 |
| B2 | 0xB8 | 0xB9 |
| MB | 0xBA | 0xBB |
| MC | 0xBC | 0xBD |
| MD | 0xBE | 0xBF |

Figura 15 - Registradores de Calibração do BMP085.

Fonte: Sensortec (2009).

Após obter os valores de calibração, é necessário adquirir os valores de referência dos registradores de temperatura e pressão atmosférica. Isto é feito enviando os endereços dos registradores para o sensor. Primeiramente é enviado o endereço do registrador de controle (0xF4), depois é enviado o endereço do registrador de temperatura (0x2E), então o sensor envia os dados que são recebidos pelo microcontrolador através do buffer de recepção UCB0RXBUF e são armazenados na variável UT do programa. O mesmo procedimento é usado para pressão atmosférica, enviando para o sensor, (0xF4) endereço do registrador de controle e (0x43) endereço do registrador de pressão, sendo que o resultado é armazenando na variável UP.

Com os dados de referência contidos em UP e UT, e os valores de calibração, é possível calcular os valores de temperatura e pressão atmosférica através da sequência de equações ilustradas na Figura 16.

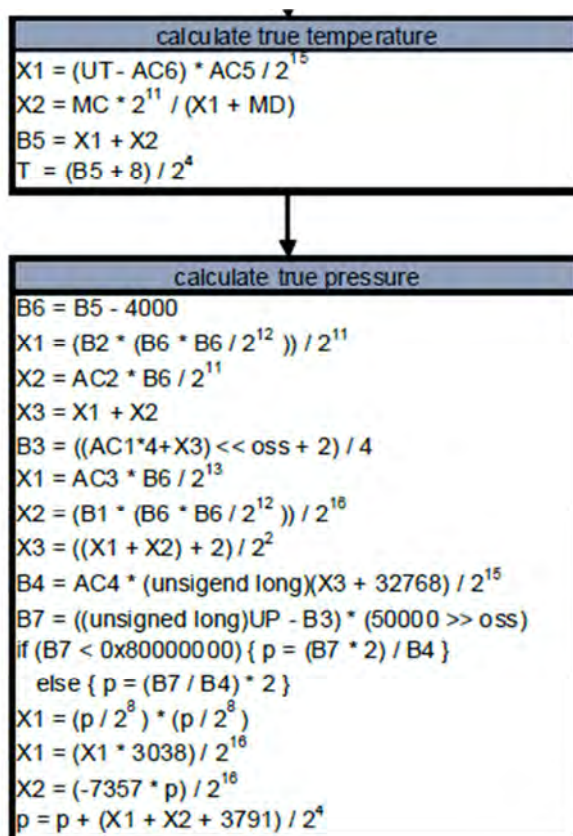


Figura 16 – Procedimento para obter a temperatura e pressão atmosférica do sensor BMP085.

Fonte: Sensortec (2009).

Para que o sistema microcontrolado possa fazer a comunicação com o sensor SHT11 empregado para medir temperatura e umidade relativa do ar, foi criado um protocolo com base no *datasheet* deste sensor. Seguindo a sequência mostrada na Figura 17, os dados são enviados bit-a-bit através de pinos de I/O do microcontrolador, seguindo uma lógica implementada através de uma máquina de estados situada na rotina de tratamento de interrupção do TimerA, conforme código apresentado no Apêndice 2.

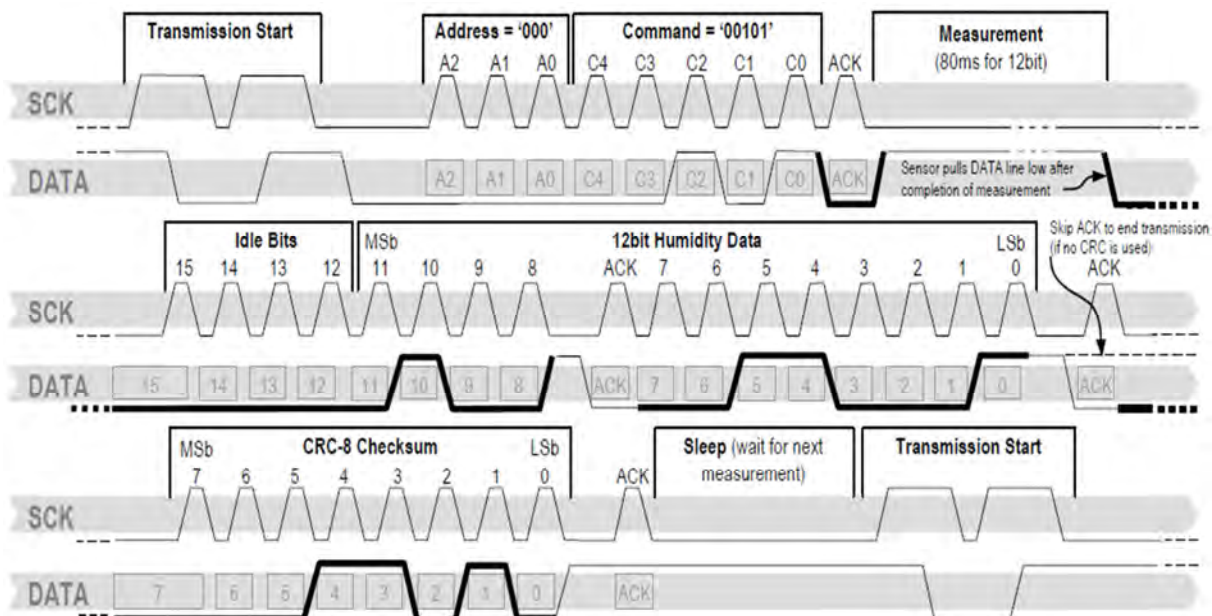


Figura 17 – Sequência de bits para comunicação com SHT11.
Fonte: Sensirion(2011).

De acordo com a Figura 17, a sequência de comunicação começa com o envio de um sinal de *start*, que consiste de uma borda de descida e uma de subida no nível alto de dois pulsos de *clock*. Em seguida é enviado um byte contendo o endereço do sensor e o código de comando, especificando ao sensor se é uma leitura de temperatura ou umidade relativa do ar. Os três primeiros bits deste *byte* são de endereço e os outros cinco bits para comando. Após envio do byte o mestre verifica se o escravo recebeu este byte através do bit de reconhecimento ACK, o qual é enviado pelo sensor que sinaliza na linha de dados com nível baixo no pulso de *clock* seguinte. Após o recebimento da confirmação, é preciso esperar um determinado tempo de 20, 80 ou 320 milisegundos para medir 8,12 ou 14bits respectivamente, da resposta de sensor, sendo o primeiro byte mais significativo (MSB- *Most Significant Byte*), seguido por um bit de ACK e o byte menos significativo (LSB- *Least Significant Byte*). Como não foi realizado o *checksum* após recebimento dos bytes, o mestre não efetua o reconhecimento através de um bit de ACK, sinalizando com a linha de dados em nível alto no pulso de *clock* subsequente.

A leitura de temperatura do sensor SHT11 é feita enviado um byte contendo o endereço do sensor (0x00) e o comando (0x03), Após receber o valor de leitura digital, calcula-se o valor de temperatura usando os coeficientes para conversão mostrados na Figura 18 e a partir de (SENSIRION, 2011):

$$T=d1+d2.SOt \quad (4)$$

onde T é a temperatura medida, $d1$ e $d2$ são coeficientes mostrados na Figura 18 e SOt é o valor de referência adquirido no sensor.

| VDD | $d_1(^{\circ}\text{C})$ | $d_1(^{\circ}\text{F})$ | SO_T | $d_2(^{\circ}\text{C})$ | $d_2(^{\circ}\text{F})$ |
|------|-------------------------|-------------------------|--------|-------------------------|-------------------------|
| 5V | -40.1 | -40.2 | 14bit | 0.01 | 0.018 |
| 4V | -39.8 | -39.6 | 12bit | 0.04 | 0.072 |
| 3.5V | -39.7 | -39.5 | | | |
| 3V | -39.6 | -39.3 | | | |
| 2.5V | -39.4 | -38.9 | | | |

Figura 18 - Coeficientes de conversão de temperatura para o sensor SHT11.
Fonte: Sensirion(2011).

Para leitura de umidade relativa do ar é usado o mesmo procedimento, enviando o byte com endereço (0x00) e comando (0x05). Fazendo os cálculos a partir dos coeficientes de conversão, mostrados na Figura 19 e de (SENSIRION, 2011).

$$RHlinear=C1+C2.SORH+C3.SORH^2 \quad (5)$$

$$RHtrue=(T^{\circ}\text{c}-25).(t1+t2.SORH)+RHlinear \quad (6)$$

sendo $C1$, $C2$, $C3$, $t1$ e $t2$ os coeficientes apresentados na Figura 19, $SORH$ é o valor de referencia adquirido no sensor e $T^{\circ}\text{c}$ é a temperatura obtida por meio de (4).

| SO _{RH} | C ₁ | C ₂ | C ₃ |
|------------------|----------------|----------------|----------------|
| 12 bit | -2.0468 | 0.0367 | -1.5955E-6 |
| 8 bit | -2.0468 | 0.5872 | -4.0845E-4 |

| SO _{RH} | t ₁ | t ₂ |
|------------------|----------------|----------------|
| 12 bit | 0.01 | 0.00008 |
| 8 bit | 0.01 | 0.00128 |

Figura 19 - Coeficientes para conversão de umidade relativa do ar do sensor SHT11.

Fonte: Sensirion(2011).

3.2.2.2 Circuito de interfaceamento serial

A comunicação do sistema microcontrolado com o dispositivo externo usado para a visualização dos dados ocorre através de uma interface serial padrão RS-232, configurada para trabalhar no modo UART, com *baud rate* de 9600, 8 bits de dados, um *stop* bit e sem o uso bit de paridade.

A Figura 20 mostra o circuito de conversão de sinais para o padrão RS-232, utilizando o circuito integrado MAX3222, o qual é ligado ao microcontrolador e ao conector DB-9 para conexão, através de cabo, ao computador. Os capacitores utilizados para o MAX3222 foram escolhidos com base no manual de instruções do fabricante.

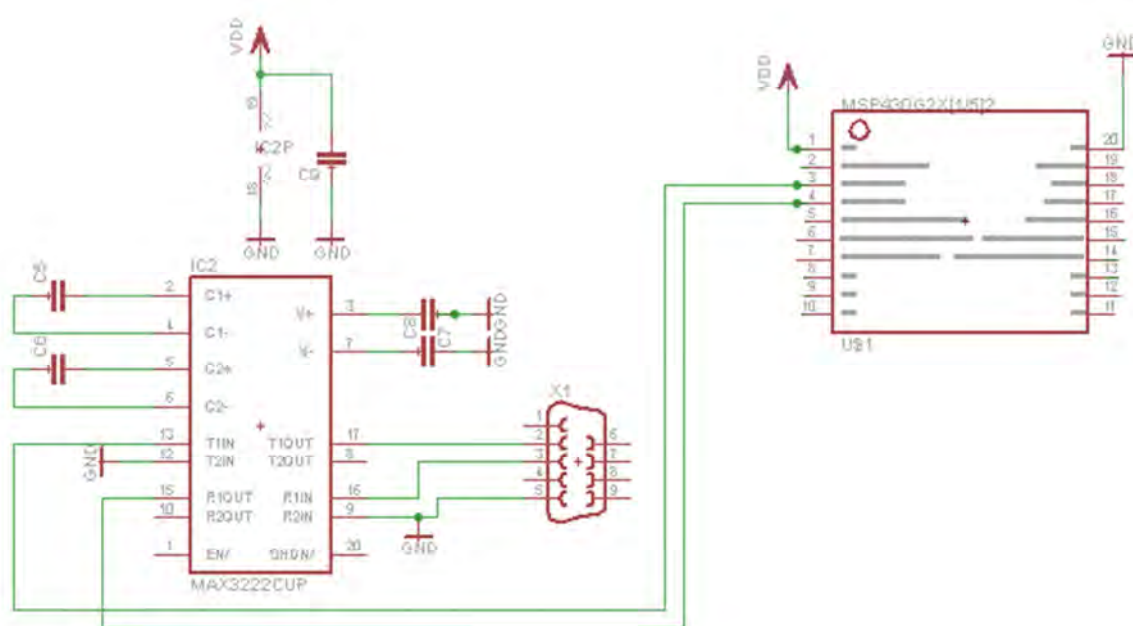


Figura 20 – Circuito de interface serial com o computador.
Fonte: Autoria própria.

3.2.2.3 Fonte de alimentação

Para alimentação do sistema e dos sensores, foi utilizada uma fonte de telefone celular que se conecta na placa através de um conector do tipo DCJ. A fonte alimenta placa com uma tensão de 9V, mas como o circuito precisa de uma tensão de entre 1,8V e 3,6V para funcionar, foi usado o regulador de tensão LM1117-3.3, conforme mostra Figura 21, para fornecer uma tensão regulada em 3.3V.

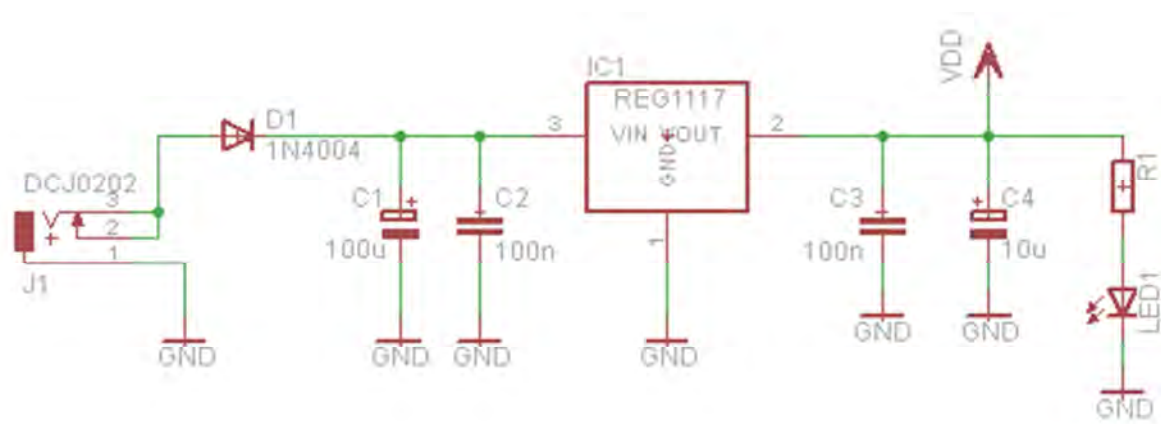


Figura 21 - Circuito Fonte de Alimentação.
Fonte: Autoria própria.

3.2.2.4 Periféricos utilizados e configurações

O microcontrolador foi configurado para trabalhar com uma frequência de *clock* de 16MHz. As configurações de inicialização da CPU podem ser vistas no Apêndice 3. O código de configuração da porta 1 pode ser visto no Apêndice 4, onde P1.0 é usada para leitura de velocidade do vento, configurada como entrada com resistor de *pull-up* ativado e interrupção por borda de descida ativa. Os pinos P1.1 e P1.2 são utilizadas para comunicação serial RXD e TXD ligadas ao MAX3222, enquanto P1.3 é configurada como entrada para leitura de direção do vento. Os pinos P1.6 e P1.7 foram usadas para comunicação com o sensor BMP085 através de uma interface serial configurada para o modo I²C.

Os pinos P2.0, P2.1, P2.2 da porta 2 conforme as configurações apresentadas no Apêndice 4 são configurados como saída e usadas para envio do comando para o sistema de medição da direção do vento, enquanto P2.3 e P2.4

fazem a comunicação com o sensor SHT11, e P2.5 configurada como saída para realizar a operação de *clear* no sensor BMP085.

Os dois módulos temporizados (TimerA0 e TimerA1) presentes no microcontrolador, foram utilizados e suas configurações são apresentadas no Apêndice 5. O *TimerA0* configurado para operar no modo crescente e gerar interrupção a cada 10 μ s, sendo usado implementação do protocolo de comunicação com o sensor SHT11. O *TimerA1* também configurado para modo crescente para gerar interrupções a cada 1ms, sendo empregado para uma base de tempo para as demais funções do programa.

4. RESULTADOS

A estação meteorológica desenvolvida é mostrada na Figura 22, sendo composta por quatro módulos dispostos em uma haste feita de ferro. No topo, a placa que efetua a leitura da direção do vento é protegida por uma caixa de plástico vedada e acoplada a uma biruta. Logo abaixo o anemômetro foi fixado na mesma haste. Os sensores temperatura, umidade relativa do ar e de pressão atmosférica, são abrigados dentro de um cano de PVC. Por fim a placa do sistema microcontrolado foi inserida dentro de uma caixa plástica vedada, que se conecta aos sensores através de cabos e ao computador através do conector DB-9.

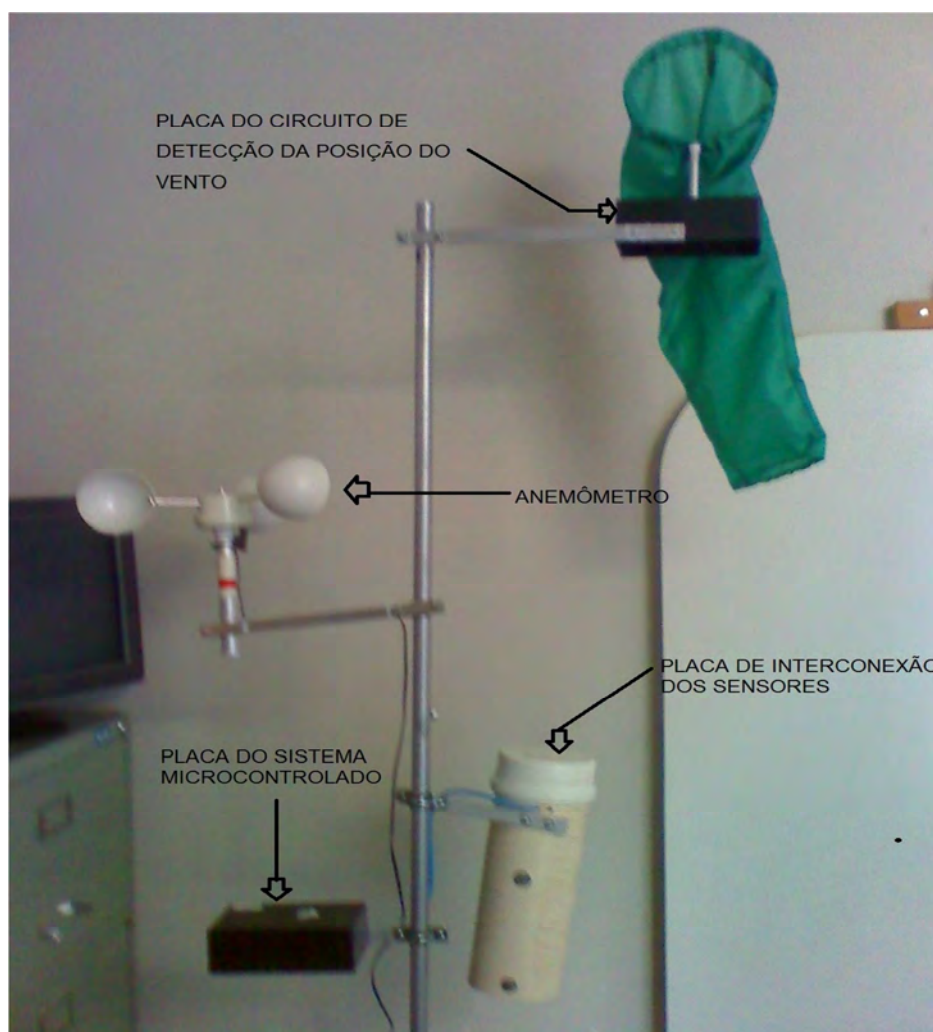


Figura 22 - Estação meteorológica desenvolvida.
Fonte: Autoria Própria.

4.1 PLACA DO CIRCUITO DE DETECÇÃO DA POSIÇÃO DO VENTO

O circuito de detecção da posição do vento foi implementado em uma placa de circuito impresso (PCI), como pode ser visto na Figura 23. Esta placa contém oito sensores magnéticos dispostos de modo que o ímã acoplado ao eixo da biruta fique posicionado sobre um dos sensores. Observa-se que o arranjo dos sensores e que o ímã tem um campo de ação que garante que apenas um dos *reed-switches* seja acionado em qualquer posição que o eixo se encontre. Além disso, a placa possui um soquete de 16 vias para encaixe do multiplexador 74HC151, um conector de pinos que faz a ligação com a placa do sistema microcontrolado, e um led para indicar que a placa está alimentada.

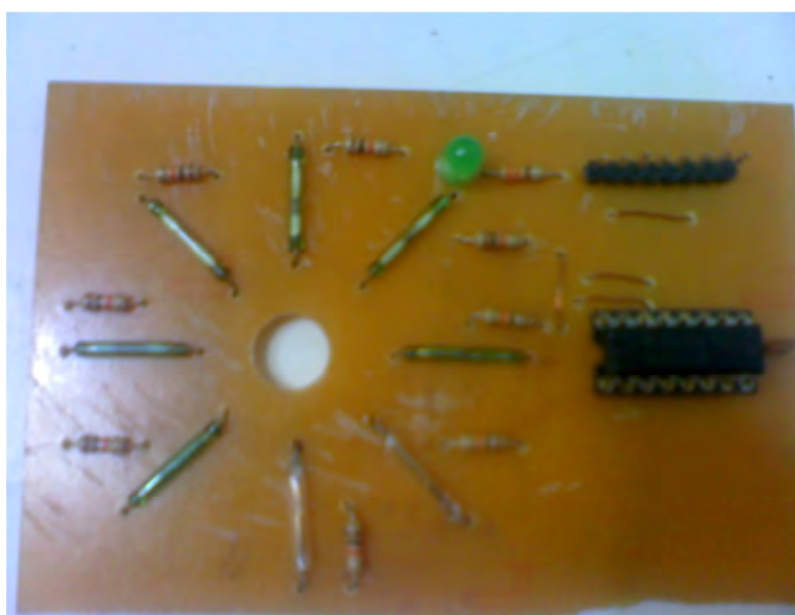


Figura 23 - Placa Indicadora de Posição do Vento.
Fonte: Autoria Própria.

4.2 PLACA DE INTERCONEXÃO DOS SENSORES

A Figura 24 mostra a placa para conexão dos sensores SHT11 e BMP085, a qual possui conectores barra de pinos tipo fêmea para o encaixe dos sensores, resistores de *pull-up* usados na linha de dados do SHT11, e linha de dados e de *clock* do BMP085. A placa fica no abrigo meteorológico indicado na

Figura 22. Portando para conexão da placa com o sistema microcontrolado é usado um conector barra de pinos macho e um cabo.

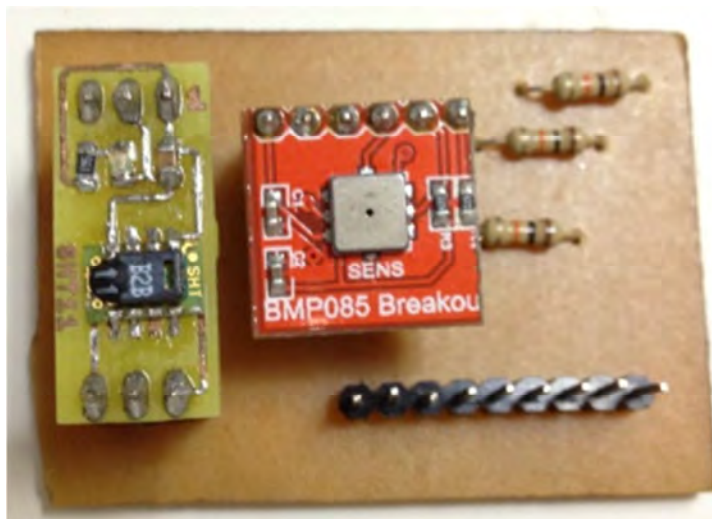


Figura 24 - Placa de conexão dos sensores SHT11 e BMP085.
Fonte: Autotia Própria.

4.3 PLACA DO SISTEMA MICROCONTROLADO

A placa do sistema microcontrolado, mostrada na Figura 25, integra todo o sistema, sendo responsável pela alimentação de todos os circuitos, abriga o microcontrolador que controla a estação, e provê uma interface de comunicação serial UART padrão RS-232 para conexão com computador para monitoramento dos dados meteorológicos pelo usuário através do terminal Access Port.

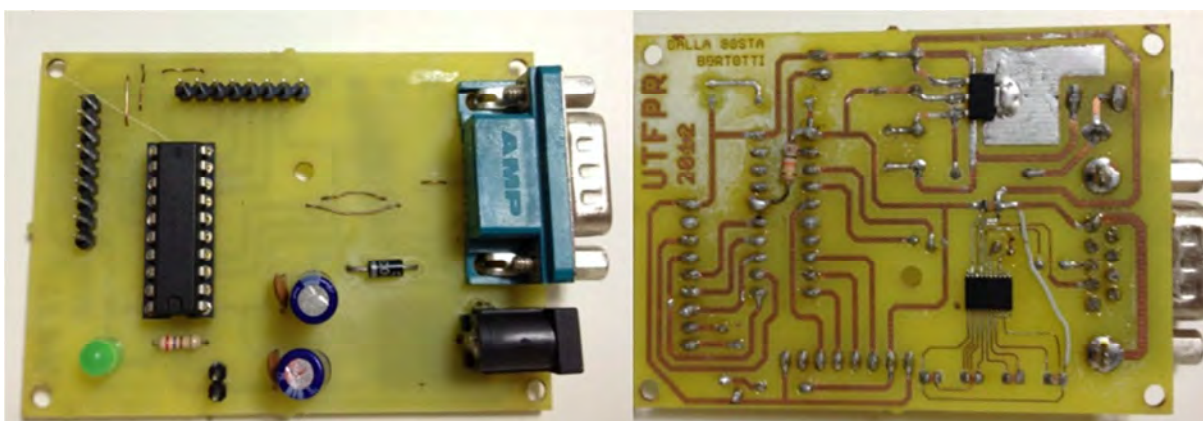


Figura 25 - Placa do sistema microcontrolado.
Fonte: Autotia Própria.

4.4 MEDIÇÕES DOS PARAMETROS METEOROLÓGICOS

Com todas as placas prontas e a estação meteorológica montada, foi possível testar a estação, medindo-se os parâmetros meteorológicos.

Nos testes foram feitas 24 medidas, uma para cada hora do dia, simulando o funcionamento normal de uma estação, variando dados climáticos, e lendo os valores de temperatura, pressão atmosférica, umidade relativa do ar, velocidade e direção do vento, dados estes mostrados na Figura 26, que apresenta uma tela do terminal de comunicação *AcessPort*, e mostrados em forma de gráficos pela Figura 27, Figura 28, Figura 29, Figura 30 e Figura 31.

| N | T-BMPO85[°c] | PRESSÃO[hPa] | VEL[km/h] | R.H[%] | T-SHT11[°c] | DIREÇÃO |
|----|--------------|--------------|-----------|--------|-------------|----------|
| 1 | 25,60 | 917,94 | 0 | 66,52 | 25,84 | norte |
| 2 | 25,60 | 918,10 | 2 | 66,92 | 25,75 | norte |
| 3 | 25,60 | 918,25 | 1 | 67,38 | 25,63 | norte |
| 4 | 25,60 | 918,22 | 2 | 67,66 | 25,48 | norte |
| 5 | 25,60 | 918,37 | 1 | 67,93 | 25,35 | nordeste |
| 6 | 25,60 | 918,43 | 0 | 68,30 | 25,20 | nordeste |
| 7 | 25,60 | 917,7 | 0 | 68,57 | 25,92 | leste |
| 8 | 27,20 | 915,40 | 0 | 65,23 | 27,84 | leste |
| 9 | 27,20 | 914,34 | 0 | 62,49 | 28,63 | sudeste |
| 10 | 27,20 | 913,32 | 0 | 59,24 | 29,41 | sudeste |
| 11 | 27,20 | 912,57 | 0 | 57,88 | 30,0 | sudeste |
| 12 | 28,80 | 911,51 | 0 | 55,6 | 30,98 | sudeste |
| 13 | 28,80 | 911,1 | 0 | 53,22 | 31,48 | sul |
| 14 | 28,80 | 910,78 | 0 | 53,74 | 30,71 | sul |
| 15 | 28,80 | 911,46 | 0 | 54,77 | 30,23 | sul |
| 16 | 28,80 | 912,7 | 0 | 55,54 | 29,66 | sudoeste |
| 17 | 28,80 | 912,34 | 0 | 56,56 | 29,46 | sudoeste |
| 18 | 28,80 | 908,78 | 0 | 56,88 | 29,73 | sudoeste |
| 19 | 30,30 | 908,29 | 0 | 57,0 | 29,81 | sudoeste |
| 20 | 28,80 | 910,32 | 0 | 60,18 | 28,93 | oeste |
| 21 | 28,80 | 911,46 | 0 | 69,79 | 28,71 | noroeste |
| 22 | 28,80 | 911,92 | 3 | 65,16 | 28,48 | noroeste |
| 23 | 27,20 | 912,32 | 0 | 69,25 | 27,96 | noroeste |
| 24 | 27,20 | 912,93 | 0 | 66,28 | 28,13 | noroeste |

Figura 26 - Tela de exposição dos dados ao usuário.
Fonte: Autoria Própria.

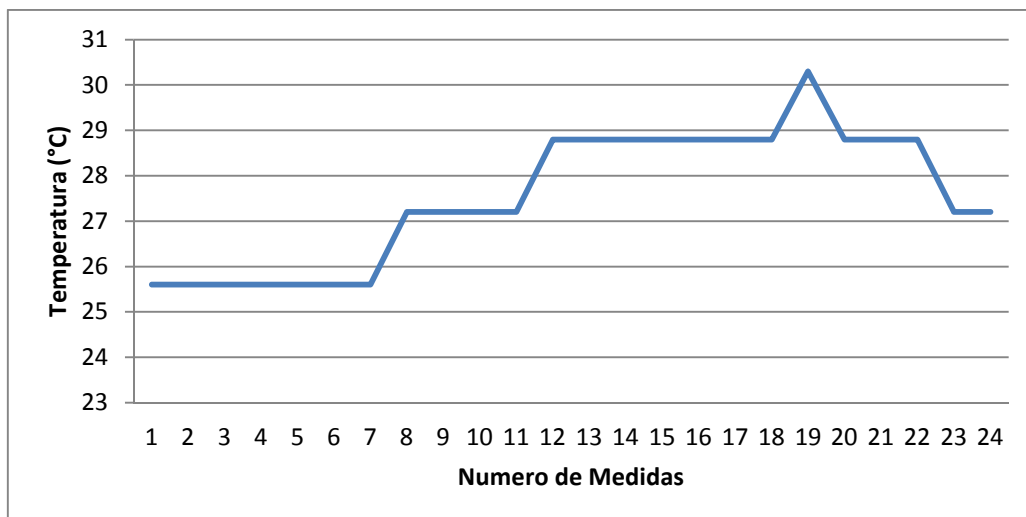


Figura 27 - Gráfico da temperatura do sensor BMP085.
Fonte: Autoria própria.

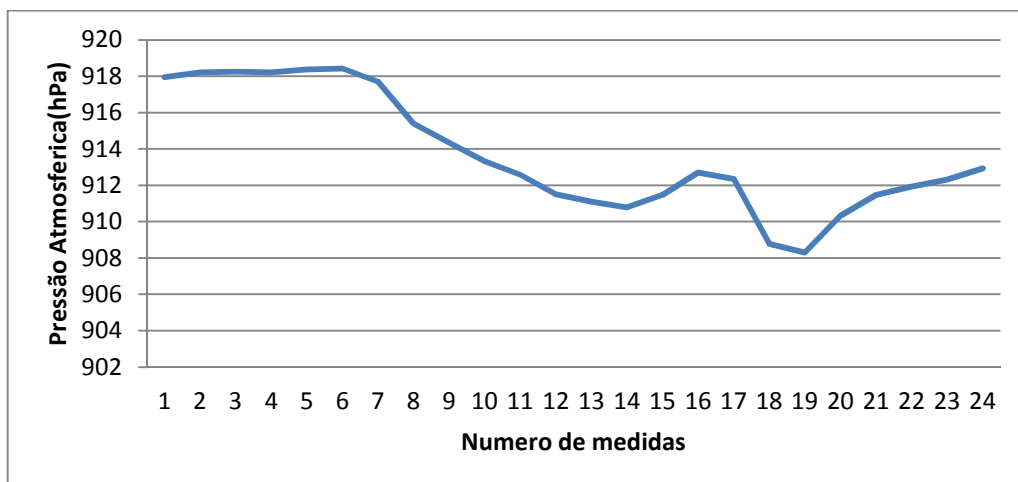


Figura 28 - Gráfico da pressão atmosférica.
Fonte: Autoria própria.

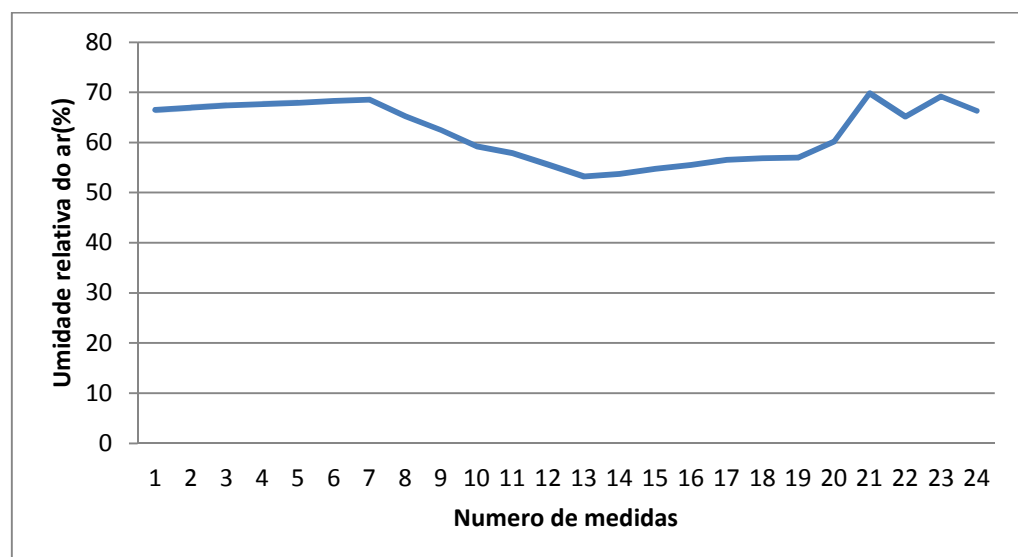


Figura 29 - Gráfico de umidade relativa do ar.
Fonte: Autoria própria.

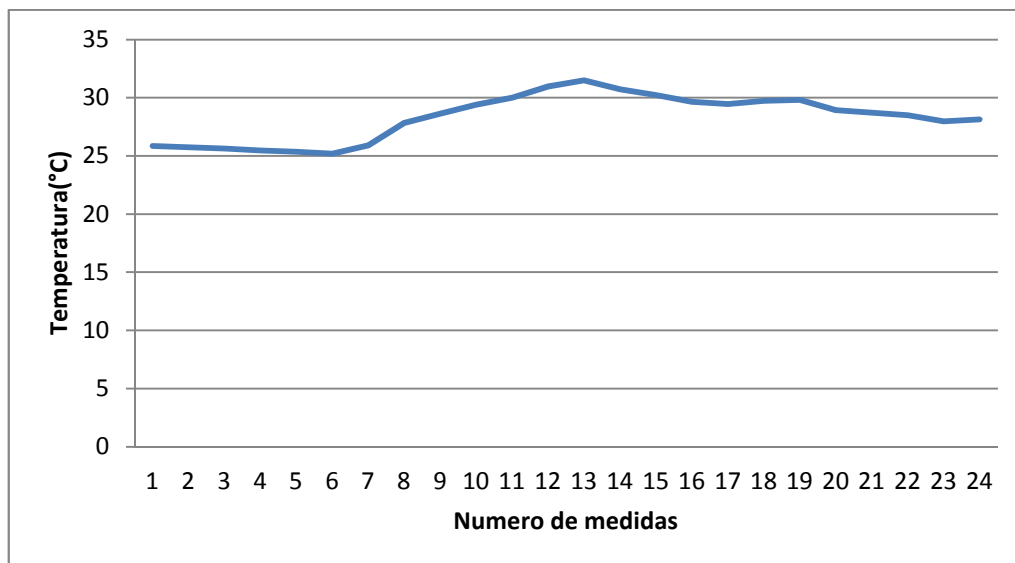


Figura 30 - Gráfico de temperatura do sensor SHT11.
Fonte: Autoria Própria.

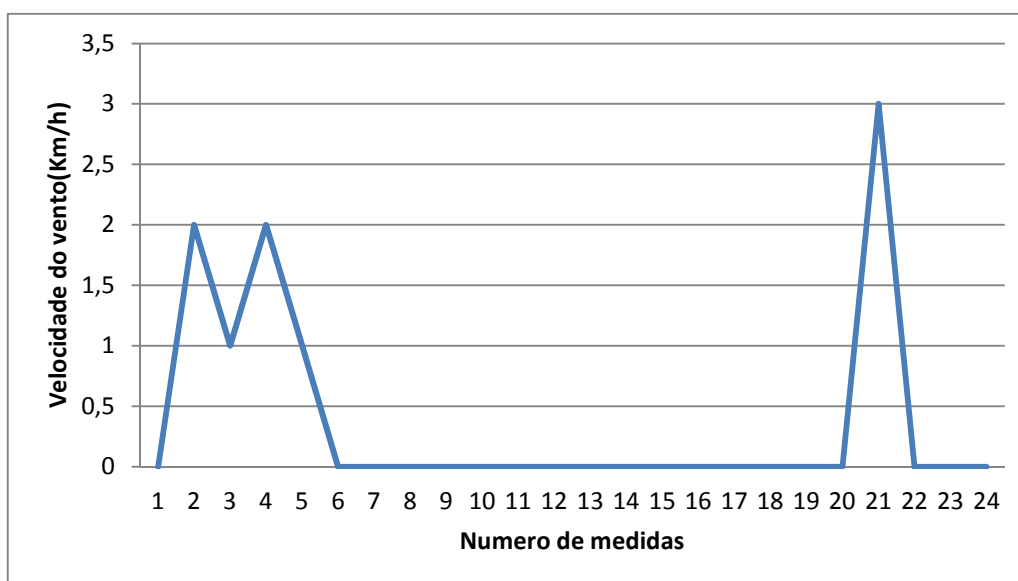


Figura 31 - Gráfico de velocidade do vento.
Fonte: Autoria própria.

4.5 CUSTOS DOS MATERIAS UTILIZADOS NA ESTAÇÃO

Os materiais utilizados, na grande maioria foram adquiridos no Brasil. Os sensores SHT11 e BMP085 foram importados, na Tabela 3 os valores destes estão corrigidos com as taxas de importação. De acordo com a Tabela 3, o preço total dos principais componentes utilizados na estação fica em torno 345 reais, abaixo do valor das estações meteorológicas vendidas no mercado.

Tabela 3 - Tabela de custos da estação.

| ITEM | DESCRIÇÃO | FORNECEDOR | QUANT. | VALOR UNIT | VALOR TOTAL |
|------------------|--|--------------------|--------|------------|-------------|
| Anemômetro | Medidor de Velocidade do Vento | Cup Anemômetros | 1 | R\$150,00 | R\$150,00 |
| BMP085 | Sensor de Pressão Atmosférica | Wide.hk | 1 | R\$59,55 | \$ 59,55 |
| Biruta | Medidor de Direção do Vento | Cup Anemômetros | 1 | R\$ 0,00 | R\$ 0,00 |
| Conector | DB-9 | Áudio K | 1 | R\$ 1,50 | R\$ 1,50 |
| Conector | DCJ | Áudio K | 1 | R\$ 2,30 | R\$ 2,30 |
| Fonte | Fonte Telefone Celular 9V | Eletrônica Guarani | 1 | R\$ 15,00 | R\$15,00 |
| LM1117-3,3 | Regulador de Tensão 3,3V | Farnel | 1 | R\$6,00 | R\$6,00 |
| MAX 3222 | Comunicação Serial | Farnel | 1 | R\$12,00 | R\$12,00 |
| Microcontrolador | MSP-430g2553 | Farnel | 1 | R\$ 9,00 | R\$9,00 |
| Placa fenolite | Placa fenolite 10x10 | Áudio K | 2 | R\$ 3,90 | R\$ 7,80 |
| Reed switch | Sensor Magnético | Áudio K | 8 | R\$2,40 | R\$19,20 |
| Resistor | Resistor 10k | Áudio K | 11 | R\$ 1,00 | R\$11,00 |
| SHT11 | Sensor de Temperatura e Umidade Relativa do ar | M Harte | 1 | R\$50,65 | R\$50,65 |
| 74HC151 | Multiplexador 8 Canais | Farnel | 1 | R\$1,00 | R\$1,00 |
| Valor Total | | | | | R\$345,00 |

Fonte: Autoria própria.

5. CONCLUSÕES

A partir do conhecimento das características de funcionamento dos sensores, seus protocolos de comunicação, tendo o domínio do microcontrolador e da linguagem de programação utilizada, foi possível desenvolver uma estação meteorológica de baixo custo para aquisição de temperatura, umidade relativa do ar, pressão atmosférica, velocidade e direção do vento.

As principais dificuldades encontradas durante o desenvolvimento da estação meteorológica foram os protocolos de comunicação dos sensores BMP085 e SHT11, além de dificuldades na programação do microcontrolador.

Como trabalhos futuros, pode-se desenvolver um sistema que faça a comunicação da estação com o microcomputador sem o uso de fios, e também acrescentar uma bateria com carregamento através de energia solar, permitindo o uso da estação em lugares remotos onde não há energia elétrica e cabos para comunicação de dados.

REFERÊNCIAS

ARAÚJO, Inês. Disponível em: <<http://vieiradasilva.host22.com/?p=423>> Acesso 27 de Outubro de 2012.

BRUSAMARELLO, V. **RTD's Termistores e Termopares** . Disponível em: <<http://www.chasqueweb.ufrgs.br/~valner.brusamarello/eleinst/ufrgs8.pdf>> Acesso 29 março de 2013.

CAETANO, Miguel José C. **I²C Verification suite**.2010. 110f. Dissertação (Mestrado Engenharia Electrotécnica e de Computadores) Faculdade de Engenharia. Universidade do Porto. Porto, Julho 2010.

CANZIAN, Edmur. **Comunicação Serial**. Disponível em: <http://www.capriconsultorios.com/Aula4-Comun_serial.pdf> Acesso em 27 de Fevereiro de 2013

CORTELETTI, D. **Introdução à programação de microcontroladores microchip PIC**. SENAI-RS, Outubro 2006.

CUNHA, Alessandro F. **MSP430- Liguagem C**. Disponível em: <<http://www.eletrica.ufpr.br/~rogerio/MSP430/00%20-%20CD%20DO%20ALUNO%20-%20MSP430%20RF/03%20-%20APOSTILAS/APOSTILA%20MSP430%20-%20C%20-%20PARTE%20IV.pdf>> Acesso em 05 de Abril de 2013.

DEMAR. Disponível em: <http://www.demar.eel.usp.br/electronica/aulas/Sensores_de_temperatura.pdf> Acesso em 27 de 03 de 2013.

ESPÍNDOLA, Water L. PIECARZ, Leandro S. **USBScope**. 2007. 62f. Monografia (Graduação em Engenharia Elétrica) Universidade Federal do Parana . Curitiba, 2007.

FRANÇA, José A. **Termopares**: Um Tutorial. Campina Grande, Agosto de 1997.

JUCA, Sandro. Instituto Federal de Educação, Ciência e Tecnologia. Disponível em: <<http://www.maracanau.ifce.edu.br/~sandrojuca/Microcontroladores1.pdf>> Acesso em 25 de Fevereiro de 2013.

MACIEL, Brain K; Soares, Luciane B & Aguiar, Ygor Q. **Barramento I²C e SPI**. Disponível em: <<http://www.ebah.com.br/content/ABAAe7xoAL/barramento-serial-spi-i-c>> Acesso em 25 de janeiro de 2013.

MOURA, Marcelo M. **Phoenix Observatorio Astronomico**. Disponível em: <http://www.observatorio-phoenix.org/k_ensaios/24_k16_c.htm> Acesso em 12 de outubro de 2012.

MULLER, Ivan. **Sistemas Instrumentados Portateis** : Esferas Instrumentadas para Sensoriamento de forças em Três Dimeções. 2008. 102f. Dissertação (Mestrado em Engenharia Eletrica) Escola de Engenharia Universidade Federal do Rio Grande do Sul. Porto Alegre 2008.

NARCISO, Denise. FALÇÃO, Flavia. NASCIMENTO, Leonardo. SILVA, Estelita. **Estudo de Comunicação Serial (UART)**. Disponível em: <<http://www.cin.ufpe.br/~sms/Arquivos/vt/ComunicacaoSerialXParalela.ppt>> Acesso em 29 de abril de 2013.

PEREIRA, Fábio. (2003). **Microcontroladores PIC- programação em c**. São Paulo: Érica, 2003.

SEIÇA, Alvaro. **Transdução: Processos de Transferência na Literatura e Arte**. 2011. 95f .Dissertação (Mestrado em Criações Literárias Contemporâneas) Universidade de Évora. Évora 2011.

SENSIRION. **Datasheet SHT1x (SHT10, SHT11, SHT15)**, Dezembro de 2011.

SENSORTEC, Bosh. **BMP085 Digital pressure sensor**. Reutlingen, Germany: Bosh ,Outubro de 2009.

SHINGUE, Carlos Y. **Eletronica Instrumentação**. Disponível em: <http://www.demar.eel.usp.br/eletronica/aulas/Sensores_de_temperatura.pdf> Acesso em 12 de Setembro de 2012.

SILVA, Paulo S. D. (2013). Portal da Universidade Estadual Paulista. Disponível em: <<http://www.dee.feb.unesp.br/~pss/1262/02%20-%20Linguagens%20de%20Alto%20Nivel%20-%20Fundamentos.pdf>> Acesso em 20 de Fevereiro de 2013.

SILVA, Domiciano C.M. **Mundo Educação- Trabalhos e pesquisas escolares**. Disponível em: <<http://www.mundoeducacao.com.br/fisica/velocidade-periodo-frequencia-no-mcu.htm>> Acesso em 14 de Março de 2013.

SILVA, Gustavo M. **Termopares.** Disponível em: <http://ltodi.est.ips.pt/gsilva/instrumentacao2/a_teoricas/1a-Termopares.pdf> Acesso em 10 de maio de 2012.

SUDT. **SUDT Software.** Disponível em: <<http://www.sudt.com/en/ap/index.html>> Acesso em 22 de Março de 2013.

TÉCNICAS e equipamentos para medição . Disponível em: <<http://ventos.pbworks.com/w/page/9917410/T%C3%A9cnicas%20e%20equipamentos%20para%20medi%C3%A7%C3%A3o>> Acesso em 05 de Abril de 2013

TEXAS INSTRUMENTS. **MSP-EXP430G2 LaunchPad Experimenter.** Dallas, texas, Estados Unidos, Julho de 2010.

_____. **MIXED SIGNAL MICROCONTROLLER.** Dallas, Texas, Estados Unidos, Abril de 2011.

_____. **Code Composer Studio (CCStudio):** Integrated Development Environment (IDE) v5. Disponível em: <<http://www.ti.com/tool/ccstudio#Technical%20Documents>> Acesso em 02 de Março de 2013.

TORRES, Eloize C. **Adaptação do Texto Clima e Agricultura.** IFF. 2013.

WENDLING, Marcelo. **Sensores.** Colegio Tecnico Industrial de Guaratingueta. Guaratingueta, 2010.

APÊNDICE 1

Circuito de medição de direção do vento

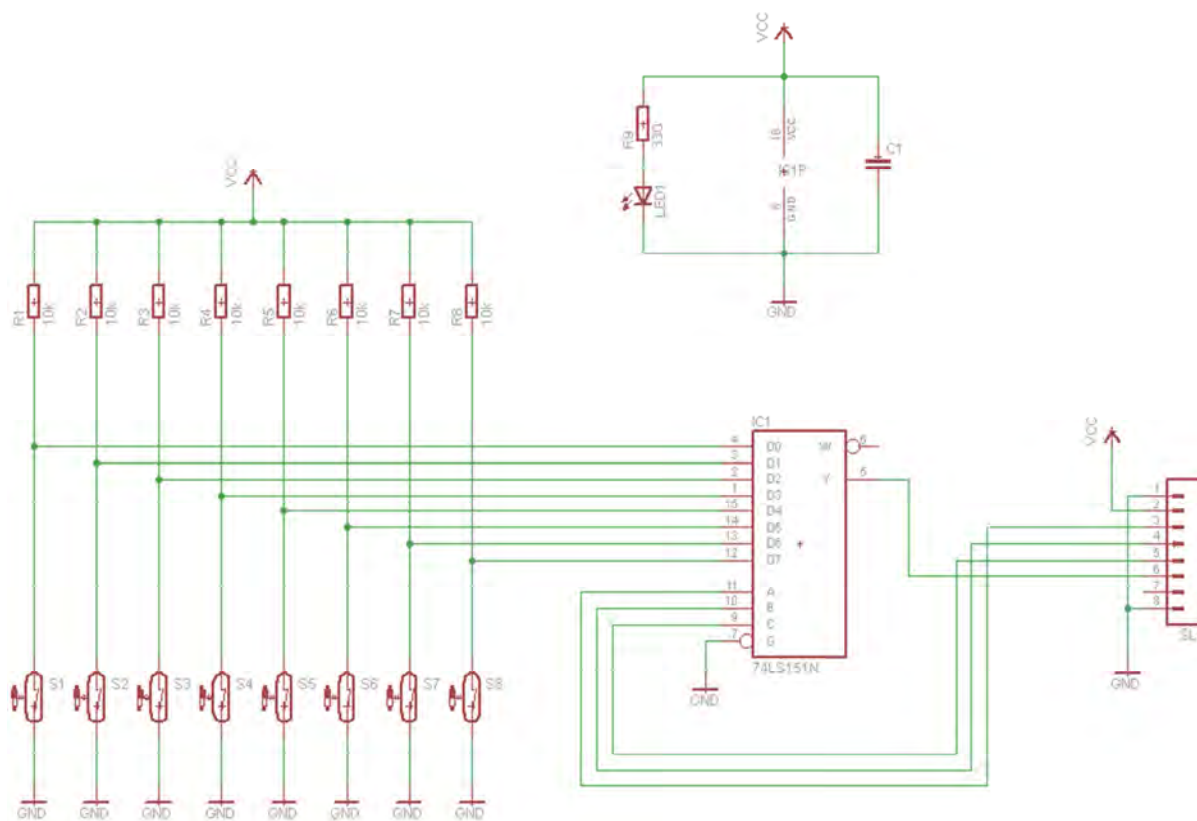


Figura 32 - Circuito de medição de direção do vento.
Fonte: Autoria própria.

APÊNDICE 2

Código desenvolvido para leitura do sensor SHT11

```

#pragma vector=TIMER0_A0_VECTOR
__interrupt void TA0CCR0_RTI(void){
    //P1OUT ^= BIT4;
    switch(ST){

    case 0: // gera o start

        switch(st_1){
            case 0:
                switch(st_2){

                    case 0:

                        cnt_bits = 9;
                        st_2 = 1;
                        break;

                    case 1:
                        SPI_CLK_L;
                        SPI_DATA_OUT;
                        SPI_DATA_H;

                        st_2= 2;
                        break;

                    case 2:
                        SPI_CLK_H;
                        st_2 = 3;
                        break;

                    case 3:
                        SPI_CLK_L;
                        st_2 = 4;
                        break;

                    case 4:
                        cnt_bits--;
                        if(cnt_bits > 0) st_2 = 1;
                        else{
                            st_1 = 1;
                            st_2 = 0;
                            cnt_bits=0;
                        }
                        break;

                }

            break;

            case 1:
                st_1 =2;
                break;
            case 2:
                //SPI_DATA_OUT;
                //SPI_DATA_H;
                SPI_CLK_L;
                st_1 = 3;
                break;
        }
    }
}

```

```

    case 3:
        SPI_CLK_H;
        st_1 = 4;
    break;
    case 4:
        SPI_DATA_L;
        st_1 = 5;
    break;
    case 5:
        SPI_CLK_L;
        st_1 = 6;
    break;
    case 6:
        st_1 = 7;
    break;
    case 7:
        SPI_CLK_H;
        st_1 = 8;
    break;
    case 8:
        SPI_DATA_H;
        st_1 = 9;
    break;
    case 9:
        SPI_CLK_L;
        st_1 = 10;
    break;
    case 10:
        //SPI_DATA_L;
        st_1 = 11;
    break;
    case 11:
        bit = BIT2;
        st_1 = 0;
        ST = 1; // aq
    break;
}
break;

case 1: // coloca endereço 000
    envia_byte(COMANDO);
break;

case 2: // Verifica ACK
    verifica_ACK();
break;

case 3: // TEMPO
    tempo();
break;

case 4: // RECEBE BYTE 1
    recebe_byte();
break;

case 5: // ARMAZENA MSB
    if(COMANDO == TEMPERATURASHT){
        TEMP = RX_DATA;
        TEMP = TEMP << 8;
    }
    else{

```

```
                UMID = RX_DATA;
                UMID = UMID << 8;
            }
            ST = 6;
break;

case 6: // ENVIA ACK
        envia_ACK();
break;

case 7: // RECEBE BYTE 2
        recebe_byte();
break;

case 8: // ARMAZENA LSB
        if(COMANDO == TEMPERATURASHT){
            TEMP |= RX_DATA;
        }else{
            UMID |= RX_DATA;
        }
        ST = 9;
break;

case 9: // ENVIA ACK
        envia_NACK();
break;

case 10: // Desabilita temporizador para encerrar aquisição
        TA0CTL &= ~MC0;
        FLAGS &= ~LEITURA_SHT11_OK;
        //FLAGS |= LEITURA_SHT11_OK;
        ST=0;
break;
}
}
```

APÊNDICE 3

Código desenvolvido para configuração da CPU

```
void ini_cpu(void){

    WDTCTL = WDTPW + WDTHOLD;    // Para WDT+
    // Configuracoes do sistema de clock
    // MCLK = 15 MHz, SMCLK = MCLK, ACLK = LFXT1 = 32.768 Hz (cristal)
    // DCOCTL |= DCO0 + DCO1;
    // BCSCTL1 |= XT2OFF + RSEL0 + RSEL1 + RSEL2 + RSEL3;
    //16Mhz
    BCSCTL1 = CALBC1_16MHZ;          // Set range
    DCOCTL = CALDCO_16MHZ;          // Set DCO step + modulation*/
    //BCSCTL3 |= XCAP0 + XCAP1 + LFXT1OF;
    while(BCSCTL3 & LFXT1OF);      // Sai do loop quando LFXT1 estiver estavel

    __enable_interrupt();          // Habilita interrupcoes
}
```

APÊNDICE 4

Código desenvolvido para configuração das portas 1 e 2

```
void ini_porta(void){  
  
    P1DIR|= BIT1 + BIT2 + BIT4 + BIT5 + BIT6 + BIT7;    //somente p1.0 e  
p1.3 como entrada.  
    P2DIR|= BIT0 + BIT1 + BIT2 + BIT5;  
    P1REN |= BIT0;// + BIT3;    // Resistor de Pull-up/Pull-  
down em P1.0  
    P1OUT |= BIT0;// + BIT3;    // Ativa resistor de Pull-up de  
P1.0  
    P1IES = 0x01;    // interrupção borda descida para P1.0  
    P1IFG &= ~BIT0;    // Garante que a flag P1IFG esteja resetada  
    P1IE  |= BIT0;    // Habilita a geração de interrupção de P1.0  
    //P1OUT &= ~(BIT4 + BIT5 + BIT6 + BIT7);  
    P1SEL = BIT1 + BIT2 + BIT6 + BIT7;  
    P1SEL2 = BIT1 + BIT2 + BIT6 + BIT7;  
}
```

APÊNDICE 5

Código desenvolvido para configuração dos *timers*

```
void ini_timer(void){
TA0CTL = TASSEL1; // + MC0; // clock: SMCLK ; Modo: Up
TA1CTL = TASSEL1 + MC0;
TA1CCR0 = 16000; // Contador ate 1ms
TA1CCTL0 = CCIE; // Habilita interrupcao para canal 0 (CCR0) do TA

TA0CCR0 = 160; // Contador ate 10 us /
TA0CCTL0 = CCIE; // Habilita interrupcao para canal 0 (CCR0) do TA
}
```