

**DISSERTAÇÃO DE MESTRADO PROFISSIONAL EM FÍSICA
COM ÊNFASE EM INSTRUMENTAÇÃO CIENTÍFICA**

**REGISTRADOR MULTIPROPÓSITO MODULAR
PARA MONITORAMENTO REMOTO**

RICARDO AFFONSO GRINGS HERBERT

CENTRO BRASILEIRO DE PESQUISAS FÍSICAS

**RIO DE JANEIRO
2009**

RICARDO AFFONSO GRINGS HERBERT

**REGISTRADOR MULTIPROPÓSITO MODULAR
PARA MONITORAMENTO REMOTO**

**Dissertação submetida à Coordenação de
Formação Científica do Centro Brasileiro de
Pesquisas Físicas como requisito parcial à
obtenção do título de mestre em física com ênfase
em instrumentação científica.**

Orientador: Prof. Dr. Geraldo Cernicchiaro

**RIO DE JANEIRO
2009**

[ficha catalográfica]

Aos carinhosos pais
Ivone e Bernardo,
que sucessivamente
obtiveram sucesso
na educação de seus
sete filhos.

AGRADECIMENTOS

Ao Geraldo Cernicchiaro, pela orientação desenvolvida nesta caminhada.

Ao Alexandre Benevento, pelo apoio e suporte indispensáveis ao desenvolvimento deste trabalho.

Aos professores do Mestrado Profissional em Física com ênfase em Instrumentação, Ademarlaudo Barbosa, Antônio Carlos Baratto, Geraldo Cernicchiaro, Herman Lima Júnior, Luís Sampaio, Marcelo Portes de Albuquerque e Márcio Portes de Albuquerque, pela dedicação com que investem no desenvolvimento deste país.

À Gil Souza, secretária do Departamento Física Aplicada do CBPF, que bravamente suportou e resolveu os quase quotidianos problemas trazidos por este que vos escreve.

À família do Laboratório de Instrumentação e Medição, Alexandre Toledo, Bruno Rente, Edivaldo Santos Filho, Eric Cruz, Erick Ortiz, Douglas Barros, Gabriela Negreiros, Gabriela Mello, Ihani Junger, Isabelle de Moraes, João Carlos Apolonio, Marcelo Simões e Rafael Barmak, Ricardo di Lucia, pelo auxílio prestado e por suportarem a tortura de serem obrigados a ouvir as minhas piadas endas.

À Raquel Correa, amiga de todas as horas, pela oportunidade de integrar o Programa Novos Tecnologistas do Instituto Nacional de Tecnologia e pela parcela de energia por ela irradiada que a mim chegou.

Aos funcionários da oficina mecânica do CBPF, em especial ao Eduardo Proença, Antônio Santana e Jefferson Molina, pelas peças e serviços indispensáveis ao desenvolvimento deste trabalho.

À Família Grings Herbert, unida ainda que dispersa.

Ao Instituto Nacional de Tecnologia, pelo suporte dado a esse trabalho e pela oportunidade de participar do Programa Novos Tecnologistas.

Ao Centro Brasileiro de Pesquisas Físicas, por me acolher em suas dependências e fornecer o ambiente e as condições propícias à progressão do trabalho e ao meu desenvolvimento profissional e pessoal.

RESUMO

Neste trabalho é apresentado o desenvolvimento de Hardware e Software de um Sistema Registrador Multipropósito Modular para monitoramento remoto de parâmetros físicos tais como temperatura, resistividade elétrica, turbidez, acidez, dentre outros, a partir de sensores e transdutores analógicos ou digitais. Sua principal aplicação seria em atividades de fiscalização e controle de poluição ambiental. O Sistema Registrador armazena os dados em dispositivo de memória. Podemos considerá-lo multipropósito devido à possibilidade de ser configurado para diversas aplicações, inclusive distintas do monitoramento ambiental. Sua arquitetura modular permite que módulos adicionais com funções específicas sejam desenvolvidos e incorporados. Em princípio, o sistema é composto pelos seguintes módulos: Módulo Alimentação, que fornece as tensões de operação dos demais circuitos a partir de bateria e/ou da rede elétrica; Módulo Controle, que gerencia a operação do sistema; Módulo Entrada, responsável pela digitalização dos sinais provenientes dos sensores; Módulo Condicionamento de Sinais, que processa analogicamente o sinal dos sensores adequando-os à faixa de tensão apropriada ao Módulo Entrada; Módulo Armazenamento, que armazena em cartão de memória não volátil os dados digitalizados dos sensores. Devido a considerações de segurança de dados e sigilo, o processamento do sistema é distribuído pelos diversos microcontroladores. Um protótipo foi construído e encontra-se em fase de testes no Laboratório de Instrumentação e Medidas do Centro Brasileiro de Pesquisas Físicas.

Palavras-chave: aquisição de dados, instrumentação eletrônica, microcontroladores, monitoramento remoto, sistemas embarcados.

ABSTRACT

This work presents the hardware and software development of a multipurpose modular data recorder system for remote measurements of physical parameters such as temperature, electric resistivity, turbidity, acidity, among others, acquired from analog or digital sensors and transducers. Its main application targets on activities of control and supervision of environmental pollution. The recorder system stores data into a nonvolatile memory device. It's multipurpose due to the possibility to be configured for several applications, including applications other than environmental monitoring. It's modular architecture allows the development and addition of new modules, carrying specific tasks. In principle, the system is composed of the following modules: Supply Module, which furnishes from a battery or mains supply the operating voltages needed by other modules; Control Module, which manages the system operation; Input Module, responsible for the digitization of sensors' signals; Signal Conditioning Module, which processes analog signals coming from sensors and adjusts to the voltage range accepted by the Input Module; Storage Module, which stores into a nonvolatile memory the digitized data from the sensors. Due to considerations of security and confidentiality, the systems' processing is distributed among microcontrollers. A working system was built and is currently under test procedures at the Instrumentation and Measurement Lab – LIM of the Brazilian Center for Physics Research – CBPF.

Keywords: data acquisition, embedded systems, electronic instrumentation, microcontrollers, remote monitoring.

LISTA DE ILUSTRAÇÕES

Figura 1-1 – Diagrama em blocos do Sistema Caipora.....	20
Figura 2-1 – Circuito para conexão de cristal ao microcontrolador.....	27
Figura 2-2 – Circuito equivalente de cristal oscilador.....	28
Figura 2-3 – Curva típica de variação da frequência de oscilação de um cristal com a temperatura.....	29
Figura 2-4 – Cartão de memória SD Card, tamanho padrão.....	32
Figura 2-5 – Estrutura em camadas do modelo OSI.....	37
Figura 2-6 – Quadro típico de dados (um <i>byte</i>) TIA/EIA-232.....	40
Figura 2-7 – Conexão entre dispositivos SPI.....	42
Figura 2-8 – Sequência referente ao dispositivo mestre escrevendo no dispositivo escravo...	43
Figura 2-9 – Sequência referente ao dispositivo mestre lendo do dispositivo escravo.....	44
Figura 2-10 – Saída em coletor aberto.....	45
Figura 2-11 – Conexão de múltiplas saídas em coletor aberto.....	46
Figura 2-12 – Arquitetura do conversor A/D do tipo aproximações sucessivas.....	48
Figura 2-13 – Circuito simplificado do pino de entrada analógica do microcontrolador.....	49
Figura 2-14 – Diagrama do sensor de temperatura TMP101.....	51
Figura 2-15 – Exemplo de circuito de referência de tensão do tipo <i>bandgap</i>	54
Figura 2-16 – Regulador ajustável de tensão MIC5205.....	55
Figura 3-1 – Ambiente de desenvolvimento Freescale CodeWarrior.....	58
Figura 3-2 – Depuração do programa executado no microcontrolador.....	59
Figura 3-3 – Foto da placa de desenvolvimento DEMO9S08QG8.....	62
Figura 3-4 – Tela do programa Eagle para desenho do diagrama esquemático.....	63
Figura 3-5 – Tela do programa Eagle para desenho da placa de circuito impresso.....	64
Figura 4-1 – Diagrama em blocos do Sistema Caipora.....	69
Figura 4-2 – Diagrama de derivação das tensões de alimentação.....	74
Figura 4-3 – Foto do Módulo Alimentação.....	74
Figura 4-4 – Fluxograma da sequência de inicialização do Sistema Caipora.....	75
Figura 4-5 – Circuito para o monitoramento da tensão da bateria.....	76
Figura 4-6 – Foto do Módulo Controle.....	78
Figura 4-7 – Sequência de inicialização do Módulo Controle.....	79
Figura 4-8 – Diagrama de transição de estados do processo de aquisição de dados.....	82

Figura 4-9 – Foto do Módulo Armazenamento de Dados.....	83
Figura 4-10 – Diagrama de fluxo da inicialização do cartão de memória.....	85
Figura 4-11 – Esquema de um canal do Módulo Condicionamento de Sinais.....	87
Figura 4-12 – Foto do Módulo Condicionamento de Sinais.	89
Figura 4-13 – Foto do Módulo Entrada.....	90
Figura 4-14 – Foto dos módulos conectados através da Placa de Interconexão.....	92
Figura 4-15 – Concepção da montagem do sistema.....	93
Figura 4-16 – Vista interna do Sistema Caipora.	94
Figura 4-17 – Foto do sistema operando conectado a um microcomputador.....	95
Figura 4-18 – Painel de configuração da interface serial.	96
Figura 4-19 – Painel de visualização dos dados.....	97
Figura 5-1 – Histograma da tensão de referência 1.65V.....	102
Figura 5-2 –Tensão de referência 1.65V, entre dias 21/set/2008 e 07/out/2008.....	102
Figura 5-3 –Tensão de referência 1,65V, dia 23/set/2008, 14h a 16h.....	103
Figura 5-4 –Tensão bandgap do Módulo Alimentação, dia 23/set/2008, 14h a 16h.....	105
Figura 5-5 –Tensão bandgap do Módulo Entrada, dia 23/set/2008, 14h a 16h.....	105

LISTA DE TABELAS

Tabela 2-1 – Comparação das características de microcontroladores.....	24
Tabela 2-2 – Estrutura dos comandos SD Card.....	33
Tabela 2-3 – Sinais da interface serial presentes no conector DE-9 – dispositivo DTE.....	40
Tabela 2-4– Alinhamento dos dados no registrador Temperatura do TMP101	52
Tabela 3-1– Funções disponibilizadas pelo módulo TimeDate do Codewarrior	61
Tabela 4-1– Comparação da função dos pinos do microcontrolador nos módulos Alimentação, Controle e Entrada.....	70
Tabela 4-2– Sinais do Barramento Central – conector de 40 pinos.....	72
Tabela 4-3 – Lista parcial dos comandos implementados no Módulo Controle.....	80
Tabela 4-4 – Lista parcial dos comandos implementados no Módulo Controle [continuação].	81
Tabela 4-5 – Faixas de tensão de entrada do Módulo Condicionamento de Sinais.....	88
Tabela 4-6 – Estimativa do consumo de energia.....	98
Tabela 5-1 – Resultados da caracterização do Módulo Condi. de Sinais – Modo tensão. ...	100
Tabela 5-2 – Resultados da caracterização do Módulo Condi. de Sinais – Modo corrente.	100
Tabela 5-3 – Frequências dos resultados da conversão da tensão de Referência 1,65V.....	101
Tabela 5-4 – Frequências dos resultados da conversão do divisor de tensão.....	104
Tabela 5-5 – Frequências dos resultados da conversão da referência interna bandgap do microcontrolador dos módulos Entrada e Alimentação.....	104
Tabela A.1 – Consumo de corrente Módulo Alimentação.....	117
Tabela A.2 – Consumo de corrente Módulo Controle.....	117
Tabela A.3 – Consumo de corrente Módulo Entrada.....	118
Tabela A.4 – Consumo de corrente Módulo Condicionamento de Sinais.....	118
Tabela A.5 – Consumo de corrente Módulo Armazenamento.....	119

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ADC	<i>Analog to Digital Converter</i> . Conversor analógico digital
ADE	Ato Declaratório Executivo
ASCII	<i>American Standard Code for Information Interchange</i>
ANEEL	Agência Nacional de Energia Elétrica
BDM	<i>Background Debug Module</i> . Módulo de depuração presente nos microcontroladores HCS08 do fabricante Freescale
CAD	<i>Computer Aided Design</i> . Projeto auxiliado por computador
CNC	<i>Computer Numeric Control</i> . Dispositivo de controle numérico
Cofis	Coordenação-Geral de Fiscalização da Secretaria da Receita Federal
CONAMA	Conselho Nacional do Meio Ambiente
CR	<i>Carriage Return</i> – 0x0D. Caractere não imprimível de retorno de linha
CRC	<i>Cyclic Redundancy Check</i> . Código de detecção de erros
DIP	<i>Dual Inline Package</i> – Tipo de encapsulamento para circuitos integrados
E/S	Entrada/Saída
ESR	<i>Equivalent Series Resistance</i> . Parâmetro que expressa a parte resistiva da impedância apresentada por um capacitor
FEEMA	Fundação Estadual de Engenharia do Meio Ambiente
I ² C	<i>Inter Integrated Circuit</i> . Barramento de comunicação entre circuitos integrados que utiliza apenas duas linhas de dados: temporização e entrada/saída de dados
IEC	<i>International Electrotechnical Commission</i>
INMETRO	<i>Instituto Nacional de Metrologia, Normalização e Qualidade Industrial</i> .
ISO	<i>International Organization for Standardization</i>
lsb	<i>Least Significant Bit</i>
OSI	<i>Open Systems Interconnection</i>
ppm	Parte por milhão
PI	Partículas Inaláveis
PTS	Partículas Totais em Suspensão
PRONAR	Programa Nacional de Controle da Poluição do Ar
RAM	<i>Random Access Memory</i> . Memória volátil.
SMV	Sistema de Medição de Vazão

SRAM	<i>Static RAM</i> . Memória volátil do tipo estática.
SRF	Secretaria da Receita Federal
SPI	<i>Serial Peripheral Interface</i> . Interface de comunicação serial síncrona que utiliza três linhas de dados: saída de dados, entrada de dados e temporização
USB	<i>Universal Serial Bus</i> . Interface de comunicação disponível em computadores

SUMÁRIO

1	INTRODUÇÃO.....	15
1.1	MOTIVAÇÃO DO TRABALHO	15
1.2	PARÂMETROS FÍSICOS, QUÍMICOS E BIOLÓGICOS.....	17
1.3	CARACTERÍSTICAS E FUNÇÕES DO SISTEMA CAIPORA.....	19
1.4	ORGANIZAÇÃO DO TRABALHO	20
2	FUNDAMENTOS TÉCNICOS E TEÓRICOS.....	22
2.1	MICROCONTROLADOR.....	22
2.1.1	<i>Configuração do microcontrolador.....</i>	<i>25</i>
2.1.1.1	Operação do microcontrolador a partir do oscilador interno	26
2.1.1.2	Operação do microcontrolador a partir de cristal externo	27
2.1.2	<i>Reinicialização do microcontrolador.....</i>	<i>29</i>
2.1.3	<i>Módulo de depuração do microcontrolador.....</i>	<i>30</i>
2.2	DISPOSITIVOS DE ARMAZENAMENTO	31
2.2.1	<i>Cartão de memória SD Card.....</i>	<i>31</i>
2.2.1.1	Acesso ao Cartão no Modo SD	32
2.2.1.2	Acesso ao Cartão no Modo SPI.....	33
2.2.1.3	Comandos de Acesso ao Cartão	33
2.2.1.4	Estrutura de Armazenamento de Dados no Cartão.....	34
2.2.1.5	Procedimento de Escrita de um Bloco no Cartão	34
2.2.1.6	Registradores internos do Módulo Armazenamento	34
2.2.1.7	Cálculo do Tamanho do Cartão.....	35
2.2.1.8	Detecção de Erros no Cartão	35
2.3	PROTOCOLOS DE COMUNICAÇÃO.....	36
2.3.1	<i>Protocolo de Camada Física EIA/TIA-232.....</i>	<i>39</i>
2.3.2	<i>Protocolo de Comunicação SPI.....</i>	<i>41</i>
2.3.3	<i>Protocolo de Comunicação I²C.....</i>	<i>42</i>
2.3.4	<i>Malha de corrente 4 a 20mA</i>	<i>46</i>
2.4	CONVERSOR ANALÓGICO-DIGITAL	47
2.4.1	<i>ADC do tipo aproximações sucessivas</i>	<i>47</i>
2.4.2	<i>ADC Interno do Microcontrolador.....</i>	<i>48</i>
2.5	TRANSDUTORES DE TEMPERATURA	50
2.5.1	<i>Sensor de Temperatura por Junção Semicondutora p-n</i>	<i>50</i>
2.5.2	<i>Sensor de Temperatura com interface I²C.....</i>	<i>51</i>
2.5.3	<i>Sensor de Temperatura Interno do Microcontrolador</i>	<i>53</i>
2.6	REFERÊNCIA DE TENSÃO DO TIPO BANDGAP	53
2.6.1	<i>Referência de Tensão Interna do Microcontrolador</i>	<i>54</i>
2.6.2	<i>Regulador de tensão com referência interna.....</i>	<i>54</i>
3	UTILITÁRIOS DE DESENVOLVIMENTO	57
3.1	COMPILADOR FREESCALE CODEWARRIOR	57
3.1.1	<i>Módulo I²C para Codewarrior</i>	<i>60</i>
3.1.2	<i>Módulo de Relógio de Tempo Real para Codewarrior.....</i>	<i>61</i>
3.1.3	<i>Programação do Microcontrolador</i>	<i>61</i>
3.2	EAGLE LAYOUT EDITOR.....	62
3.3	FRESADORA CNC	64
3.3.1	<i>Circuitcam.....</i>	<i>64</i>
3.3.2	<i>Boardmaster</i>	<i>65</i>
3.4	CYGBWIN	65
3.4.1	<i>Comando Hexdump</i>	<i>65</i>
3.4.2	<i>Comando dd.....</i>	<i>66</i>
3.4.3	<i>Comando tr.....</i>	<i>66</i>
3.5	LABVIEW.....	66

4	DESENVOLVIMENTO DO PROJETO	68
4.1	MÓDULO ALIMENTAÇÃO	73
4.1.1	<i>Monitoramento da Tensão da Bateria e da Alimentação +5V</i>	76
4.1.2	<i>Monitoramento da Temperatura Ambiente do Sistema</i>	77
4.1.3	<i>Hora do Sistema</i>	77
4.2	MÓDULO CONTROLE	78
4.2.1	<i>Comandos de Configuração e Controle do Sistema Caipora</i>	80
4.2.2	<i>Processo de Aquisição de Dados</i>	81
4.3	MÓDULO ARMAZENAMENTO DE DADOS	83
4.3.1	<i>Procedimento de Inicialização do Cartão de Memória SDCard</i>	84
4.3.2	<i>Estrutura do Bloco de Dados</i>	84
4.3.3	<i>Leitura do SDCard em um Microcomputador</i>	85
4.4	MÓDULO CONDICIONAMENTO DE SINAIS	86
4.5	MÓDULO ENTRADA	89
4.6	MÓDULO INTERFACE / MÓDULO TRANSMISSÃO / MODEM GPRS	91
4.7	MONTAGEM DO SISTEMA CAIPORA	92
4.8	PROGRAMA DE VISUALIZAÇÃO DE DADOS	95
4.9	CONSUMO DE ENERGIA	97
5	RESULTADOS E DISCUSSÕES	99
5.1	CARACTERIZAÇÃO DO MÓDULO CONDICIONAMENTO DE SINAIS	99
5.2	RESULTADOS DE MEDIÇÃO COM O SISTEMA	101
5.3	MEDIÇÃO DA ESTABILIDADE DO RELÓGIO DE TEMPO REAL	106
5.4	AUTONOMIA DO SISTEMA QUANDO OPERADO SOMENTE POR BATERIAS	106
6	CONCLUSÃO	108
	REFERÊNCIAS	112
	APÊNDICE A – ESPECIFICAÇÕES ELÉTRICAS	116
A.1	ALIMENTAÇÃO:	116
A.2	MÓDULO ENTRADA:	116
A.2.1	<i>ADC interno ao microcontrolador</i>	116
A.3	MÓDULO CONDICIONAMENTO DE SINAIS:	116
A.3.1	<i>Conexão dos sensores</i> :	116
A.4	MÓDULO ARMAZENAMENTO DE DADOS:	116
A.5	ESTIMATIVA DO CONSUMO DE ENERGIA	117
A.5.1	<i>Módulo Alimentação</i> :	117
A.5.2	<i>Módulo Controle</i> :	117
A.5.3	<i>Módulo Entrada</i> :	118
A.5.4	<i>Módulo Condicionamento de Sinais</i> :	118
A.5.5	<i>Módulo Armazenamento</i> :	119

1 INTRODUÇÃO

Esta iniciativa, a partir de parceria inter-institucional entre o Instituto Nacional de Tecnologia – INT e o Centro Brasileiro de Pesquisas Físicas – CBPF, visa fornecer ferramentas para análises quantitativas de parâmetros físico-químicos relevantes para fins de monitoramento e fiscalização de poluição ambiental. Para tanto, propõe-se sistema eletrônico destinado à aquisição, armazenamento e transmissão de dados resultantes destas medições: o Sistema Caipora. Um protótipo do Sistema Caipora foi construído e encontra-se operando em fase de testes. Os parâmetros medidos podem estar disponíveis sob forma de tensão ou corrente analógica, sendo digitalizados após passarem por circuito de condicionamento de sinais. Alternativamente, parâmetros podem ser recebidos na forma digital por uma interface digital de comunicação. Os dados são armazenados localmente em um dispositivo removível de memória não volátil (SDCard) e transmitidos para um computador para fins de controle e processamento dos dados. O dispositivo incorpora circuito de alimentação elétrico que entra em ação, garantindo o funcionamento de todos os módulos, quando há interrupção no fornecimento de energia por um período de cerca de 45 horas.

Determina-se o grau de poluição do ar, da água e/ou do solo através da medição de determinados parâmetros físico-químicos obtidos a partir de transdutores elétricos homologados por instituições habilitadas para executar a verificação de conformidade, tais como, Inmetro, IPT, INT, dentre outros.

1.1 MOTIVAÇÃO DO TRABALHO

A origem desta iniciativa remete à experiência do SMV – Sistema de Medição de Vazão, dispositivo destinado à medição do volume produzido de bebidas para fins de fiscalização tributária. O SMV consiste de medidores instalados nas unidades de envase da

indústria de bebidas além de equipamentos destinados à transmissão dos dados resultantes da medição para a Secretaria da Receita Federal – SRF.

O SMV foi instituído através da Medida Provisória nº 2.158-35, de 24 de agosto de 2001, que obrigou sua instalação, num primeiro momento nas indústrias de cerveja, e atribuiu à Secretaria de Receita Federal a função de regular a instalação dos SMVs. A Instrução Normativa SRF nº 265, de 20 de dezembro de 2002, estabeleceu que Cofis – Coordenação-Geral de Fiscalização – publicaria Atos Declaratórios Executivos (ADEs) especificando: “as condições de funcionamento, bem assim as características técnicas e de segurança dos equipamentos; os procedimentos para homologação e credenciamento dos equipamentos e respectivos fabricantes dos mesmos; os limites mínimos de produção ou faturamento, a partir do qual os estabelecimentos ficarão obrigados à instalação dos equipamentos” (BRASIL, 2002, [1]). Desta forma, o Ato Declaratório Executivo Cofis nº 7, de 20 de maio de 2004, trouxe forma ao SMV, definindo as etapas de integração e instalação, verificação da conformidade, pré-qualificação, calibração, avaliação da conformidade e homologação de Sistemas de Medição de Vazão. O primeiro SMV foi homologado através do ADE Cofis nº 9, de 20 de julho de 2004.

Para a proposta do SMV, ficou estabelecido que a medição dos parâmetros físicos temperatura, vazão e condutividade permitiam estimar o volume produzido de bebidas. Desta forma, o SMV registra dados referentes a estes parâmetros, e por meio de interface de rede segura, transmite-os para a SRF. O valor obtido através da leitura do medidor de vazão permite estimar o volume total que atravessa a tubulação, o medidor de condutividade permite identificar o produto sendo envasado, uma vez que cervejas e refrigerantes apresentam tarifações distintas e há linhas de envase que processam ambos produtos. Além disso, procedimentos de limpeza da linha de envase podem ser identificados pela condutividade e temperatura, sendo possível posteriormente considerar estes períodos na totalização do volume produzido. Um aspecto relevante do SMV consiste em suas medições não serem utilizadas diretamente para fins de cálculo dos tributos devidos pelo fabricante de bebidas. Entretanto, permitem ao fisco averiguar se há correspondência entre o volume produzido declarado pelo fabricante e aquele medido pelo SMV. Os dados relativos à produção são enviados diretamente à SRF, sem intervenção humana, sendo uma cópia posteriormente remetida pela SRF ao fabricante.

Os ADEs determinam as ocasiões em que a instalação do equipamento é obrigatória, as instituições de natureza pública aptas a homologar e calibrar os equipamentos que realizam as funções de medição de vazão, medição de condutividade, registro, VPN e firewall.

Somente os equipamentos pré-qualificados podem ser integrados a um SMV. Determinam ainda as especificações mecânicas, elétricas, de segurança, o formato do arquivo de dados para transmissão para a SRF, métodos de ensaio dos equipamentos individuais e métodos de ensaio do SMV instalado.

Como resultado, no ano de 2005 a arrecadação de tributos da indústria de cerveja cresceu 15%, face ao aumento de 6% da produção, mesmo sem terem sido alteradas as tarifas incidentes sobre o produto (MEDINA, 2006, [2]). Desta forma, evidencia-se a redução na evasão fiscal neste setor e a eficácia do SMV. Cabe ressaltar que o custo inicial do sistema, baseado na integração de diversos equipamentos disponíveis no mercado, era da ordem de cento e cinquenta mil reais (R\$150.000,00), restringindo sua aplicação a operações de grande retorno econômico.

Partindo do exemplo do SMV, propõe-se desenvolver um sistema registrador multipropósito destinado a atividades de monitoramento contínuo de parâmetros indicadores de poluição do meio-ambiente e posteriormente sugerir legislação específica que regule as características e situações nas quais a instalação do dispositivo seja obrigatória. Um aspecto importante deste projeto consiste em implementar os circuitos necessários à execução das funções descritas, de modo a tornar viável economicamente sua aplicação na área ambiental. As medições feitas pelos Sistemas Caipora identificariam uma condição de anormalidade, alertando para a necessidade de verificação do fenômeno que está ocorrendo. O Sistema Caipora, instalado próximo à planta de uma indústria química cujo *portfolio* de produtos incluía uma substância que represente risco potencial de poluição de um córrego local, poderia sinalizar o vazamento da substância em concentrações muito inferiores àquelas que seriam visualmente perceptíveis. A partir deste alerta, medidas corretivas seriam tomadas no sentido de estancar o derramamento da substância, prevenindo ou reduzindo as proporções de desastres ambientais. Escolheu-se o nome ‘Sistema Caipora’ devido ao papel deste ser mitológico do folclore brasileiro na proteção da natureza.

1.2 PARÂMETROS FÍSICOS, QUÍMICOS E BIOLÓGICOS

O ar, solo e água podem ser caracterizados por meio de parâmetros físicos, químicos e biológicos. O Conselho Nacional do Meio Ambiente, CONAMA, é o órgão colegiado consultivo e deliberativo do Ministério do Meio Ambiente responsável, dentre outras atribuições, por “*estabelecer normas, critérios, e padrões relativos ao controle e à*

manutenção da qualidade do meio ambiente, com vistas ao uso racional dos recursos ambientais” (BRASIL, 2006, p. 12, [3]).

No que tange a qualidade da água, a resolução CONAMA nº357, de 17 de março de 2005, estabelece os parâmetros adequados à classificação da qualidade da água. A resolução subdivide os corpos de água doce, salobra e salina em treze classes de qualidade. São definidos padrões de qualidade para cada classe, com limites individuais estabelecidos para 36 parâmetros inorgânicos e 54 parâmetros orgânicos. Estabelece ainda indicadores como presença materiais flutuantes, óleos e graxas, características organolépticas (odor e sabor), presença de corantes, resíduos sólidos objetáveis, coliformes termotolerantes, demanda bioquímica de oxigênio, oxigênio dissolvido, turbidez, cor verdadeira e pH (BRASIL, 2005, [5]).

Em 1998, a ANEEL – Agência Nacional de Energia Elétrica, por meio da resolução 396, estabeleceu que, *“em todos os aproveitamentos hidrelétricos, os Concessionários e os Autorizados ficam obrigadas a instalar, manter e operar estações fluviométricas e pluviométricas na região do empreendimento”*. A resolução exige que a estação seja telemétrica para aproveitamentos superiores a 500km², com registro de hora em hora e disponibilização de no mínimo três vezes ao dia. (BRASIL, 1998, [8])

Para o monitoramento da qualidade do ar, a resolução CONAMA nº3, de 28 de julho de 1990 [6], define padrões de qualidade do ar através de limites para os seguintes parâmetros: partículas totais em suspensão, fumaça, partículas inaláveis, dióxido de enxofre, monóxido de carbono, ozônio e dióxido de nitrogênio. A resolução estabelece ainda os métodos de referência do Inmetro para medição destes parâmetros. Há a possibilidade de que sejam utilizados métodos equivalentes. Cabe ressaltar que os parâmetros especificados pela legislação não são estáticos. Havendo indícios de que um parâmetro não listado está interferindo no meio ambiente, este parâmetro deve ser incluído no monitoramento. A resolução define poluente atmosférico como *“qualquer forma de matéria ou energia com intensidade e quantidade, concentração, tempo ou características em desacordo com os níveis estabelecidos, e que tornem ou possam tornar o ar: impróprio, nocivo ou ofensivo à saúde; inconveniente ao bem-estar público; danoso aos materiais, à fauna e flora; prejudicial à segurança, ao uso e gozo da propriedade e as atividades normais da comunidade”* (CONAMA, 1990, [6]).

A rede de monitoramento da qualidade do ar da região metropolitana da cidade do Rio de Janeiro apresenta 32 estações manuais e seis estações automáticas. São monitorados os parâmetros dióxido de enxofre (SO₂), partículas totais em suspensão (PTS), partículas

inaláveis (PI), monóxido de carbono (CO), oxidantes fotoquímicos expressos como ozônio (O₃), hidrocarbonetos totais (HC) e dióxido de nitrogênio (NO₂). Entretanto, algumas estações monitoram apenas os parâmetros PI e/ou PTS. As estações automáticas normalmente monitoram uma quantidade maior de parâmetros, podendo monitorar também a direção e velocidade do vento, temperatura e umidade ambiente. Entretanto, das 32 estações, cerca de 20 monitoram apenas um ou dois parâmetros (FEEMA, 2008?, p. 36, [7]).

1.3 CARACTERÍSTICAS E FUNÇÕES DO SISTEMA CAIPORA

Ao longo da fase de concepção do Sistema Caipora foram enumeradas as diversas características, divididas em dois grupos: obrigatórias, fundamentais ao funcionamento do sistema, e desejáveis, que embora também sejam importantes, podem ser implementadas numa etapa posterior. Desta forma, como características obrigatórias destacam-se: efetuar a leitura de canais analógicos, condicioná-los e digitalizá-los; armazenar as leituras destes canais e outras informações pertinentes em memória não volátil, em formato codificado e/ou criptografado; transmitir os dados por meio de interface específica.

Como características desejáveis, destacam-se: interface com sistema de posicionamento global, GPS, obtendo informação acerca da posição do sistema, data e hora correntes; apresentar em mostrador local (*display*) identificação do sistema, resultados mais recentes da leitura dos canais analógicos, percentual utilizado do dispositivo de armazenamento, data e hora, dentre outras informações. O sistema deve ainda monitorar as tensões de alimentação, desligando-se em condições anormais, e registrar a ocorrência destes eventos.

Uma vez que o Sistema Caipora candidata-se à função de fiscal ambiental, dá-se atenção especial à integridade do sistema, com ênfase à proteção dos dados armazenados no dispositivo de armazenamento. Com este intuito, o processamento feito pelos microcontroladores deve ser distribuído entre os módulos, com garantia de que o módulo responsável pela interface com o 'mundo exterior' não possua funções de escrita no dispositivo de armazenamento. Analogamente, a leitura dos dados armazenados no dispositivo de armazenamento deve ser feita mediante solicitação a um dispositivo intermediário, o qual pode ou não aceitar a solicitação, ao invés de a leitura ser feita diretamente ao dispositivo de armazenamento. Estas medidas têm por finalidade impedir o acesso não autorizado aos dados armazenados, aumentando portanto a confiabilidade das medições feitas com o sistema.

No capítulo 4 apresenta-se uma implementação do Sistema Caipora, cujo diagrama em blocos pode ser visto na Figura 1-1. A disposição do Sistema Caipora em módulos permite que novas funções sejam adicionadas e que partes do sistema sejam desenvolvidas gradualmente.

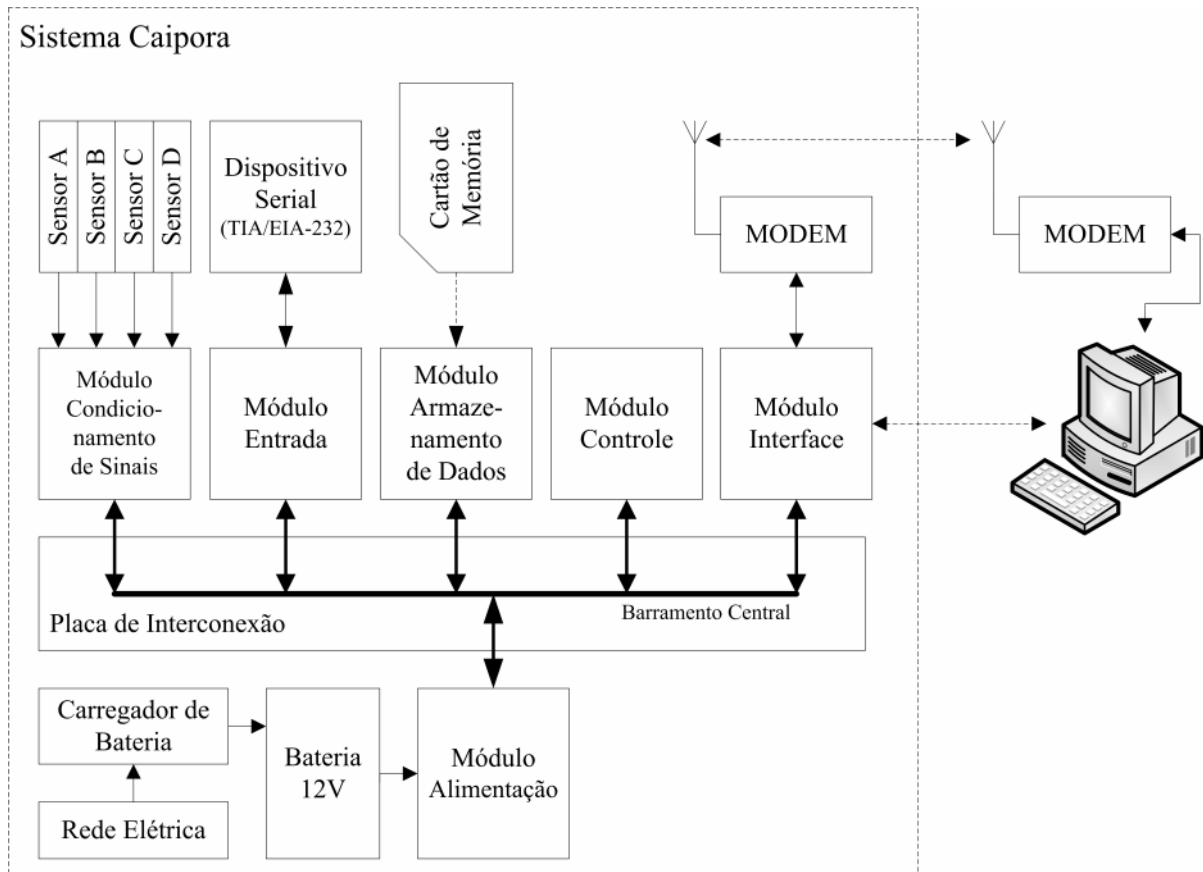


Figura 1-1 – Diagrama em blocos do Sistema Caipora.

1.4 ORGANIZAÇÃO DO TRABALHO

O trabalho está organizado em seis capítulos. Após este capítulo introdutório, o capítulo 2 busca apresentar os fundamentos das tecnologias e dispositivos empregados na implementação do Sistema Caipora, discutindo-se as características de microcontroladores, das interfaces de comunicação utilizadas, do conversor analógico-digital, da referência de tensão e do dispositivo de armazenamento de dados. O capítulo 3 dedica-se ao ambiente computacional de desenvolvimento, apresentando os programas necessários à implementação do trabalho. Faz-se uma descrição sucinta do compilador utilizado para gerar o programa executado no microcontrolador, e dos programas utilizados para converter dados, projetar e

fabricar placas de circuito impresso. O capítulo 4 apresenta uma implementação do Sistema Caipora, descrevendo as características e funções dos diversos módulos que o compõem e dos programas, tanto embarcados quanto executados em estação remota, necessários à sua operação. O capítulo 5 apresenta os resultados da caracterização de partes isoladas do Sistema Caipora e de medições feitas com o Sistema Caipora. O capítulo 6 reúne as conclusões do trabalho e sugere tarefas que dêem continuidade ao desenvolvimento do Sistema Caipora, através de melhorias das funções já implementadas e da adição de novas funcionalidades. O Apêndice A apresenta as Especificações Elétricas do Sistema Caipora.

2

FUNDAMENTOS TÉCNICOS E TEÓRICOS

Para auxiliar na compreensão do funcionamento do Sistema Caipora, neste capítulo são introduzidos conceitos técnicos e teóricos acerca de microcontroladores, oscilador a cristal, dispositivos de armazenamento de dados, conversores analógico-digital e digital-analógico, referência de tensão do tipo bandgap, interfaces seriais de comunicação EIA/TIA-232, I²C e SPI. Em alguns casos, são comparados dispositivos com funcionalidade semelhante, como forma de dar subsídio à escolha de um dispositivo em detrimento de outro. Outrossim, por coesão, dá-se ênfase aos dispositivos utilizados na implementação do sistema.

2.1 MICROCONTROLADOR

Microcontrolador é, fundamentalmente, um computador contido em um único circuito integrado. Normalmente consiste de um núcleo processador, memória RAM e diversos periféricos de entrada e/ou saída de dados (BARRETT e PACK, 2006, [9]). De acordo com o tamanho da palavra processada por uma instrução, há microcontroladores de 8, 16 e 32 bits. Um recurso relativamente recente é a incorporação de memória flash ao microcontrolador. Nas tecnologias anteriores, era comum que o microcontrolador pudesse ser programado apenas uma vez, ou ainda, que através de uma janela no chip fosse aplicada luz ultravioleta antes de reprogramá-lo. A memória flash interna permite que o firmware seja atualizado diversas vezes, normalmente diretamente no próprio circuito projetado. Há ainda a possibilidade de que o próprio código executado no microcontrolador utilize a memória flash para armazenar dados, podendo em alguns casos inclusive alterar porções do código em

execução. Esta flexibilidade tornou corriqueiro o processo de atualização do código executado pelo microcontrolador. Praticamente todos fabricantes disponibilizam um método de programação e/ou depuração que exige acesso a poucos pinos do microcontrolador e permite atualizar o código diretamente no circuito projetado. Para tanto, normalmente conecta-se ao circuito um programador obtido do fabricante. A Tabela 2-1 apresenta comparação das principais características de cinco microcontroladores encontrados com relativa facilidade no mercado brasileiro: PIC18LF1220, fabricado pela Microchip; MSP430F1132, fabricado pela Texas Instruments; AT89S8253 e ATMEGA88, fabricados pela Atmel e MC9S08QG8, fabricado pela Freescale, sendo este último o selecionado para este projeto.

Os microcontroladores da Microchip, conhecidos como PIC, apresentam dispositivos de 8, 16 e 32 bits. Os dispositivos de 8 bits estão agrupados nas famílias 16F e 18F. O dispositivo selecionado para comparação, PIC18LF1220, é o que suporta a maior frequência de operação. Entretanto, apresenta apenas um MSSP, acrônimo de *Master Synchronous Serial Port*, responsável por disponibilizar as interfaces I²C e SPI ao usuário. Durante o desenvolvimento do programa, deve-se optar entre a interface SPI e a interface I²C. A Microchip disponibiliza compilador de linguagem C, MPLAB C18, suportado apenas pela família 18F. A programação é feita por um circuito específico, conhecido como *In-Circuit Debugger* ou ICD.

A Texas Instruments produz a família de microcontroladores de 16 bits MSP430. O dispositivo considerado, MSP430F1132, possui memória semelhante aos demais dispositivos, e a menor frequência máxima de operação. Entretanto, este fator não é significativo, uma vez que no projeto de dispositivos embarcados normalmente prioriza-se o baixo consumo de energia em detrimento do desempenho computacional. Adicionalmente, este fator é em parte compensado pelo barramento de 16 bits do microcontrolador, que permite que uma instrução desloque dois *bytes* de dados na memória. Dentre os dispositivos, foi o único a apresentar apenas encapsulamentos de montagem em superfície (*Surface Mount Device* – SMD). Entretanto, não oferece suporte à interface SPI nem I²C. Um dispositivo que oferece suporte a estas interfaces, o MSP430F1232, está disponível apenas em encapsulamentos com pelo menos 28 pinos. Ainda assim, não é possível utilizar simultaneamente as interfaces I²C e SPI.

O microcontrolador AT89S8253 fabricado pela Atmel utiliza o conjunto de instruções do Intel 8051. Permite ainda que seja adicionada memória SRAM externa ao microcontrolador. Entretanto, não apresenta conversor analógico-digital incorporado e está disponível apenas em encapsulamentos com 40 pinos ou mais. O microcontrolador ATMEGA88 da Atmel apresenta a maior memória SRAM do comparativo. Possui também

quantidade de memória *flash* semelhante aos demais dispositivos. Permite também que as interfaces de comunicação serial, I²C e SPI sejam utilizadas simultaneamente. A relação custo-benefício destes microcontroladores, quando comparados aos demais dispositivos, é desfavorável.

Tabela 2-1 – Comparação das características de microcontroladores.

MICRO-CONTROLADOR	MC9S08QG8	PIC18LF1220	MSP430F1132	AT89S8253	ATMEGA88
Fabricante:	Freescale	Microchip	Texas Instruments	Atmel	Atmel
Núcleo:	HCS08	PIC18	MSP430	8052	AVR
Largura do barramento:	8 bits	8 bits	16 bits	8 bits	8 bits
Frequência Máxima:	20MHz	40MHz	8MHz	24MHz	20MHz
Faixa de tensão:	1,8 - 3,6V	2,0 - 5,5V	1,8 - 3,6V	2,7 - 5,5V	1,8 - 5,5V
Encapsulamento:	DIP16 / TSSOP16	DIP18 / SOIC18 / SSOP20	SOIC20 / TSSOP20	DIP40/42 TQFP44 PLCC44	DIP28 TQFP32
Memória Flash:	8KB	4KB	8KB	12KB	8KB
Memória RAM:	512B	256B	256B	256B	1024B
EEPROM:	–	256B	–	2KB	512B
ADC:	8 canais 10 bits	7 canais 10 bits	5 canais 10 bits	–	6 canais 10 bits
Oscilador Interno:	Sim	Sim	Sim	Sim	Sim
Serial RS232	Sim	Sim	Sim	Sim	Sim
SPI	Sim	Não	Não	Sim	Sim
I ² C	Sim	Não	Não	Não	Sim
Programação <i>In-Circuit</i>	Sim, BDM	Sim, ICD	Sim, MSP FET	Sim, ISP	Sim, DebugWIRE
Preço ^a R\$	7,00	16,35	21,61	24,25	14,42
Preço US\$	0,85 / 1k	2,25 / 1k	2,80 / 100u	4,60 / 1k	2,46 / 1k

Fonte: Freescale Semiconductor, Microchip, Texas Instruments e Atmel.

^a Fonte: Consulta online ao distribuidor Farnell em janeiro de 2009.

A família de microcontroladores HCS08 possui arquitetura do tipo von Neuman, isto é, o código de programa e as variáveis e dados do usuário compartilham o mesmo espaço de endereçamento. Possuem o barramento de dados interno da CPU com duas seções distintas: um de leitura, outro de escrita (PEREIRA, 2005, [10]). Consistem de um processador, memória *Flash*, memória estática e diversos dispositivos periféricos embutidos em um mesmo encapsulamento ou chip. O processador, de 8 bits, suporta frequência de operação de até 20MHz, podendo ser gerada internamente através do módulo *Internal Clock Source* (ICS), utilizando-se um cristal ou ressonador externo ao microcontrolador, ou ainda utilizando-se uma referência externa de onda quadrada. Este *clock* principal, serve de referência para a obtenção das frequências necessárias à operação dos diversos periféricos embutidos no microcontrolador. A memória Flash apresenta retenção de dados por, no mínimo, 15 anos, sendo que em aplicações típicas à temperatura ambiente esta estimativa sobe para 100 anos. Como periféricos, destacam-se os módulos conversor analógico digital, comparador analógico, comunicação serial EIA/TIA 232, comunicação SPI, comunicação I²C, gerador de temporização e pulsos. Devido à limitação física do número de pinos do microcontrolador, algumas funções podem estar atreladas a um pino específico, o que impede o funcionamento simultâneo de tais funções. Por exemplo, ao utilizar um pino como entrada analógica para o ADC, não se pode utilizar a função alternativa de entrada/saída digital neste pino. Em outros casos, é possível que a função seja redirecionada para outro pino. Um exemplo é a função I²C, que pode utilizar os pinos destinados ao oscilador a cristal (em sistemas que não o utilizem) ou um determinado par de pinos de entrada/saída. Ainda assim, o MC9S08QG8 suporta operação simultânea das interfaces de comunicação serial assíncrona EIA/TIA 232, serial síncrona SPI e I²C (FREESCALE, 2008, [11]).

Diversos microcontroladores suportam, respeitadas suas especificidades, os procedimentos descritos a seguir. Entretanto, as seções a seguir descrevem os microcontroladores Freescale da família HCS08, uma vez que este foi utilizado na implementação do Sistema Caipora.

2.1.1 Configuração do microcontrolador

As diversas opções de configuração do sistema são definidas através da gravação de registradores de função especial. Na família HCS08, tais registradores são mapeados na memória em duas seções: Registradores de página direta (0x0000h a 0x005Fh) e registradores de página alta (0x1800h a 0x184Fh). Os registradores possuem largura de 8 bits. Cada bit do registrador pode estar associado a uma função, sendo portanto necessário o mascaramento dos

bits desejados antes de escrever no registrador. Ao ler a posição de memória correspondente, obtém-se a informação acerca do estado atual da função controlada pelo registrador. Para evitar que o programador tenha que memorizar o endereço de cada registrador que desejar acessar, associa-se a cada registrador (e por vezes a um bit específico do registrador) um mnemônico indicativo de sua função. Por exemplo, o resultado da conversão do ADC é disponibilizado nos endereços 0x0012h e 0x0013h. No compilador estes endereços são referenciados como *ADCRH (ADC Result High)* e *ADCRL (ADC Result Low)*. Outra vantagem do uso de mnemônicos é a portabilidade do código entre microcontroladores de um mesmo fabricante. Por exemplo, o registrador *ADCRH* pode estar localizado em um endereço de memória diferente em outro microcontrolador, e ainda assim o código ser compilado sem maiores alterações.

Os pinos do microcontrolador configurados como entrada podem ter habilitado internamente um resistor de *pullup*. Além de dispensar a utilização de um resistor externo, permite que o *pullup* seja habilitado e desabilitado conforme necessário. Os pinos configurados como saída podem ser configurados para suprirem correntes superiores aos cerca de 2mA, através da função *output drive strength*.

Para reduzir a emissão de radiações eletromagnéticas, pode-se reduzir a taxa de variação da tensão de saída da porta, através do controle *output slew rate control*. Desta forma, o tempo de transição entre os níveis alto e baixo passa de 3ns para cerca de 30 ns (FREESCALE, 2008, p. 278, [11]).

O microcontrolador permite a escolha da fonte de frequência de oscilação dentre diversas opções: oscilador interno do microcontrolador, oscilador a cristal, ressonador cerâmico e frequência externa.

2.1.1.1 Operação do microcontrolador a partir do oscilador interno

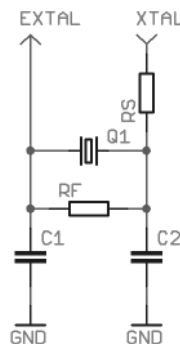
Nas aplicações que não requerem grande precisão ou estabilidade do *clock* do sistema é possível utilizar o oscilador interno do microcontrolador (ICS) como origem do sinal de *clock*. Devido às características do processo de fabricação do microcontrolador, há variação significativa da frequência gerada pelo oscilador interno de cada chip. Para tanto, pode-se ajustar a frequência de operação através dos registradores *FTRIM* e *TRIM*. Normalmente o chip vem de fábrica com um valor aproximado de ajuste armazenado na memória *flash*. Porém é possível ajustar com maior precisão durante o procedimento de programação do microcontrolador. O resultado deste ajuste é armazenado em duas posições fixas da memória *flash* do microcontrolador, para então, durante a inicialização do microcontrolador, serem

copiados para os registradores FTRIM e TRIM. A partir de então, o microcontrolador opera em até 0,2% da frequência desejada.

Após o ajuste, a frequência pode desviar do valor esperado em até 2% ao longo das faixas de temperatura de operação e de tensão de alimentação, aceitável para a operação de uma porta serial assíncrona, porém inapropriada para aplicações de contagem de tempo, como implementação de relógio de tempo real. Para aplicações que requeiram uma base de tempo mais precisa, utiliza-se um cristal externo.

2.1.1.2 Operação do microcontrolador a partir de cristal externo

A frequência obtida a partir de um cristal é mais precisa e estável que aquela gerada por um ressonador cerâmico ou pelo circuito oscilador interno do microcontrolador. O cristal, por ser um dispositivo passivo, não gera oscilações em seus terminais. Para tanto, é necessário introduzi-lo na malha de realimentação de um circuito amplificador, incorporado ao microcontrolador. Desta forma, basta conectar o cristal aos pinos EXTAL e XTAL do microcontrolador, dimensionar os capacitores C_1 e C_2 e, se necessário, os resistores R_S e R_F (conforme visto na Figura 2-1) de acordo com os dados da folha de especificações do cristal e do microcontrolador.



Fonte: (FREESCALE, 2008, p. 277, [11])

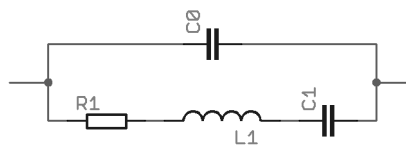
Figura 2-1 – Circuito para conexão de cristal ao microcontrolador.

Os resistores R_S e R_F apresentam apenas ajustes do ponto de operação do cristal, com pouca ou nenhuma influência sobre a frequência de operação. Operar o cristal em um nível de potência excessivo resulta na degradação de suas características, podendo causar instabilidade de frequência ou danos físicos ao cristal. A precisão de um cristal é usualmente expressa em ppm (partes por milhão). Um valor de 10ppm equivale a uma tolerância de até ± 1 segundo a cada 100.000 segundos (cerca de 28 horas). Cabe ressaltar que, uma vez que dois cristais

apresentem frequências distintas, porém com tolerâncias similares, ambos estarão sujeitos a desvios absolutos semelhantes na contagem de tempo.

Um cristal de 32768Hz permite que seja obtida uma frequência de 1Hz (período de 1s) após divisão por 2^{15} , tarefa que pode ser efetuada pelo microcontrolador, seja pela divisão direta do sinal de *clock*, seja pelo uso de interrupções. Este valor de cristal é amplamente utilizado em aplicações de calendário e relógio, entretanto também podem ser utilizados cristais com outras frequências nestas aplicações.

O cristal é um dispositivo piezelétrico, ou seja, ao sofrer uma deformação mecânica apresenta em seus terminais uma tensão elétrica, e vice-versa. O circuito elétrico equivalente do cristal oscilador consiste de um circuito RLC em paralelo com uma capacitância C_0 , esta última normalmente dominada pela capacitância do encapsulamento do dispositivo, conforme mostrado na Figura 2-2.

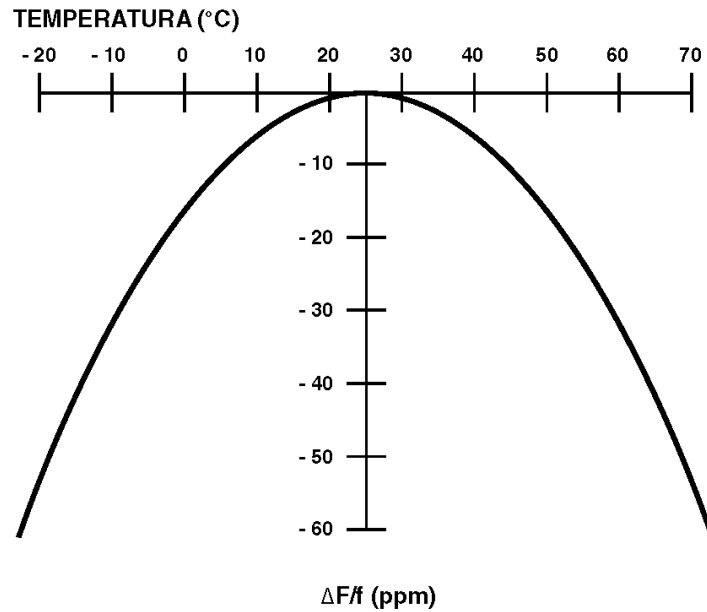


Fonte: (SAFARI, 1999, p. 150, [12])

Figura 2-2 – Circuito equivalente de cristal oscilador.

Os microcontroladores da família HCS08 permitem que a frequência do cristal, na faixa de 31,25 a 38,4kHz, seja multiplicada por 512 por um circuito de FLL (*Frequency Locked Loop*). Assim, a partir de um cristal de 32,768kHz obtém-se frequência de 16,7MHz, sendo a frequência do barramento 8,38MHz. Esta frequência é dividida por dois, quatro ou oito, caso necessário. Alternativamente, pode-se desabilitar o circuito de multiplicação FLL e utilizar diretamente o cristal de valor desejado.

Uma característica importante dos osciladores a cristal é a sensibilidade da frequência de oscilação com a variação da temperatura, expressa em ppm/°C. A Figura 2-3 mostra uma curva típica da variação da frequência de oscilação com a temperatura para um cristal de 32,768kHz.



Fonte: (VISHAY, 2007, p. 1, [13])

Figura 2-3 – Curva típica de variação da frequência de oscilação de um cristal com a temperatura.

A curva do desvio da frequência de oscilação com a temperatura, com forma de uma parábola com concavidade para baixo, pode ser aproximada pela equação (2-1), onde T é a temperatura de operação do cristal, T_i é a temperatura na qual o cristal foi calibrado, normalmente 25°C e B é o coeficiente parabólico. Observa-se que o cristal em questão tenderá a oscilar sob frequência menor quando operado em temperaturas diferentes da T_i .

$$\Delta freq = B \cdot (T - T_i)^2 \quad (2-1)$$

Outro fator que colabora para desvios significativos da frequência de oscilação consiste em utilizar capacitâncias distintas daquelas especificadas pelo fabricante. Usualmente o valor da capacitância de carga C_L é fornecido pela folha de especificação do cristal, sendo necessário igualar o valor de C_1 em série com C_2 a este valor (ver Figura 2-1). Deve-se ainda levar em conta a capacitância das trilhas que ligam o cristal aos pinos do microcontrolador, que é da ordem de .

2.1.2 Reinicialização do microcontrolador

O microcontrolador pode ser reinicializado a partir de diversos meios, e portanto é importante poder identificar a causa da última reinicialização. O registrador *System Reset Status Register* (SRS) informa se a reinicialização foi motivada por uma tentativa de acesso a posições de memória inexistentes no chip (*Illegal Address Detect*) ou execução de instruções inválidas (*Illegal Opcode Detect*). Desta forma, é possível que durante a inicialização o

microcontrolador execute ações distintas de acordo com a situação que gerou a reinicialização.

O microcontrolador é reinicializado pelo pino de *reset* externo, pelo circuito que detecta que a tensão de alimentação está abaixo de um limiar e pelo timer de ‘computador operando corretamente’, caso estas funções sejam habilitadas. Este último consiste de um timer decrescente que deve ser periodicamente recarregado pelo programa sendo executado, e é normalmente conhecido como ‘cão de guarda’ ou *watchdog*. Caso o programa não recarregue o timer a tempo, o circuito de ‘cão de guarda’ identifica uma condição de erro e reinicializa o microcontrolador.

2.1.3 Módulo de depuração do microcontrolador

O microcontrolador incorpora um módulo denominado *Background Debug Module* – BDM, contendo circuitos destinados à sua programação e depuração. O acesso a este módulo se dá por meio de dois pinos. Trata-se de uma interface de auxílio ao desenvolvimento do programa, por meio da qual se lê e escreve na memória, programa-se o microcontrolador e controla-se a execução do programa. Por motivos de segurança, a interface BDM deve ser desabilitada na versão final, como forma de evitar o acesso direto aos registradores internos e variáveis armazenadas no microcontrolador.

O módulo de depuração é transparente, isto é, permite a leitura e escrita da memória e de registradores internos do microcontrolador sem interferir na execução do programa. A integração com o programa depurador permite que na tela do computador sejam vistas as alterações dos valores de variáveis e que se controle a execução do programa, parando nos pontos de interesse para identificar eventuais fontes de erros.

O programador conecta-se à interface BDM por meio de conector de seis pinos que contém os sinais alimentação (V_{DD}), terra (GND), dados da interface BDM (BKGD) e reinicialização do microcontrolador (RESET). Caso o circuito apresente alimentação própria, o sinal de V_{DD} não precisa ser aplicado ao conector BDM. O sinal de RESET é necessário para que o programador coloque o microcontrolador no modo BDM.

Desta forma, apenas dois pinos do microcontrolador são utilizados para que sejam disponibilizadas as funções de depuração em tempo real: BKGD e RESET. Ainda assim, pode-se utilizar posteriormente o pino BKGD como saída, e pino RESET como entrada, caso o desenvolvimento do programa não necessite mais das funções BDM.

2.2 DISPOSITIVOS DE ARMAZENAMENTO

Dispositivos de armazenamento são responsáveis pela conservação da informação de interesse por um período relativamente longo de tempo. Podem ser divididos em duas classes: dispositivos com partes móveis e dispositivos sem partes móveis. Na primeira classe estão incluídos os discos flexíveis, discos rígidos, discos ópticos de escrita única ou regraváveis, dentre outros. Na segunda classe encontram-se os dispositivos de memória de estado sólido, em especial os dispositivos de armazenamento em memória do tipo *flash* e EEPROM (*Electrically Erasable Read Only Memory*). Estas últimas são ditas memórias não voláteis porque retêm os dados mesmo após a interrupção no fornecimento de energia. A memória EEPROM permite que um único *byte* seja apagado e/ou programado a cada vez, ao passo que a memória *flash* requer que um bloco inteiro de memória seja apagado numa única operação [14]. Desta forma, para se escrever apenas um *byte* na memória *flash* faz-se necessário ler o conteúdo de um bloco, alterar o *byte* desejado e reescrever o bloco inteiro na memória. A memória *flash* permite densidades maiores de memória, uma vez que sua estrutura utiliza menos transistores para armazenar um bit de informação quando comparada com a memória EEPROM.

2.2.1 Cartão de memória SD Card

O cartão de memória *SD Card*, mostrado na Figura 2-4, é um dispositivo de armazenamento de dados sem partes móveis que utiliza memória de estado sólido não volátil. Este tipo de mídia é amplamente utilizado em câmeras fotográficas digitais, tocadores de música digital, celulares, dentre outros. O acrônimo *SD Card* refere-se a *Secure Digital Card*, ficando a cargo da *SD Card Association* a elaboração e manutenção das especificações técnicas. Para se ter acesso às especificações completas do SD Card, é necessária a filiação à *SD Card Association*, além da assinatura de um termo de sigilo. Entretanto, a *SD Card Association* fornece uma versão reduzida da especificação *SD Simplified Specification* (SD GROUP, 2006, [15]), suficiente para o desenvolvimento de interface de leitura e escrita do cartão. Nesta versão reduzida não são descritos os recursos de proteção digital de conteúdo (*Content Protection for Recordable Media (CPRM)*), destinada principalmente a dados multimídia, nem as especificações dos cartões do tipo SDHC (*Secure Digital High Capacity*), com capacidade acima de 2GB. Ainda assim, o uso desta versão reduzida não exime o desenvolvedor de qualquer responsabilidade quanto ao seu licenciamento junto à *SD Card Association*.

O cartão está disponível em três tamanhos diferentes: SD tamanho padrão (24mm x 32mm), miniSD (20mm x 21,5mm) e microSD (11mm x 15mm). Há adaptadores que permitem a inserção dos cartões miniSD e microSD no conector destinado ao cartão tamanho padrão.



Fonte: (SD CARD ASSOCIATION)

Figura 2-4 – Cartão de memória SD Card, tamanho padrão.

A interface elétrica do cartão consiste de nove pinos. Três pinos destinam-se à alimentação elétrica do cartão. Uma vez que o cartão pode ser acessado por dois modos distintos (barramento SD e barramento SPI), a função dos seis pinos restantes é definida pelo modo em que se está operando.

O cartão possui uma chave seletora na lateral, destinada à proteção contra a escrita. É função do conector mecânico sinalizar a presença do cartão e a posição da chave de proteção contra a escrita por meio de micro interruptores. Entretanto, a posição desta última não possui conexão elétrica interna com o cartão. A responsabilidade por impedir a gravação em cartões protegidos compete ao sistema ao qual ele é conectado. Caso esta proteção não seja implementada no sistema, os dados do cartão poderão ser apagados mesmo que a posição da chave esteja no modo somente leitura.

2.2.1.1 Acesso ao Cartão no Modo SD

O barramento SD consiste de cinco linhas de dados bidirecionais (DAT0-3 e CMD) e uma linha de temporização com a função de *clock*. Os comandos e suas respectivas respostas são transmitidos pela linha CMD, ao passo que os dados utilizam as linhas DAT0-3. Comparado ao barramento SPI, permite taxas de transferência de dados cerca de quatro vezes maiores sob a mesma frequência do barramento. Uma vez que somente neste modo há suporte a frequências de até 50MHz, o acesso ao cartão no modo SD pode ser até oito vezes mais rápido que no modo SPI, resultando em taxa máxima teórica de transferência de dados de 25MB/s.

2.2.1.2 Acesso ao Cartão no Modo SPI

O protocolo SPI, descrito na seção 2.3.2, foi utilizado para implementar o controle do cartão de memória neste projeto. Neste modo, são utilizadas uma linha de dados de saída do cartão, uma linha de entrada de dados, uma linha de seleção do cartão no barramento e uma linha de *clock* do barramento. A frequência máxima de clock no modo SPI é de 25MHz, o que resulta numa taxa máxima de transferência de dados teórica de cerca de 3MB/s. No modo SD há 4 linhas para a transmissão, o que eleva a taxa máxima de transferência (teórica) para cerca de 12MB/s.

2.2.1.3 Comandos de Acesso ao Cartão

Os comandos tem tamanho fixo de 48 bits, sendo compostos por um bit de início, um bit indicativo de que a transmissão tem sentido para o cartão, seis bits com o código do comando, 32 bits com o conteúdo do comando, sete bits com o código de verificação CRC e um bit indicando o fim do comando, conforme mostrado na Tabela 2-2. Os números entre colchetes indicam a posição dos bits dentro da estrutura do comando.

Tabela 2-2 – Estrutura dos comandos SD Card.

<i>Início</i> [47]	<i>Indicador</i> [46]	<i>Comando</i> [45:40]	<i>Argumento</i> [39:8]	<i>CRC7</i> [7:1]	<i>Fim</i> [0]
'0'	'1'				'1'

No modo SPI o cartão SD oferece suporte para 25 comandos, sendo que apenas um subconjunto foi implementado. O comando CMD0 reinicializa o cartão para estado 'livre'. É o primeiro comando enviado ao cartão durante a inicialização no modo SPI. O CMD1 retorna o status de operação do cartão, além de acionar a inicialização do cartão. Os comandos CMD17 e CMD24 efetuam respectivamente a leitura e a escrita de um bloco de dados de 512 *bytes*. Ambos tomam como argumento os quatro *bytes* (32 bits) correspondentes ao endereço do primeiro *byte* do bloco a ser lido/escrito. Há ainda comandos para ler e escrever múltiplos blocos, apagar um intervalo de memória, proteger o cartão de memória ou parte dele contra a escrita, confirmar se o cartão suporta a tensão de alimentação disponível e outros comandos de uso específico.

2.2.1.4 Estrutura de Armazenamento de Dados no Cartão

Os dados são estruturados no cartão em setores de 512 *bytes*. Embora seja possível ler blocos tão pequenos quanto um *byte*, o endereço deve ser alinhado de forma que a leitura/escrita não cruze as bordas de 512 *bytes*, isto é, os dados lidos/gravados devem pertencer ao mesmo setor. O tamanho do bloco de leitura/escrita pode ser ajustado através do comando CMD16, sendo o valor padrão 512 *bytes*.

Cartões com capacidade igual ou superior a 4GB utilizam uma versão mais recente do *SD Card Physical Layer System Specification*, denominada SDHC (*Secure Digital High Capacity*, ou Cartão SD de alta capacidade). Nesta nova revisão, dentre outras atualizações, o endereço aponta para o setor, e não mais para o *byte* dentro do setor. Uma vez que para o cartão normal o endereçamento é de 32 bits, e este seleciona apenas um *byte* no cartão, a região endereçável consiste de $2^{32} = 4\text{GB}$.

2.2.1.5 Procedimento de Escrita de um Bloco no Cartão

A sequência de escrita de um bloco de dados consiste em enviar o comando CMD24, enviar *bytes* 0xFF até que a resposta do cartão ao comando seja recebida. Caso a resposta não indique problemas, procede-se enviando o caractere indicador de início de bloco 0xFE, seguido dos 512 *bytes* do bloco de dados e de dois *bytes* do CRC16. Enviam-se caracteres 0xFF até que se obtenha uma resposta do cartão. A resposta permite identificar se o bloco foi escrito ou rejeitado devido a erro de CRC16 ou devido a um erro interno de escrita do cartão.

2.2.1.6 Registradores internos do Módulo Armazenamento

O cartão de memória SDCard apresenta alguns registradores internos, destacando-se os registradores OCR, CID e CSD, descritos sucintamente a seguir.

O registrador OCR (*Operating Conditions Register*), de 32 bits, informa a faixa de tensão de operação do cartão de memória, se o cartão é de alta capacidade e se o procedimento de inicialização interno do cartão foi finalizado. Utiliza-se o comando CMD58 para ler o OCR.

O registrador CID (*Card Identification*), de 128 bits, informa a identificação do fabricante do cartão, informações diversas sobre o produto e o número de série de 32 bits único do cartão. Utiliza-se o comando CMD10 para ler o CID.

O registrador CSD (*Card-Specific Data*), de 128 bits, fornece informações acerca das características do cartão, como tempo de acesso, consumo máximo de corrente durante a leitura e escrita e os parâmetros que permitem calcular o tamanho do cartão. Utiliza-se o comando CMD9 para ler o CSD.

2.2.1.7 Cálculo do Tamanho do Cartão

A capacidade do cartão é calculada através da equação (2-2), substituindo-se os valores dos parâmetros C_SIZE, C_SIZE_MULT e BLOCKLEN lidos do registrador CSD.

$$\text{Capacidade} = (C_SIZE + 1) \cdot 2^{(C_SIZE_MULT+2)} \cdot \text{BLOCKLEN} \quad (2-2)$$

Uma vez que na versão 1.1 do *SD Card Physical Layer System Specification* C_SIZE varia de 0 a 4095, C_SIZE_MULT varia de 0 a 7 e BLOCKLEN varia de 512 a 2048 *bytes*, o tamanho máximo que pode ser representado é de 2^{32} *bytes* = 4GB.

2.2.1.8 Detecção de Erros no Cartão

A integridade dos dados do cartão de memória é verificada pelo código de detecção de erros CRC, acrônimo de *cyclic redundancy check*. Utiliza-se CRC de sete bits para os comandos de acesso ao cartão, e CRC de 16 bits para a transferência dos blocos de dados. No modo SPI, o uso do CRC é habilitado/desabilitado por meio do comando CMD59. Para os dados transferidos em blocos de 512 *bytes* utiliza-se o CRC de 16 bits.

Para calcular o CRC16, cada bit dos dados que compõem o pacote é considerado como o coeficiente de uma potência do polinômio. Este polinômio é então multiplicado por X^{16} e então dividido pelo polinômio do gerador ($X^{16} + X^{12} + X^5 + 1$) usando aritmética de módulo dois. Nesta aritmética, as operações são efetuadas bit a bit, sem que se leve em consideração os ‘vai-um’ nem os ‘empresta-um’. Desta forma, tanto a soma quanto a subtração módulo 2 são a simples operação lógica ‘OU exclusivo’, ao passo que a multiplicação torna-se a operação lógica ‘E’. O resto desta divisão é o CRC desejado. Uma vez que o bloco possui 512 *bytes*, isto é, 8192 bits, o polinômio será da ordem X^{511} . O CRC detecta todos os erros de um e dois bits, todos os erros com número ímpar de bits, todos os erros em rajada de comprimento menor ou igual a 16 bits, 99.997% de erros em rajada de 17 bits, e 99.998% dos erros em rajadas de 18 bits ou mais (TANEMBAUM, 1988, p. 208-212, [16]).

Para agilizar o cálculo do CRC16, utiliza-se uma tabela de 256 números inteiros positivos de 16 bits (*unsigned integer*), armazenada na memória *flash* do microcontrolador, ocupando 512 *bytes* da memória não volátil. Obtém-se o CRC16 utilizando-se a equação (2-3), onde ‘CRC_{novo}’ será o valor do CRC atualizado; ‘CRC_{atual}’ é o valor do CRC atual; ‘<<

8' é o operador que desloca o valor 8 bits para a esquerda; '>> 8' é o operador que desloca o valor 8 bits para a direita; '⊕' é o operador lógico 'OU Exclusivo'; 'Dado' é o *byte* de informação do bloco e $CRC16_{tabela}[posição]$ toma o valor da posição selecionada, entre colchetes, da tabela de aceleração. Calcula-se o CRC para cada *byte* adicionado ao bloco de dados.

Durante a gravação/leitura de um bloco de 512 *bytes*, inicia-se o valor do CRC16 com zero. A cada *byte* escrito/lido, atualiza-se o valor do CRC16. Após 512 atualizações, obtém-se o CRC16 do bloco de dados. Para o comando de escrita de um bloco, este CRC16 é enviado ao cartão, que indica se o valor confere com o CRC16 calculado por ele. Para o comando de leitura, o CRC16 é enviado logo após o bloco de dados, cabendo ao receptor conferir se os códigos conferem.

$$CRC_{novo} = (CRC_{atual} \ll 8) \oplus CRC16_{tabela} [(CRC_{atual} \gg 8) \oplus Dado] \quad (2-3)$$

O CRC de sete bits é calculado de maneira análoga, porém utiliza-se o polinômio $X^7 + X^3 + 1$. Entretanto, uma vez que o procedimento de cálculo do CRC7 é mais curto e esta rotina é utilizada apenas para a verificação de erros no prefácio de cada comando, não se faz necessário utilizar uma tabela de aceleração para obter o CRC7.

2.3 PROTOCOLOS DE COMUNICAÇÃO

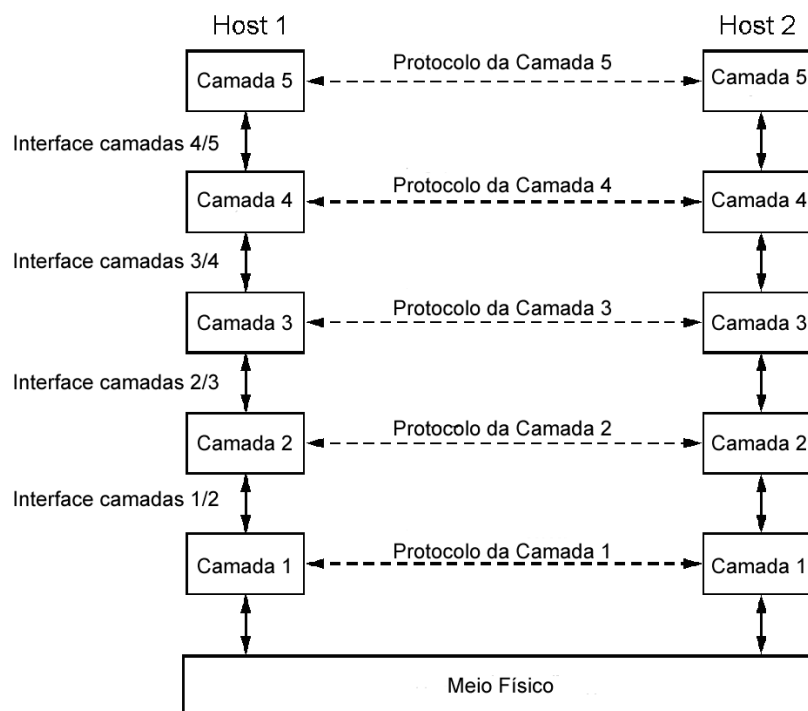
A ISO, Organização Internacional para a Padronização, publicou em 1984 um modelo de referência para descrição de redes denominado OSI (*Open Systems Interconnection*). Em 1994 o documento foi revisado, resultando no ISO-7498:1994 (ISO/IEC, 1996, [17]). No Brasil a ISO é representada pela Associação Brasileira de Normas Técnicas – ABNT. O modelo decompõe uma rede de comunicação arbitrária em uma pilha com sete camadas: 7-aplicação, 6-apresentação, 5-sessão, 4-transporte, 3-rede, 2-enlace e 1-física^b. Em cada camada certos tipos de função de comunicação de rede são descritas.

A Figura 2-5 apresenta o modelo em camadas OSI. O usuário do sistema de comunicação, normalmente um outro sistema que precisa se comunicar na rede, interage com a camada 7-aplicação. O meio físico está conectado à camada 1-física. Algumas implementações não utilizam todas as camadas do modelo (CROVELLA, 1999, [18]).

Neste contexto, serviço é um conjunto de operações que uma camada fornece para a camada imediatamente superior. O serviço define quais operações a camada está preparada

^b Os nomes das camadas, em língua inglesa, são: *application layer*, *presentation layer*, *session layer*, *transport layer*, *network layer*, *data link layer*, *physical layer*.

para efetuar em benefício de seus usuários, mas não diz nada a respeito de como estas operações são implementadas. Um serviço refere-se a uma interface entre duas camadas, com a camada inferior sendo o provedor do serviço e a camada superior sendo o usuário do serviço. Protocolo, por outro lado, é um conjunto de regras que governam o formato e significado dos quadros, pacotes ou mensagens que são trocados pelas entidades correspondentes de uma camada. As entidades usam protocolos para implementar suas definições de serviço. O protocolo pode ser alterado, desde que o serviço visível aos usuários não mude. Para que haja comunicação entre dois sistemas, os protocolos selecionados para cada camada devem ser compatíveis. Normas adicionais padronizam os protocolos utilizados. (TANENBAUM, 1988, [16])



Fonte: (TANENBAUM, 1988, p. 10, [16])

Figura 2-5 – Estrutura em camadas do modelo OSI.

A camada 1-física é responsável pela transmissão dos bits através do canal de comunicação e por lidar com interfaces mecânicas e elétricas. Efetua a interface com o meio físico de transmissão. A camada 2-enlace tem as funções de fornecer uma interface de serviços para a camada 3-rede, determinar como os bits da camada 1-física são agrupados em quadros, administrar erros de transmissão, controlar o acesso ao meio físico, controlar o fluxo

de quadros de forma que um receptor lento não seja inundado por um transmissor rápido e tarefas de supervisão do enlace (TANEMBAUM, 1988, [16]).

A camada 3-rede compreende funções relacionadas ao roteamento das mensagens por meio de endereçamento. A camada 4-transporte tem a função de aceitar dados da camada 5-sessão, dividí-los em partes menores se necessário e transferi-los para a camada 3-rede, garantindo que os fragmentos sejam recebidos pela camada correspondente do receptor, onde são concatenados. A camada 5-sessão permite que usuários em máquinas diferentes estabeleçam sessões entre si. Para tanto, tem a função de estabelecer, manter e encerrar um canal de comunicação de dados. Esta camada é raramente implementada em sistemas de comunicação (CROVELLA, 1999, [18]).

A camada 6-apresentação tem função de converter dados entre os formatos transmitidos pela rede para os formatos acessíveis ao usuário, definidos na camada de aplicação. A camada 7-aplicação fornece serviços diretamente à aplicação do usuário. O uso e formatação destes serviços é definido no protocolo da camada 7-aplicação. O usuário interage com a rede através de funções e serviços fornecidos pela camada 7-aplicação, transmitindo e recebendo dados através destes serviços. Em muitos sistemas de comunicação de sensores, as funções executadas pelas camadas 3-rede, 4-transporte, 5-sessão e 6-apresentação são condensadas e incorporadas vestigialmente às funções da camada 7-aplicação (CROVELLA, 1999, [18]).

Faz-se necessário definir alguns termos fundamentais: Uma Estação representa um único elemento de comunicação da rede. Cada Estação terá uma implementação de rede OSI como meio de uso da rede. O Controle de Acesso ao Meio é o método através do qual as estações determinam quando há permissão para transmitir, ou seja, usar o meio físico. Normalmente é implementado na camada enlace. Um exemplo de método de controle de acesso é o mestre-escravo. Neste método, uma Estação mestre da rede é encarregada de iniciar toda a comunicação. As Estações escravo somente respondem à estação mestre, e somente transmitem quando a estação mestre inicia a comunicação. O Endereçamento é o conceito que atribui identificadores únicos a cada Estação da rede, permitindo identificar a Estação na rede. A Arbitração define o procedimento a ser seguido quando mais de uma Estação desejam acessar a rede simultaneamente (CROVELLA, 1999, [18]).

2.3.1 Protocolo de Camada Física EIA/TIA-232.

Refere-se à interface serial assíncrona, definida originalmente pela norma EIA/TIA-232 da Electronics Industry Association em 1962. Atualmente na revisão F (1997), a norma define as características elétricas, funcionais e mecânicas para a conexão e comunicação entre um dispositivo DTE (*Data Terminal Equipment*) e um DCE (*Data Communication Equipment*). No modelo OSI, refere-se ao protocolo entre camadas ‘física’.

O nível lógico é definido por tensão. Para o transmissor, o bit ‘1’ é codificado por -5 a -15V, ao passo que o nível baixo por +5 a +15V. Observa-se que há uma inversão em relação ao senso comum, sendo o bit 1 representado por tensão negativa. Do lado do receptor, uma tensão entre -3 e -15V determina um bit ‘1’ e entre +3 e +15 um bit ‘0’. A diferença entre o menor valor transmitido e o menor valor detectado como um nível válido ($|\pm 5| - |\pm 3| = 2$) corresponde à margem de ruído, definida como “a máxima amplitude em tensão de um sinal externo que pode ser algebricamente somado ao nível de entrada, sem ruído, sem causar o desvio da tensão de saída do nível lógico aceitável”^c (JEDEC, 1997, [19]) (tradução nossa). Ou seja, um sinal (ruído) adicionado ou subtraído com amplitude menor que a margem de ruído não alteraria, em tese, o sinal transmitido.

Embora a norma original estabelecesse o comprimento máximo do cabo permitido, a norma atual determina a capacitância máxima do cabo em 2500pF (MAXIM-DALLAS, 2001, [20]). O parâmetro capacitância/metro normalmente é informado pelos fabricantes de cabos, bastando dividir a capacitância máxima por este parâmetro para obter a estimativa do comprimento máximo do cabo.

O nível de terra (0V), assim como qualquer valor no intervalo -3 a +3V, é inválido. A função de desligamento automático *autoshtutdown*, presente em diversos dispositivos de interface RS232 do fabricante Maxim, monitora a tensão nas linhas de entrada, e ao verificar níveis inválidos nas entradas da interface por um período superior a 30μs, coloca o dispositivo em um modo de baixo consumo de energia. Este modo é interrompido tão logo se detecte atividade no barramento serial, sendo necessários cerca de 100μs para o retorno do dispositivo ao modo ativo (MAXIM-DALLAS, 2001, [20]).

A norma original especifica 24 sinais, acomodando-os em um conector de 25 pinos. Entretanto, utiliza-se normalmente um subconjunto com apenas oito destes sinais alojados em

^c A definição original, em língua inglesa, é: “The maximum voltage amplitude of extraneous signal that can be algebraically added to the noise-free worst-case input level without causing the output voltage to deviate from the allowable logic voltage level”.

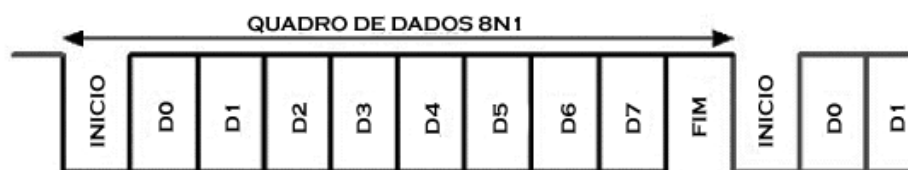
conector de nove pinos. Por exemplo, a implementação da porta serial EIA/TIA-232 presente nos computadores pessoais é deste tipo. Os sinais presentes no conector de nove pinos são apresentados na Tabela 2-3. Além destes oito sinais, o conector incorpora também a referência de terra ou *ground*.

O sinal de *clock*, embora previsto na especificação original, não é transmitido, sendo necessário que tanto receptor quanto transmissor operem numa mesma velocidade (*baud rate*), selecionada antes do início da comunicação. Embora existam rotinas que detectam qual *baud rate* está sendo utilizado e o ajustam automaticamente, estas necessitam que um determinado caractere, seja o primeiro caractere recebido. Desta forma, caso um caractere diferente do previsto seja enviado, transmissor e receptor poderão estar operando a velocidades distintas, resultando na interrupção da comunicação. Estima-se aceitável uma discrepância de até cerca de 2% entre a frequências do receptor e do transmissor (MAXIM-DALLAS, 2003, [21]).

Tabela 2-3 – Sinais da interface serial presentes no conector DE-9 – dispositivo DTE.

PINO	SINAL		DIREÇÃO	FUNÇÃO
3	TD	<i>Transmitted Data</i>	Saída	canal de transmissão de dados para DCE.
2	RD	<i>Received Data</i>	Entrada	canal de recepção de dados do DCE.
7	RTS	<i>Request to Send</i>	Saída	requisição de envio de dados para DCE.
8	CTS	<i>Clear to Send</i>	Entrada	DCE permite que DTE envie de dados.
6	DSR	<i>DCE Ready</i>	Entrada	DCE informa que está disponível.
4	DTR	<i>DTE Ready</i>	Saída	DTE informa que está disponível.
9	RI	<i>Ring Indicator</i>	Entrada	Indica que DCE está chamando.
1	CD	<i>Carrier Detect</i>	Entrada	Indicador de detecção de portadora.

Os dados são transmitidos sequencialmente, em quadros, conforme mostrado na Figura 2-6. Cada quadro é composto de um bit de início, bits de dados (entre 6 a 8) e um bit de parada. Os dados são transmitidos com o bit menos significativo primeiro.



Fonte: (MAXIM, 2001, p. 1, [20])

Figura 2-6 – Quadro típico de dados (um byte) TIA/EIA-232.

Implementações mais simples utilizam apenas o canal de transmissão de dados (TD), o canal de recepção de dados (RD) e a referência de terra para estabelecer comunicação bidirecional. Neste caso, o controle do fluxo de dados é feito por *software*, ou simplesmente removido. Há ainda situações em que utiliza-se a comunicação serial, porém utilizando sinalização de tensão na faixa 0 a 5V, compatível com dispositivos de lógica TTL. Normalmente é utilizado quando deseja-se estabelecer comunicação entre circuitos próximos, por exemplo, numa mesma placa de circuito impresso.

Recentemente a interface serial vem sendo substituída nos computadores pessoais por outras interfaces com características elétricas diferentes e incompatíveis, como a porta USB. Neste caso é possível utilizar um adaptador Serial-USB, que encapsula os dados transmitidos, recebidos e os sinais da interface serial dentro do protocolo USB, criando uma porta serial emulada no computador. Uma vez que para o aplicativo sendo executado no computador a porta serial emulada é semelhante à porta serial tradicional, a migração dos aplicativos torna-se transparente.

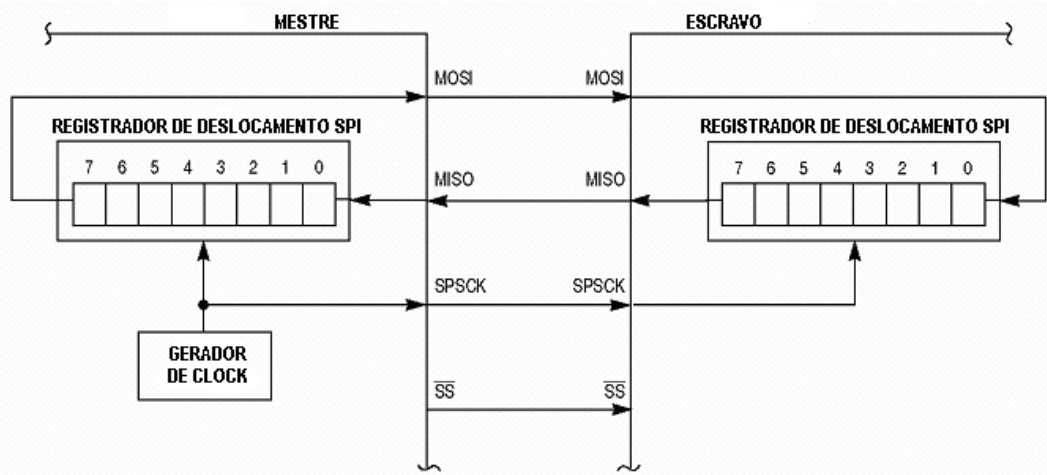
2.3.2 Protocolo de Comunicação SPI

A interface de comunicação SPI – *Serial Peripheral Interface* foi desenvolvida pela Motorola e atualmente é mantida pela Freescale Semiconductor. Utiliza barramento do tipo mestre-escravo, serial, dúplex e síncrono. Uma vez que a Freescale não fornece diretamente as especificações da interface SPI, utiliza-se a documentação referente ao controlador SPI incorporado em seus microcontroladores, contida no documento *Serial Peripheral Interface (SPIV3) Block Description* (FREESCALE, 2001, [22]).

A interface SPI é composta por uma linha de transmissão de dados do dispositivo mestre para o escravo (MOSI – *Master Out Slave In*), uma do escravo para o mestre (MISO – *Master In Slave Out*), uma linha de sincronização contendo o sinal de *clock* (SCLK) e uma linha de seleção para cada dispositivo escravo conectado ao barramento (SSn – *Slave Select*). Embora haja suporte a vários dispositivos mestre compartilhando o barramento com diversos dispositivos escravo, a configuração mais comum utiliza apenas um dispositivo mestre.

A conexão entre um dispositivo mestre e um dispositivo escravo é mostrada na Figura 2-7. A comunicação é iniciada pelo dispositivo mestre, que seleciona um dos dispositivos escravo do barramento através do sinal ‘#SS’, ativo em nível baixo. Um único dispositivo escravo pode ser selecionado a cada instante. Observa-se que há um registrador de deslocamento em ambos dispositivos. Conforme o mestre envia um caractere e fornece o sinal de clock para o dispositivo escravo, simultaneamente o conteúdo do registrador de

deslocamento do dispositivo escravo é transmitido para o dispositivo mestre. Desta forma, há uma troca, iniciada pelo dispositivo mestre, entre o conteúdo dos registradores (CATSOULIS, 2005, [23]).



Fonte: (FREESCALE, 2008, p. 215, [11])

Figura 2-7 – Conexão entre dispositivos SPI.

2.3.3 Protocolo de Comunicação I²C

A interface de comunicação I²C – Inter Integrated Circuit, foi desenvolvida pela Philips Semiconductor, atualmente NXP Semiconductor. Permite a interconexão de diversos circuitos integrados através de apenas ‘dois fios’. Utiliza uma linha de ‘clock’ (SCL) e uma linha de dados (SDA), além da referência de terra compartilhada. As especificações do barramento estão disponíveis no ‘I²C-bus specification and user manual’ [24]. Este documento introduz a nomenclatura a seguir para descrever as funções do barramento I²C.

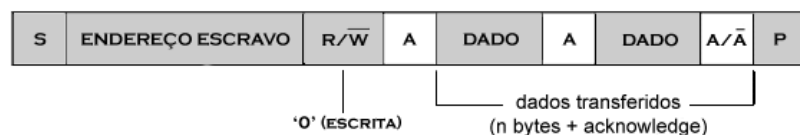
- Transmissor: dispositivo que envia dados no barramento.
- Receptor: dispositivo que recebe dados no barramento.
- Mestre: dispositivo que inicia a transferência, gera o sinal de *clock* SCL, e encerra a transmissão.
- Escravo: dispositivo endereçado pelo mestre.
- Multi-mestre: mais de um dispositivo mestre pode tentar controlar o barramento sem no entanto corromper a mensagem.
- Arbitração: Procedimento para garantir que, caso mais de um mestre tente simultaneamente controlar o barramento, somente um obtenha o acesso e sua mensagem não seja corrompida.
- Sincronização: Procedimento para sincronizar o sinal de clock de dois ou mais dispositivos.

São definidos quatro modos (frequências) de operação: modo padrão – Sm (até 100kHz); modo rápido – Fm (até 400kHz); modo rápido ‘plus’ – Fm+ (até 1Mbps); e modo de alta velocidade – Hs (até 3.4MHz).

O barramento permite a conexão de múltiplos dispositivos mestres e de múltiplos dispositivos escravos. Entretanto, em sistemas mais simples utiliza-se apenas um dispositivo mestre no barramento. Cada dispositivo escravo conectado ao barramento possui pré-configurado um endereço, inclusive os dispositivos mestre que operam como escravo também. A quantidade máxima de dispositivos no barramento é limitada pelo endereçamento utilizado (sete ou dez bits) e pela capacitância das linhas SCL e SDA, esta última sendo normalmente o fator limitante. Dezesesseis endereços são reservados para funções específicas do barramento e futuras atualizações. Assim, utilizando-se apenas dispositivos com endereçamento de sete bits, chega-se ao limite teórico de 112 dispositivos no barramento. Normalmente o dispositivo apresenta uma faixa de endereços selecionáveis através do nível lógico de um pino externo.

A condição de início (*START*) consiste de transição do nível alto para o nível baixo da linha SDA enquanto a linha SCL encontra-se em nível alto. A condição de término (*STOP*) consiste na transição do nível baixo para o nível alto de SDA enquanto SCL encontra-se em nível alto. Todas as transações no barramento iniciam-se com a condição de início e podem ser terminadas por uma condição de término. Somente o dispositivo mestre pode gerar as condições de início e término.

A Figura 2-8 apresenta a sequência que deve ser gerada pelo dispositivo mestre na linha SDA para escrever no dispositivo escravo. Os instantes em que SDA é controlada pelo mestre (direção do mestre para o escravo) estão com fundo cinza; aqueles em que SDA é controlada pelo escravo (direção do escravo para o mestre) estão em fundo branco.



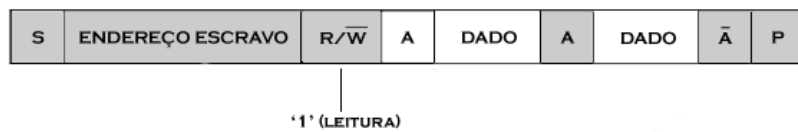
Fonte: (NXP SEMICONDUCTOR, 2007, p. 15, [24])

Figura 2-8 – Sequência referente ao dispositivo mestre escrevendo no dispositivo escravo.

Uma transação sempre é iniciada pelo dispositivo mestre, que gera a condição de início, denotada pela letra ‘S’, seguida pelo endereço do dispositivo de destino. Em seguida, o dispositivo mestre indica que pretende escrever, enviando um nível baixo para o bit ‘R/W’. O dispositivo escravo indica que foi selecionado, gerando um sinal de ACK. O mestre envia os

dados, sendo gerado pelo escravo um ACK (A) para cada *byte* recebido. A transmissão é encerrada com a geração do sinal de término (P).

Analogamente, a Figura 2-9 apresenta a sequência para a leitura de dados do dispositivo escravo. A sequência é similar à de escrita, porém o ‘R/W’ é substituído pelo nível alto (indicando que o procedimento é de leitura) e o dispositivo escravo transmite os dados. Neste caso, é o dispositivo mestre que gera o ACK a cada *byte* recebido. Ao término da transmissão, novamente uma condição de término é gerada.



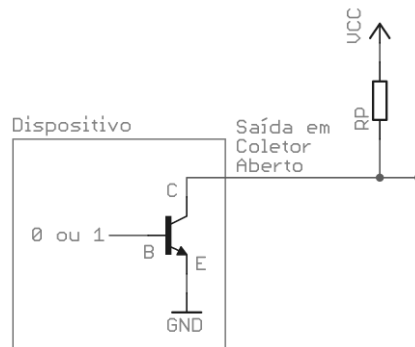
Fonte: (NXP SEMICONDUCTOR, 2007, p. 15, [24])

Figura 2-9 – Sequência referente ao dispositivo mestre lendo do dispositivo escravo.

Quando dois ou mais dispositivos mestres tentam controlar simultaneamente o barramento, um mecanismo de arbitração define qual dispositivo obterá acesso e qual terá de aguardar o término da transação atual para tentar novamente acessar o barramento. Para tanto, cada dispositivo mestre verifica bit a bit se o nível atual do barramento corresponde àquele gravado. O dispositivo que enviar um bit ‘1’ (nível alto) e verificar que o barramento encontra-se em ‘0’ (nível baixo) perde a arbitração e interrompe a transação atual. Observa-se que não há conflito entre os dois mestres, uma vez que a transação prioritária não sofreu alteração. Caso o dispositivo que perdeu a arbitração seja do tipo mestre-escravo, este deve imediatamente entrar no modo escravo, uma vez que há a possibilidade de o outro dispositivo mestre (vencedor da arbitração) estar endereçando-o. Um aspecto relevante é que a arbitração leva em conta o endereço do dispositivo endereçado, e não uma prioridade fixa atribuída a determinado dispositivo mestre.

A arbitração decorre diretamente do uso de saídas do tipo coletor aberto (ou *open collector*), mostrada na Figura 2-10. Em dispositivos com saídas deste tipo, um dos terminais do transistor é disponibilizado externamente. Normalmente este terminal é conectado à tensão de alimentação através de um resistor. Caso o transistor seja bipolar, este terminal é o coletor, correspondendo a uma saída em coletor aberto. Quando o transistor está em corte, o resistor de *pullup* RP mantém a saída em nível alto. Quando o transistor está na saturação a tensão entre os terminais coletor e emissor (ou entre dreno e fonte) é pequena, fazendo com que a saída seja grampeada para o terra. Observa-se que uma corrente flui da alimentação para o terra através do resistor RP. Desta forma, o dimensionamento do resistor RP é importante para

que não seja drenada corrente excessiva. Um valor típico do resistor RP para aplicações I²C é de 10kΩ.

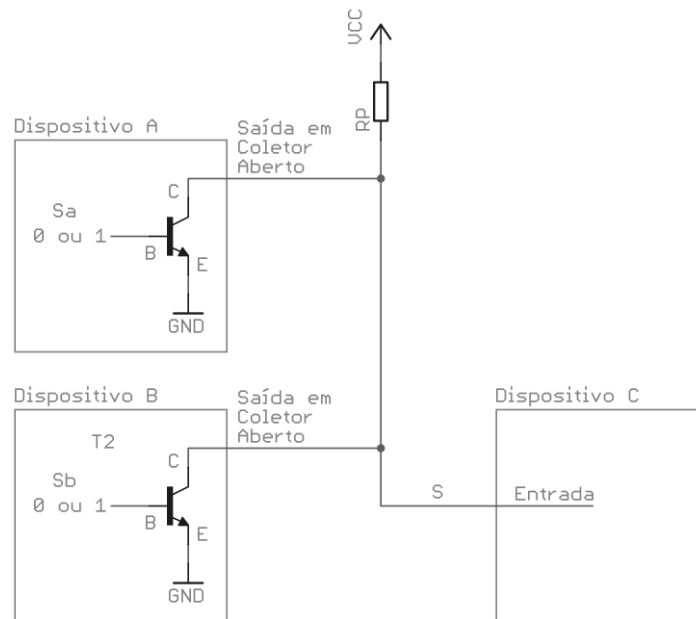


Fonte: (SCHERZ, 2006, p. 675, [25])

Figura 2-10 – Saída em coletor aberto.

O uso de lógica digital do tipo coletor aberto ou dreno aberto permite que diversas saídas sejam fisicamente conectadas, conforme mostrado na Figura 2-11. Por exemplo, sejam o dispositivo A e o dispositivo B mestres no barramento I²C, e o dispositivo C escravo. O valor que os dispositivos A e B desejam escrever no barramento é dado respectivamente por Sa e Sb, e o valor presente no barramento é dado por S. Quando Sa e Sb estão em nível alto, o resistor de pullup mantém a saída S em nível alto. Quando Sa está em nível baixo, e Sb em nível alto, a saída S vai para o nível baixo. Desse modo, uma vez que o valor presente no barramento é diferente do pretendido por Sb, este perde a arbitração. Basta que apenas uma das saídas esteja em nível baixo para que a saída vá para este nível, o que equivale à função lógica ‘E’ (ou *wired and*): $S = S_a \cdot S_b \cdot \dots \cdot S_n$.

Entretanto, esta característica torna-se o calcanhar de aquiles da interface I²C, uma vez que um dispositivo defeituoso pode interromper por completo a comunicação no barramento, bastando para isso manter a linha de dados SDA (ou a linha de *clock* SCL no caso de falha de um dispositivo mestre) em nível baixo. Neste caso, deve-se iniciar um procedimento de *bus clear*, ou de limpeza do barramento, que consiste no envio de 9 pulsos de *clock* pelo dispositivo mestre. Se após este procedimento a comunicação continuar interrompida, pode ser necessária uma reinicialização forçada, removendo e reaplicando-se a alimentação do dispositivo [24].



Fonte: (SCHERZ, 2006, p. 675, [25])

Figura 2-11 – Conexão de múltiplas saídas em coletor aberto.

Encontram-se disponíveis comercialmente diversos dispositivos I^2C , tais como sensores de temperatura, portas digitais de E/S, memórias não voláteis, ADCs, DACs, relógios de tempo real, dentre outros.

2.3.4 Malha de corrente 4 a 20mA

Neste método de transmissão, unidirecional, determina-se uma faixa para o parâmetro que deseja-se transmitir. Desta forma, caso o parâmetro seja a temperatura na faixa 0 a 100°C, atribui-se ao limite inferior da faixa a corrente de malha de 4mA, e ao valor 100°C o valor 20mA, sendo os valores intermediários mapeados dentro desta faixa de corrente. O uso de um sinal de corrente resulta em maior imunidade a diferenças entre o nível de terra do transmissor e do receptor, benefício semelhante à transmissão diferencial. Outra característica consiste em se poder verificar a integridade da conexão entre transmissor e receptor, pois se a malha for interrompida, serão medidas correntes próximas a 0mA. Em operação normal, a corrente nunca deve ser inferior a 4mA (CROVELLA, 1999, p. 2342, [18]). Uma vez que normalmente os sistemas de aquisição de dados digitalizam um sinal de tensão, faz-se necessário utilizar um dispositivo com função de transresistência. Um resistor pode ser inserido em série com a malha de corrente para efetuar a conversão corrente-tensão.

2.4 CONVERSOR ANALÓGICO-DIGITAL

“O processo de conversão de sinais contínuos no tempo (analógicos) em uma sequência digital que possa ser processada por um sistema digital requer a quantização dos valores amostrados em um número finito de níveis e a representação de cada nível por um número de bits” (PROAKIS, MANOLAKIS, 1996, [26]). O dispositivo responsável por esta tarefa é o conversor analógico digital.

O conversor analógico-digital gera uma representação digital de uma grandeza analógica. A resolução Δ do conversor é dada pela equação (2-4) (KOSONOCKY, XIAO, [27]), onde F é a faixa e N é a resolução expressa em bits do conversor.

$$\Delta = \frac{F}{2^N} \quad (2-4)$$

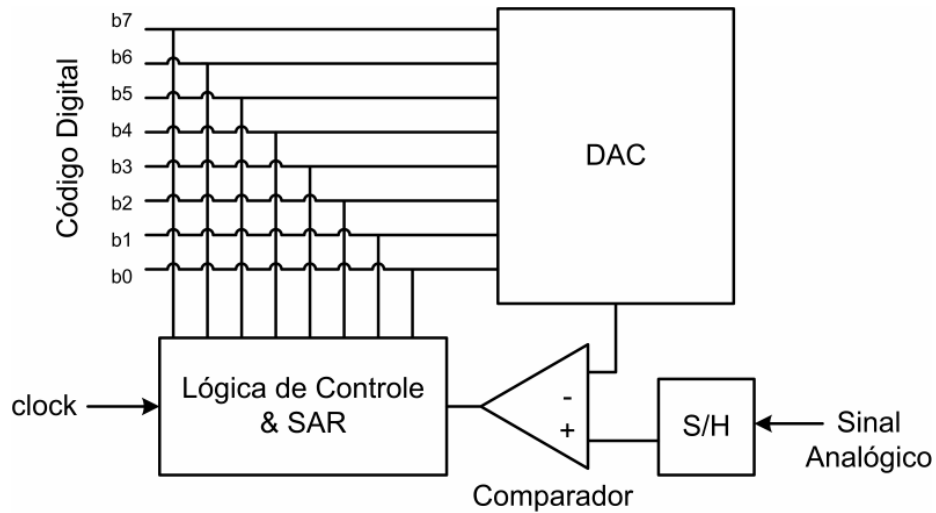
Assim, um conversor de 10 bits, com faixa de 0 a 3,3V tem resolução de 3.22mV. Um determinado valor de saída do conversor corresponde ao intervalo centrado no nível $\pm \Delta/2$. Por exemplo, o resultado quantizado 512 (10 0000 0000b) corresponde ao intervalo de entrada $512 \times 3,22\text{mV} \pm 1,61\text{mV} = 1,65\text{V} \pm 1,61\text{mV}$. A resolução pode ser expressa também pelo termo lsb, que significa *least significant bit*.

O *sample and hold* é o circuito do A/D que armazena por um período de tempo suficiente a tensão de entrada do sinal, mediante controle digital. “A função do *sample and hold* é amostrar continuamente o sinal de entrada, mantendo-o constante enquanto o conversor obtém a sua representação digital” [26]. Um exemplo de *sample and hold* consiste de um capacitor e chaves analógicas controladas digitalmente. Por meio do controle das chaves, conecta-se o capacitor ao sinal de entrada (modo *Sample* ou amostragem). Ao se desconectar o capacitor, este apresenta em seus terminais a tensão do sinal de entrada (modo *Hold* ou armazenagem). Pode-se então, por meio de outras chaves, digitalizar esta tensão em estágios subsequentes do ADC. A frequência com que a chave conecta o circuito ao sinal de interesse corresponde à taxa de amostragem.

2.4.1 ADC do tipo aproximações sucessivas

O conversor analógico-digital do tipo aproximações sucessivas utiliza um processo iterativo de comparações para digitalizar o sinal. Conforme mostrado na Figura 2-12, é composto de um circuito de *sample and hold*, comparador, conversor digital analógico e de lógica de controle. O processo de conversão consiste em comparar analogicamente o valor

armazenado no circuito de *sample and hold* com uma sequência de valores gerados internamente pelo DAC e atualizados a cada pulso de *clock*.



Fonte: (KOSONOCKY; XIAO, 1999, p. 113, [27])

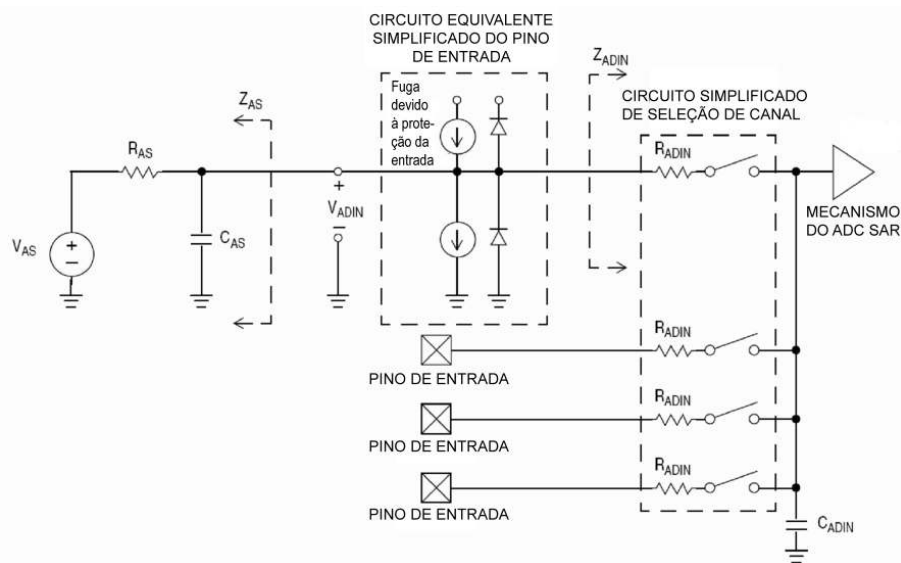
Figura 2-12 – Arquitetura do conversor A/D do tipo aproximações sucessivas.

Supondo que se deseje digitalizar uma tensão de $V_{in}=2,00V$ utilizando um conversor de 8 bits e referência de $V_{ref}=3,30V$. Inicia-se o processo com todos os bits do DAC em nível baixo, exceto o bit mais significativo (10000000b). Assim, a tensão de saída do DAC será 1,65V, correspondente a aproximadamente metade de V_{ref} . Como 2,00V é maior que 1,65V, a comparação resultará em '1', indicando que o valor do bit deve ser mantido. A conversão procede testando no próximo pulso de *clock* o bit adjacente, ou seja, alimentando o DAC com o valor 11000000b, equivalente a 2,48V (o primeiro bit sendo o resultado da comparação anterior, e o segundo o bit testado no momento). Neste caso, a comparação resultará '0', uma vez que 2,48V é superior a 2,00. Alimenta-se o DAC com 10100000b, e assim sucessivamente. Observa-se que para uma conversão com n bits de resolução são necessários n ciclos de *clock*.

2.4.2 ADC Interno do Microcontrolador

O microcontrolador MC9S08QG8 incorpora conversor analógico digital do tipo aproximações sucessivas com oito canais de 10 bits. Os canais compartilham o mesmo conversor, através de um multiplexador analógico. Adicionalmente, podem ser selecionados um diodo sensor de temperatura e um referência de tensão bandgap internos ao microcontrolador. A tensão de alimentação do microcontrolador é conectada internamente à referência do ADC. O ADC pode operar no modo contínuo ou no modo de conversão unitária.

No modo contínuo, ao término de uma conversão inicia-se automaticamente uma nova conversão. No modo de conversão unitária, a conversão é iniciada assim que é solicitada, sendo necessário aguardar o tempo de conversão até que o resultado esteja disponível. O circuito simplificado da entrada analógica do microcontrolador, com um canal conectado a uma fonte de tensão V_{AS} , é mostrado na Figura 2-13. A tensão fornecida pelo sensor conectado ao canal de ADC é representada por V_{AS} e a resistência interna desta fonte por R_{AS} . A impedância de entrada do ADC é Z_{ADIN} , e a tensão observada no pino de entrada do ADC é V_{ADIN} .



Fonte: (FREESCALE, 2008, p. 115, [11])

Figura 2-13 – Circuito simplificado do pino de entrada analógica do microcontrolador.

Os diodos mostrados na Figura 2-13 têm a função de proteger o microcontrolador de tensões eventuais aplicadas ao pino, superiores à tensão de alimentação e inferiores ao nível de terra. Quando a tensão aplicada ao pino do microcontrolador supera a tensão de alimentação, o diodo superior permite que flua uma corrente do pino para a fonte de alimentação. Desta forma, evita-se que a tensão no pino ultrapasse a tensão de alimentação. Analogamente, quando uma tensão inferior à referência de terra é aplicada ao pino, o diodo inferior permite que uma corrente flua do terra para o pino, impedindo que a tensão no pino seja inferior à referência de terra.

O resultado da conversão é disponibilizado nos registradores ADCRH e ADCRL. Caso o conversor seja configurado para oito bits, apenas ADCRL é utilizado. Para conversões de 10 bits, ADCRH retorna os dois bits mais significativos e ADCRL os oito bits menos significativos. Há um intertravamento que congela o conteúdo de ADCRL assim que o

registrador ADCRH seja lido. Desta forma, impede-se que o programa leia a parte alta do resultado de uma conversão e a parte baixa de conversão diferente. Assim que ADCRL é lido, permite-se que os registradores sejam atualizados com uma nova conversão.

2.5 TRANSDUTORES DE TEMPERATURA

Transdutor é “qualquer dispositivo capaz de transformar um tipo de sinal em outro tipo, com o objetivo de transformar uma forma de energia em outra, possibilitar o controle de um processo ou fenômeno, realizar uma medição,[...]” (HOLANDA, 1986, [28]).

Existem varios métodos e transdutores destinados à medição de temperatura. O bimetal une duas tiras de metais com coeficientes de expansão térmica diferentes, o que resulta em uma deflexão sensível à temperatura. O termopar explora o efeito Seebeck, que consiste na geração de uma tensão quando dois metais dissimilares são postos em contato. Um termômetro por bulbo de vidro utiliza utiliza uma haste contendo substância líquida com coeficiente de expansão térmica relativamente grande quando comparado com o recipiente (haste) que o contém. Normalmente a haste apresenta uma escala graduada que permite a leitura da temperatura. Termistores utilizam resistores com sensibilidade à temperatura relativamente grande. Outras técnicas utilizam transdutores de fibra ótica, transdutores sensíveis à radiação infravermelha emitida pelos objetos, características da junção semicondutora p-n, dentre outros.

2.5.1 *Sensor de Temperatura por Junção Semicondutora p-n*

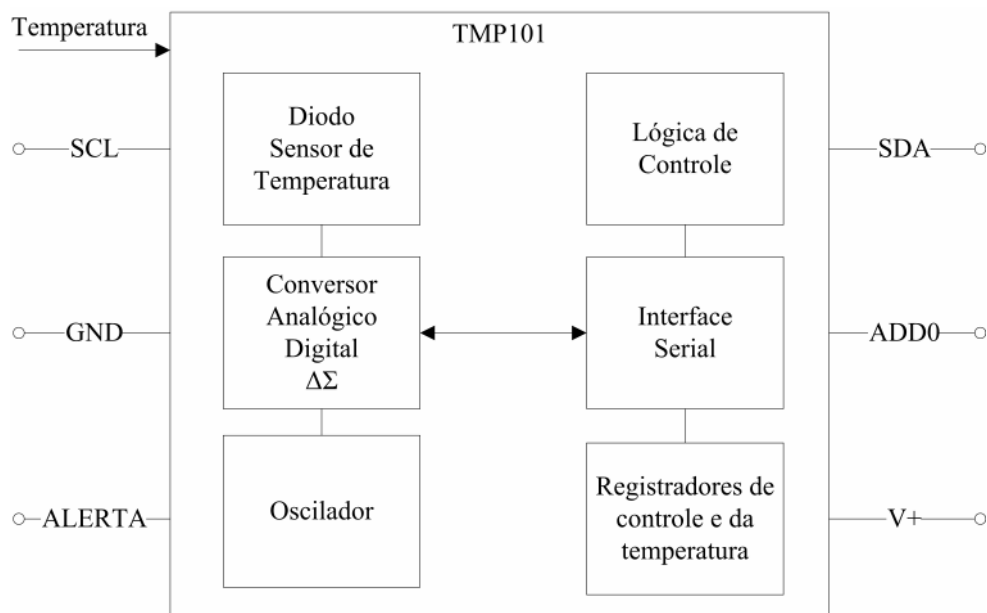
Sensores de temperatura podem ser produzidos explorando as características de uma junção semicondutora p-n. A sensibilidade da junção decorre da equação que modela a junção semicondutora, Eq. (2-5). A faixa de operação do sensor é tipicamente de -55°C a $+150^{\circ}\text{C}$. Grande parte dos sensores utilizam um transistor bipolar conectado como um diodo, ou seja, com a junção base-coletor em curto. Uma corrente constante através da junção base-emissor produz uma tensão V_{be} que é uma função linear da temperatura absoluta. A queda de tensão direta do diodo tem um coeficiente de aproximadamente $2\text{mV }^{\circ}\text{C}^{-1}$ (FRANK, 1999, [29]).

$$V_F = V_{be} = \left(\frac{kT}{q} \right) \ln \left(\frac{I_F}{I_S} \right) \quad (2-5)$$

Na Eq. (2-5) k é a contante de Boltzmann ($1.38 \times 10^{-23} \text{ J K}^{-1}$), T é a temperatura em Kelvins, q é a carga do elétron ($1.6 \times 10^{-19} \text{ C}$), I_F é a corrente direta e I_S é a corrente reversa de saturação da junção, em amperes.

2.5.2 Sensor de Temperatura com interface I²C

O sensor de temperatura com interface I²C TMP101, fabricado pela Texas Instruments, opera na faixa de -55 a 125°C com exatidão^d de 3°C (TEXAS, 2002, [30]). A resolução é seleccionável de 0,5°C a 0,0625°C. Disponível no encapsulamento SOT23-6 (cerca de 3mm x 3mm), o chip contém diodo sensor de temperatura, ADC do tipo delta-sigma, oscilador e lógica de controle internos, conforme diagrama da Figura 2-14.



Fonte: (TEXAS INSTRUMENTS, 2002, p. 1, [30])

Figura 2-14 – Diagrama do sensor de temperatura TMP101.

O dispositivo opera com alimentação entre 2,7 e 5,5V (pino V+). O endereço do chip no barramento I²C é configurado através do nível lógico do pino ADD0 (nível baixo, nível alto e ‘pino em aberto’), sendo portanto possível utilizar até três sensores TMP101 no mesmo barramento. Para reduzir o consumo do chip, o nível do pino ADD0 é amostrado no momento em que a alimentação é fornecida ao chip. Desta forma, caso seja utilizado um resistor de *pullup* para configurar o endereço do sensor, o pino será desconectado assim que o endereço for considerado, interrompendo a corrente drenada pelo *pullup*. Outra vantagem deste procedimento é que o endereço do TMP101 no barramento I²C não será alterado caso haja

^d Termo utilizado no documento original em língua inglesa: accuracy. Tradução conforme o Vocabulário Internacional de Termos Fundamentais e Gerais de Metrologia – VIM. (INMETRO, 2007, p. 26, [31])

flutuações posteriores nesse pino. Os sinais SCL e SDA correspondem a *clock* e dados do barramento I²C. O pino Alerta torna-se ativo quando a temperatura excede um limite pré-programado.

O TMP101 possui quatro registradores internos: Configuração, Thigh, Tlow e Temperatura. O registrador ‘configuração’ permite a escolha da resolução do resultado da conversão da temperatura (de 9 a 12 bits); o modo de operação e a polaridade do pino Alerta e a quantidade de conversões consecutivas acima de Thigh necessárias para o acionamento do sinal de Alerta. Permite ainda operar o TMP101 em um modo de baixa energia, neste caso é necessário escrever no bit OS cada vez que se desejar iniciar uma conversão.

Com a resolução de 9 bits, o tempo de conversão é de cerca de 40ms (incremento de 0,5°C). Para cada bit adicional, dobra-se o tempo de conversão, chegando a 320ms para a resolução de 12 bits (incremento de 0,0625°C). Desta forma, há um compromisso entre a resolução escolhida e o tempo de conversão.

Os registradores Thigh e Tlow, de 16 bits cada, quando escritos armazenam os valores de referência para a função Alerta. Quando lidos, retornam o valor atualmente configurado. O formato de Thigh e Tlow é idêntico ao do registrador Temperatura, mostrado na Tabela 2-4.

O registrador Temperatura, de 16 bits, do tipo somente leitura, armazena o valor medido da temperatura em °C no formato complemento a dois. Uma vez que o TMP101 tem resolução máxima de 12 bits, os quatro bits menos significativos são sempre lidos nulos, conforme visto na Tabela 2-4. O *byte* 1 retorna a parte inteira da temperatura, e o *byte* 2 retorna a parte fracionária.

Quando a resolução selecionada é de 12 bits, T11 a T0 são significativos. Conforme se reduz a resolução, T2, T1 e T0 passam a retornar valores nulos. Assim, com 9 bits, o valor da temperatura estará armazenada em T11 a T3, sendo os demais bits sempre lidos como zero e portanto não significativos.

Tabela 2-4– Alinhamento dos dados no registrador Temperatura do TMP101

BYTE 1								BYTE 0							
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
T11	T10	T9	T8	T7	T6	T5	T4	T3	T2	T1	T0	0	0	0	0

A função Alerta do TMP101 informa, através do pino correspondente, quando a temperatura excedeu o limite programado em Thigh. O alerta somente é desabilitado quando

o sensor ler uma temperatura com valor inferior ao configurado em Tlow. Desta forma, Thigh e Tlow determinam uma histerese para a função Alerta.

Um método simples de conversão do resultado para °C consiste em tratar o valor lido como um valor inteiro com sinal de 16 bits (*signed integer*). Neste caso, basta multiplicar o valor lido pela suposta variação de um bit, neste caso 1/256 °C.

2.5.3 Sensor de Temperatura Interno do Microcontrolador

O microcontrolador possui um diodo destinado à medição da temperatura. Este diodo está conectado internamente a um canal de ADC do microcontrolador. Uma estimativa da temperatura do chip pode ser obtida através da equação (2-6).

$$T = 25 - \left(\frac{V_{Temp} - V_{Temp25}}{m} \right) \quad (2-6)$$

Onde T é a temperatura do chip em °C, V_{temp} é a tensão medida pelo ADC em mV, V_{Temp25} é uma constante com valor estimado em 701.2mV e m é a variação de tensão do diodo sensor interno com a temperatura. Para a faixa -40 a 25°C, utiliza-se $m=1.646\text{mV}/^\circ\text{C}$; para a faixa 25°C a 85°C, $m=1,769\text{mV}/^\circ\text{C}$. Os valores de V_{Temp25} e m são fornecidos na seção A.10 da folha de especificações do microcontrolador (FREESCALE, 2008, [11]).

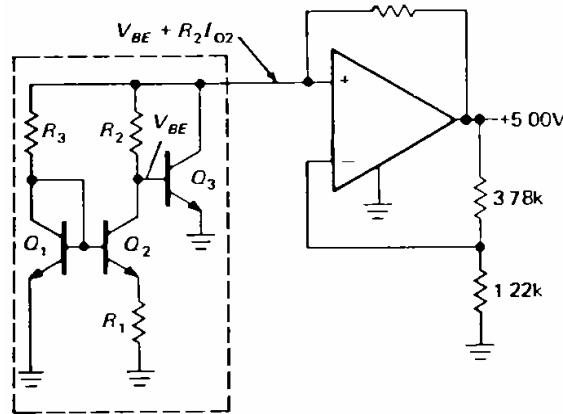
O cálculo da temperatura consiste em ler o canal de ADC referente ao sensor de temperatura. Calcula-se V_{temp} , de acordo com a tensão de referência utilizada pelo ADC. Se V_{temp} for superior à constante V_{temp25} , substitui-se na equação (2-6) o valor de m para a faixa -40 a 25°C. Caso contrário, utiliza-se o valor referente à faixa 25 a 85°C. Utilizando-se alimentação de 3.3V, a resolução do ADC é de 3,22mV. Desta forma, cada incremento no resultado do ADC corresponde a 1,8°C (25°C a 85°C) ou 1,9°C (-40°C a 25°C).

2.6 REFERÊNCIA DE TENSÃO DO TIPO BANDGAP

A referência de tensão *bandgap* visa reduzir a dependência da tensão de referência com a temperatura. A Figura 2-15, ilustra o circuito de uma fonte de referência do tipo Bandgap. Os transistores Q_1 e Q_2 apresentam características semelhantes, e normalmente estão fisicamente dispostos próximos no circuito integrado. As junções base-emissor são polarizadas com densidades de corrente diferentes, porém de forma que a razão entre as densidades de corrente sejam constantes.

A operação do circuito consiste em somar duas tensões, uma com coeficiente de variação com a temperatura positivo, outra com coeficiente de variação com a temperatura

negativo. O circuito é ajustado de forma a obter um circuito com sensibilidade à temperatura próximo de zero. Gera-se uma fonte de corrente, e utilizando-se um resistor, obtém-se a tensão de referência.



Fonte: (HOROWITZ, 1989, p. 335, [32])

Figura 2-15 – Exemplo de circuito de referência de tensão do tipo *bandgap*.

2.6.1 Referência de Tensão Interna do Microcontrolador

O microcontrolador MC9S08QG8 possui referência de tensão do tipo bandgap, de valor central 1,20V e variação $-0,02\text{V} / +0,01\text{V}$. Para utilizar a referência, habilita-se um *buffer* de tensão (amplificador com ganho unitário) conectado à saída da referência através do bit BGBE (*BandGap Buffer Enable*). A função do *buffer* é fornecer em sua saída uma tensão igual à gerada pela referência, drenando pouca corrente diretamente da referência. Isto é obtido devido à alta impedância de entrada do *buffer*. A saída do *buffer* está internamente conectada a um canal do ADC, e funciona ao longo de toda a faixa de operação do microcontrolador (1,8 a 3,6V). Embora esta referência não possa ser utilizada como referência para o ADC, sua leitura com o ADC permite determinar a tensão de alimentação do microcontrolador.

2.6.2 Regulador de tensão com referência interna

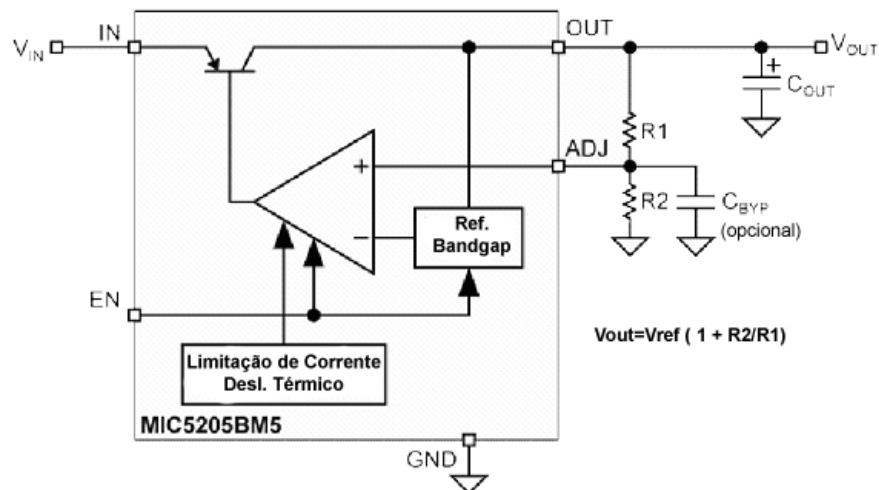
A família de reguladores MIC5205 [33], fabricados pela Micrel, utilizam uma referência interna do tipo bandgap para gerar uma tensão regulada, suportando correntes de saída de até 150mA. Estão disponíveis com tensão de saída fixa (3,3V, 5V, dentre outras) e em versão ajustável. Apresentam tensão de *drop out* máxima de 350mV (diferença entre a mínima tensão de entrada e a tensão de saída) e exatidão de 1%. Apresentam pino de

habilitação, permitindo que um microcontrolador, por exemplo, controle o fornecimento de energia aos circuitos alimentados pelo regulador.

A tensão de saída da versão ajustável é definida pelo divisor de tensão formado por R_1 e R_2 , conforme mostrado na Figura 2-16, de acordo com a equação (2-7).

$$V_{Saída} = 1.242V \cdot \left(\frac{R_2}{R_1} + 1 \right) \quad (2-7)$$

Observa-se que uma resistência de valor $R_1 + R_2$ é conectada diretamente à saída do regulador, drenando permanentemente uma corrente. Desta forma, projeta-se para que a energia dissipada no divisor seja mínima, o que é obtido utilizando-se valores altos para R_1 e R_2 , limitados a $470k\Omega$, segundo recomendação do fabricante [33].



Fonte: (MICREL, 2006, p. 6, [33])

Figura 2-16 – Regulador ajustável de tensão MIC5205.

Embora o regulador suporte tensões de entrada de até 16V, na prática esta tensão é limitada pela máxima dissipação térmica do encapsulamento (cerca de 450mW) de acordo com a corrente de operação, conforme expresso pela equação (2-8). O termo I_{terra} refere-se à corrente quiescente do regulador, de valor máximo 2.5mA [33].

$$P_{dissipada} = (V_{entrada} - V_{saida}) \cdot I_{saida} + V_{entrada} \cdot I_{terra} \quad (2-8)$$

Por exemplo, o regulador fixo com tensão de saída 3,3V, sob carga máxima (150mA), pode operar com tensão de entrada de até 6.2V [33].

$$450mW = (V_{entrada} - 3.3V) \cdot 150mA + V_{entrada} \cdot 2.5mA \quad \therefore \quad V_{entrada} = 6.2V$$

Para que o circuito opere de forma estável, recomenda-se o uso de capacitor de saída de tântalo, devido à sua baixa ESR (*Equivalent Series Resistance*, ou resistência equivalente em série). Entretanto, capacitores com baixíssima ESR devem ser evitados nesta aplicação, pois podem causar oscilações na saída.

3

UTILITÁRIOS DE DESENVOLVIMENTO

Ao longo do desenvolvimento do trabalho, diversos programas foram utilizados em tarefas de projeto e fabricação de placas de circuito impresso, programação de microcontroladores e desenvolvimento de interfaces com o usuário. Destacam-se o compilador *Codewarrior* da linguagem C e *assembly* para microcontroladores Freescale HCS08, o programa CAD para projeto de placas de circuito impresso *Eagle Layout Editor* e o ambiente de programação gráfica *Labview*, descritos sucintamente neste capítulo.

3.1 COMPILADOR FREESCALE CODEWARRIOR

O Codewarrior é um ambiente de desenvolvimento integrado que inclui compilador para as linguagens C e *assembly*, voltado para a família de microcontroladores de oito bits fabricados pela Freescale. A Figura 3-1 mostra a tela de interface do Codewarrior. A ferramenta *Processor Expert*, integrada ao Codewarrior, auxilia na configuração do microcontrolador, permitindo, por exemplo, selecionar a origem (cristal externo, *clock* externo, oscilador interno) e frequência de operação do microcontrolador, habilitar interrupções, dentre outras. Adicionalmente, incorporado ao *Processor Expert*, há módulos dedicados denominados *beans* para diversos periféricos disponíveis no microcontrolador. Estes módulos têm a função de inicialização do respectivo periférico, além de incorporar bibliotecas com as funções necessárias ao uso do módulo.

Desta forma, uma vez definida a configuração, seleciona-se no menu a opção *generate code*, a qual gera uma função de inicialização do microcontrolador, usualmente denominada

`PE_low_level_init()`. Esta função é inserida no início da função padrão de ponto de entrada da linguagem C: `main()`. Ao ser chamada, a função configura os registradores internos do microcontrolador de acordo com as configurações selecionadas no Codewarrior. São geradas também funções inerentes aos diversos módulos adicionados ao projeto. Por exemplo, a utilização do módulo serial resulta na disponibilização, dentre outras, da função `AS1_SendChar()`, que envia um *byte* através da interface serial.

Caso haja interrupções associadas ao módulo, um arquivo destinado ao tratamento das interrupções, normalmente denominado `events.c` é criado. Caso o arquivo já exista, apenas a seção referente ao novo evento de interrupção é adicionada. Por exemplo, caso se adicione ao projeto Codewarrior o módulo referente timer, automaticamente será gerada a função `TI1_OnInterrupt()`. O código desta função será executado periodicamente, a cada interrupção, com o intervalo especificado durante a configuração do módulo.

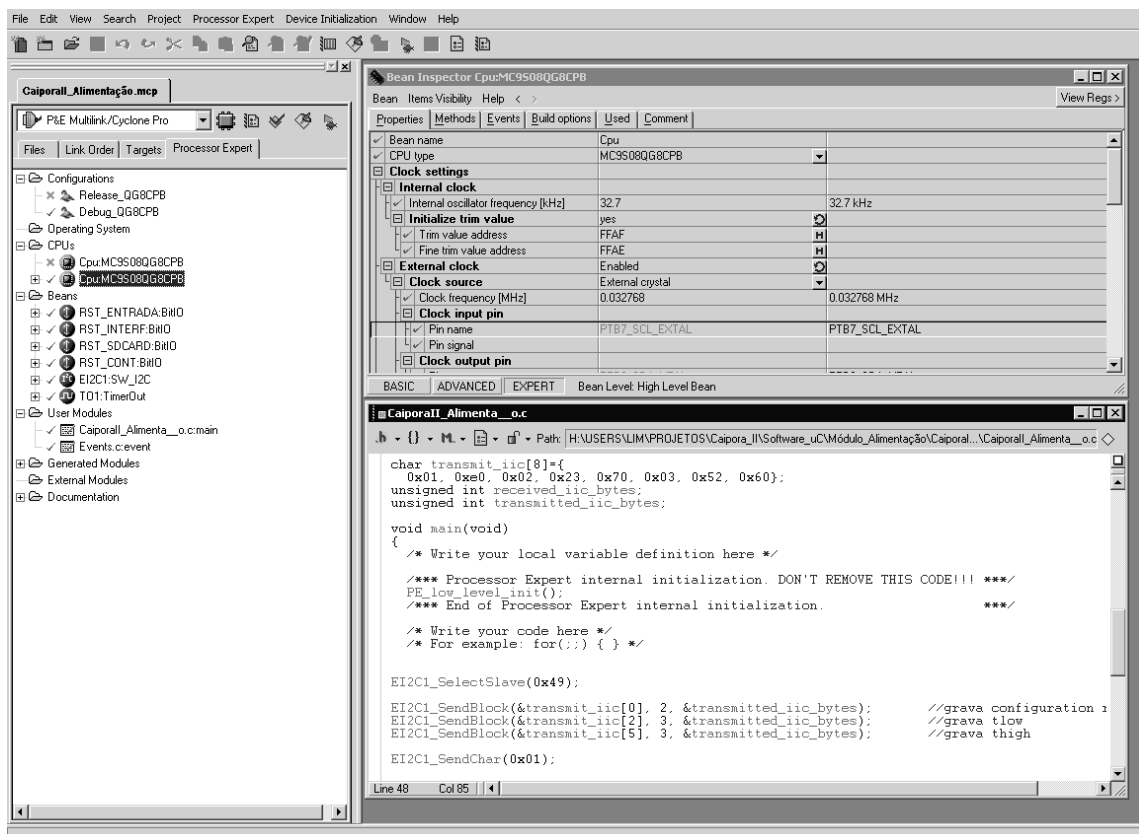


Figura 3-1 – Ambiente de desenvolvimento Freescale CodeWarrior.

Através da interface BDM transfere-se o código gerado no aplicativo Codewarrior para a memória *Flash* interna ao microcontrolador. Além da programação do microcontrolador, utiliza-se esta interface para depurar do código sendo executado no microcontrolador, o que permite diagnosticar erros de programação e tomar as medidas

corretivas necessárias. Embora existam programadores dedicados no mercado, utilizou-se a interface BDM embutida na placa de desenvolvimento DEMO9S08QG8, fabricada pela *P&E Microcomputer Systems*, para a programação do microcontrolador e depuração do programa desenvolvido.

A Figura 3-2 apresenta a tela de depuração do programa executado no microcontrolador. A depuração auxilia na identificação de problemas e na determinação de suas causas. Através da interface BDM é possível executar o código e visualizar na tela do computador a o conteúdo da memória do microcontrolador, das variáveis armazenadas na memória, o estado dos registradores internos do processador e o código fonte do programa.

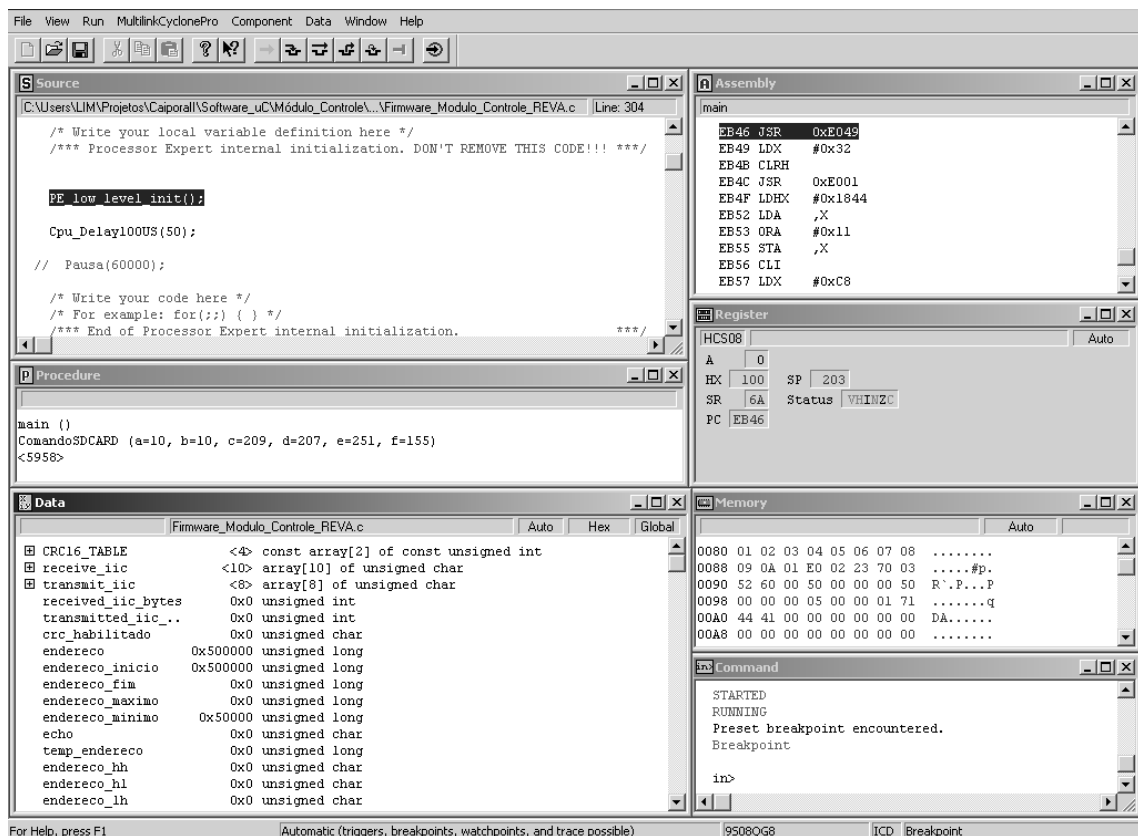


Figura 3-2 – Depuração do programa executado no microcontrolador.

Pode-se ainda alterar o conteúdo da memória RAM, e por conseguinte das variáveis nela armazenadas, interromper a execução do programa em determinado ponto (*breakpoint*) ou quando determinada posição de memória é acessada. Por exemplo, caso se suspeite que o conteúdo de determinada variável esteja sendo corrompido, pode-se depurar o programa interrompendo sua execução tão logo a posição de memória onde a variável em questão está armazenada seja escrita. Isto é bastante útil pois a mesma posição de memória poderia ter sido

lida diversas vezes, sem alterar seu conteúdo. A partir deste ponto, executa-se o programa passo a passo, observando se a alteração da variável era prevista ou não.

3.1.1 Módulo I²C para Codewarrior

O módulo I²C é responsável pela configuração e comunicação da interface I²C. Para utilizá-lo, basta adicionar o módulo a um projeto em que o *Processor Expert* esteja habilitado. A configuração do módulo especifica se o microcontrolador será mestre ou escravo no barramento I²C, qual par de pinos serão utilizados, a frequência do clock da linha SCL, o tamanho do buffer de recepção e transmissão, dentre outras configurações.

Caso seja configurado como dispositivo escravo, o endereço no barramento I²C deve ser especificado. Caso o dispositivo seja mestre no barramento, a frequência do barramento, gerada por ele é ajustada por um divisor de frequência. O *Processor Expert* calcula automaticamente o valor da frequência escolhida. Caso, por exemplo, a frequência de operação do microcontrolador seja alterada, será calculada e mostrada a nova frequência de operação. O microcontrolador permite a operação I²C até 100kHz, equivalente ao modo padrão descrito na seção 2.3.3.

A utilização do módulo I²C gera as funções necessárias à operação do módulo. Pode-se escolher quais funções estarão disponíveis no projeto por meio de um clique de mouse. Por padrão, são geradas as funções para habilitar e desabilitar o módulo I²C e enviar e receber um caractere ou bloco de caracteres. Para dispositivos escravo, são geradas também as funções que limpam do buffer de recepção e de transmissão, e que informam quantos caracteres estão nos *buffers* de transmissão e recepção. Além destas funções, são criados pontos de entrada para os eventos de recepção de um caractere e de transmissão de um caractere. Para estabelecer uma ação associada ao evento, basta inserir o código correspondente à função no arquivo `events.c`. Uma vez que o dispositivo escravo não pode iniciar uma transmissão, as funções de envio de um caractere e de envio de um bloco de caracteres aguardam a solicitação destes dados a partir do dispositivo mestre. Para dispositivos mestre é gerada uma função que recebe e armazena o endereço do dispositivo escravo que se deseja comunicar.

Por exemplo, um dispositivo mestre pode ler um bloco de dados de um dispositivo escravo através da seguinte sequência: O dispositivo mestre seleciona o dispositivo escravo no barramento (`SelectSlave`). Em seguida, envia um *byte* para o dispositivo escravo (`I2C_SendChar`). Este *byte* funciona como um comando predefinido, indicando que se deseja ler um bloco de dados. No dispositivo escravo, a recepção deste *byte* gera um evento,

após o qual enche-se o bloco de dados. Após um intervalo de tempo, o dispositivo mestre solicita que o dispositivo escravo envie o bloco (`I2C_SendBlock`). Esta solicitação gera um novo evento no dispositivo escravo, de requisição de leitura (`OnReadReq`). Este evento chama a função de envio do bloco (`I2C_ReadBlock`), e o bloco de dados é transferido do escravo para o mestre.

3.1.2 Módulo de Relógio de Tempo Real para Codewarrior

O módulo `TimeDate` implementa as funções de relógio de tempo real e calendário, utilizando um timer do microcontrolador como fonte de interrupções periódicas. A Tabela 3-1 lista as principais funções fornecidas pelo módulo `TimeDate`. O módulo armazena internamente data e hora nas estruturas `TIMEREC` (hora, minutos, segundos e centésimos de segundo) e `DATEREC` (ano, mês, dia) da linguagem C. Adicionalmente, o módulo permite que seja gerado um evento com hora marcada.

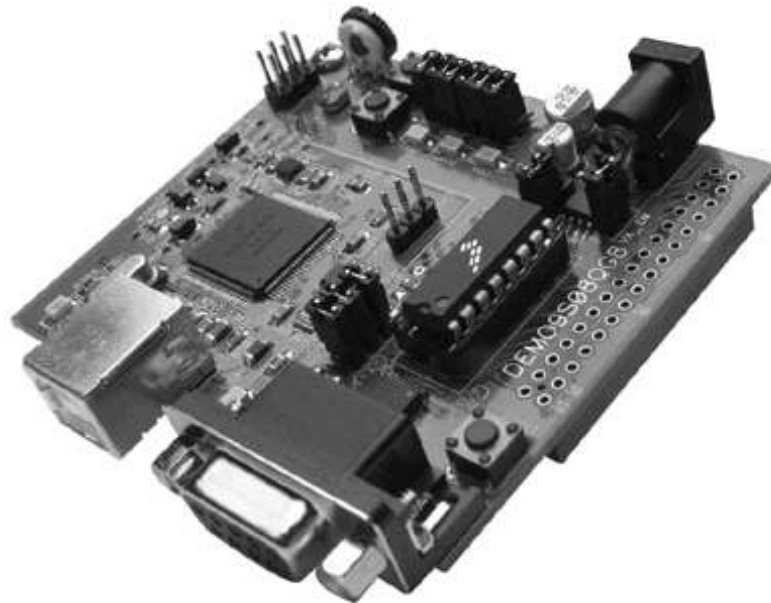
Tabela 3-1– Funções disponibilizadas pelo módulo `TimeDate` do Codewarrior

NOME DA FUNÇÃO	FUNÇÃO
<code>Enable</code>	Habilita a atualização do relógio / calendário.
<code>Disable</code>	Desabilita o relógio / calendário.
<code>SetTime</code>	Ajusta a hora do sistema.
<code>GetTime</code>	Obtém a hora do sistema.
<code>SetDate</code>	Ajusta a data do sistema.
<code>GetDate</code>	Obtém a data do sistema.
<code>GetDay</code>	Retorna o dia da semana atual.

3.1.3 Programação do Microcontrolador

Uma vez escrito o programa, este é compilado no Codewarrior através do comando `make`, disponível no menu `project`. Um arquivo executável é gerado, havendo a opção de simulá-lo no computador ou transferí-lo diretamente para a memória *flash* do microcontrolador. Neste caso, é necessário utilizar um programador. Pode-se utilizar a

ferramenta de desenvolvimento DEMO9S08QG8, que consiste de uma placa de circuito impresso voltada para a avaliação do microcontrolador MC9S08QG8. A placa, mostrada na Figura 3-3, conecta-se ao microcomputador através da porta USB. Apresenta soquete DIP para a inserção do microcontrolador, porta serial TIA/EIA-232 para comunicação do microcontrolador com dispositivos compatíveis, botões e leds indicativos ligados a pinos digitais de entrada e saída respectivamente, e interface BDM embutida, que permite utilizá-la para programar e depurar o microcontrolador na placa de desenvolvimento ou diretamente no circuito de destino, através de cabo de seis vias. Neste caso, o soquete do programador destinado ao microcontrolador deverá estar vazio.



Fonte: (FREESCALE)

Figura 3-3 – Foto da placa de desenvolvimento DEMO9S08QG8.

3.2 EAGLE LAYOUT EDITOR

O Eagle Layout Editor, fabricado pela empresa Cadsoft, é um programa de projeto de placas de circuito impresso, com função de desenho de diagramas esquemáticos e de *layout* de placas de circuito impresso. A versão gratuita limita a dois o número de camadas da placa de circuito impresso, e o tamanho da placa em 100mm x 80mm.

O projeto de uma placa de circuito impresso divide-se em duas etapas: desenho do diagrama esquemático do circuito, mostrada na Figura 3-4, e roteamento das trilhas da placa

de circuito impresso, mostrada na Figura 3-5. Adicionalmente, torna-se necessário desenvolver novas bibliotecas para componentes específicos, que não constem na biblioteca padrão do Eagle. Os comandos de verificação de erros elétricos (*Electrical Rule Check*) e de verificação de erros de projeto (*Design Rule Check*) são capazes de identificar trilhas em curto, com espessura diferente dos limites especificados no projeto, componentes posicionados em áreas proibidas, pinos não conectados, o que auxilia na identificação e correção de erros.

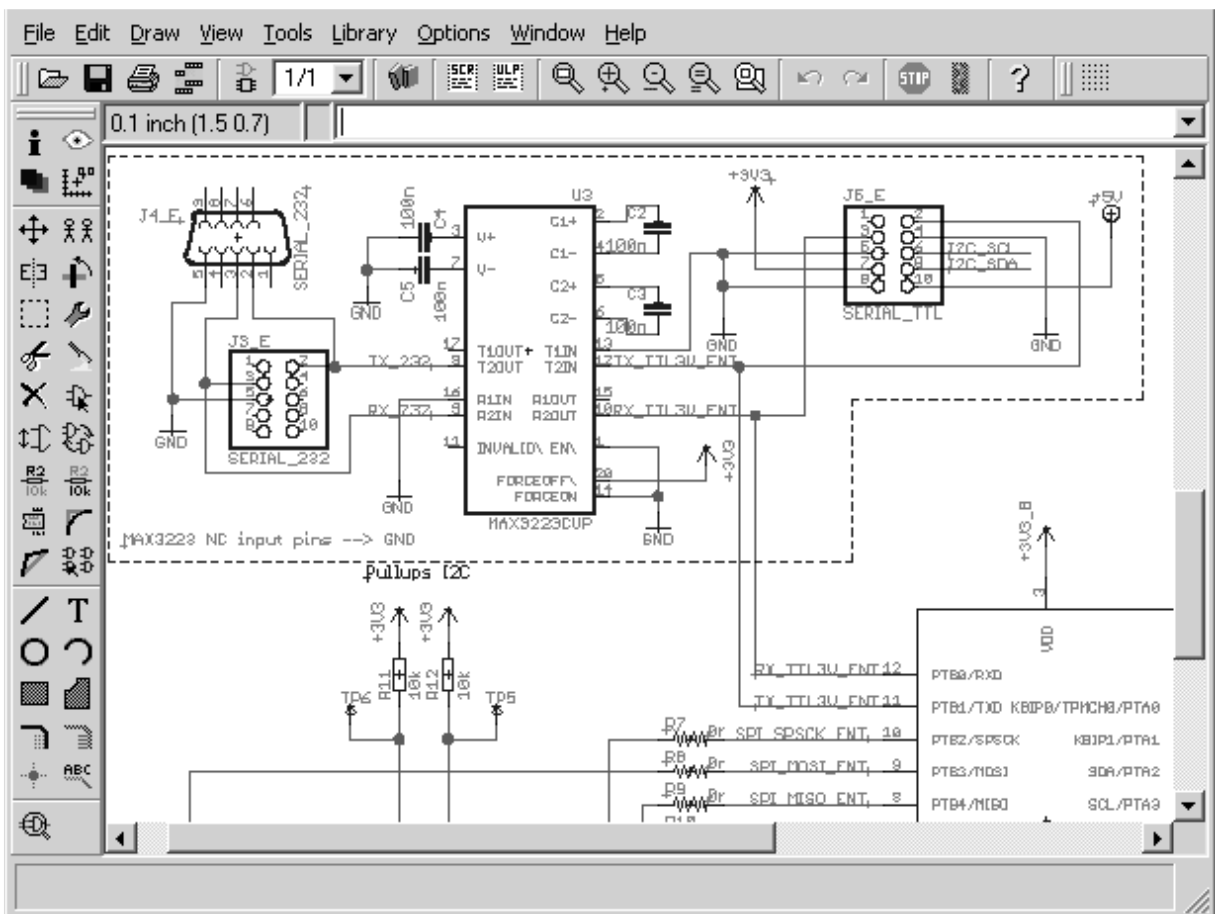


Figura 3-4 – Tela do programa Eagle para desenho do diagrama esquemático.

Uma vez terminado o *layout*, este é exportado para um formato de manufatura automatizada, através de *script* presente no Eagle. O script combina os layers necessários do *layout* para gerar cada camada da placa de destino. Usualmente utiliza-se o formato Gerber para as informações das trilhas (um arquivo para cada camada da placa) e um arquivo no formato Excellon para as informações de furação da placa.

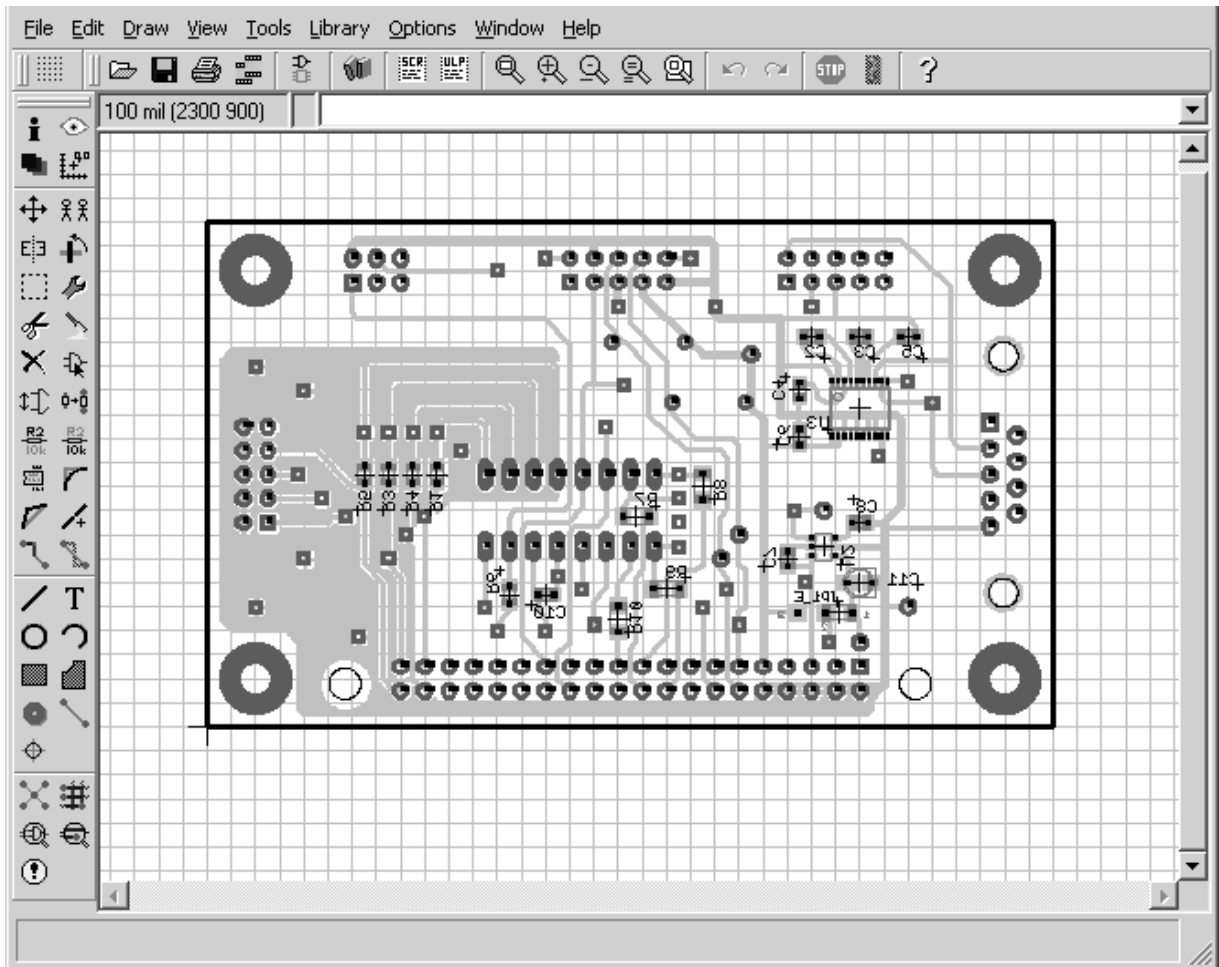


Figura 3-5 – Tela do programa Eagle para desenho da placa de circuito impresso.

3.3 FRESADORA CNC

Um método de confecção de placas de circuito impresso consiste na remoção de seções de cobre de uma placa virgem. Neste processo, uma broca girando a altas rotações (até 60.000 rpm, por exemplo) é deslocada mantendo-se contato com uma placa de cobre virgem. Uma vez que o caminho programado remove o cobre, trata-se de um processo subtrativo. A fresa desloca a broca ao longo de todo o contorno da trilha, isolando esta porção de cobre do restante da placa. São utilizadas brocas específicas para a remoção de cobre, furação e corte da placa de circuito impresso.

3.3.1 *Circuitcam*

Programa fornecido pelo fabricante de fresas LPKF que a partir dos arquivos Gerber e Excellon determina as ferramentas a serem utilizadas e o caminho a ser traçado por elas para a

produção da placa de circuito impresso. O arquivo de saída, de extensão LMD, é utilizado posteriormente pelo programa BoardMaster.

3.3.2 *Boardmaster*

Programa fornecido pelo fabricante LPKF que coordena a transmissão dos dados de produção da placa de circuito impresso diretamente à fresa. Converte o trajeto a ser traçado pela broca da fresadora especificado no arquivo LMD, gerado pelo CircuitCam, em comandos de movimentação da fresadora. Conforme necessário, solicita ao usuário que efetue a troca das ferramentas e gire a placa de cobre para a fresagem do lado oposto.

3.4 CYGWIN

O Cygwin disponibiliza aos sistemas Windows um ambiente do tipo Linux. Permite que programas desenvolvidos para Linux, através de seu código fonte e mediante algumas alterações sejam executados sob o sistema operacional Windows. Adicionalmente, a instalação padrão do Cygwin inclui as ferramentas e comandos do linux mais utilizadas. Desta forma, comandos normalmente utilizados sob a plataforma Linux podem ser utilizados no Windows através do ambiente Cygwin.

3.4.1 *Comando Hexdump*

O comando `hexdump` fornece uma visualização do conteúdo de um arquivo ou diretamente de um dispositivo de armazenamento. Mostra de um lado da tela os valores hexadecimais de um intervalo de memória, e do outro os respectivos valores ASCII. O comando abaixo inicia a visualização do conteúdo do dispositivo `/dev/sdb` a partir do endereço `0x00050000`.

```
hexdump -C -v -s0x00050000 /dev/sdb | less
```

O parâmetro `'-C'` especifica que a visualização será no modo ASCII + hexadecimal, ou seja, serão mostrados tanto o valor hexadecimal (à esquerda) quanto sua representação ASCII (à direita). O parâmetro `'-v'` especifica que todos os endereços serão mostrados (caso contrário, seções contínuas com o mesmo valor seriam ocultas e indicadas por um asterisco). `'-s0x00050000'` indica que a visualização começará a partir deste endereço. O parâmetro `'/dev/sdb'` indica o dispositivo a ser lido. Caso não se tenha certeza, pode-se remover o cartão e executar o comando novamente. Caso uma mensagem de erro seja recebida, o cartão era realmente o dispositivo `'/dev/sdb'`. Por fim, o parâmetro `'| less'` especifica que

haverá uma pausa entre cada página visualizada, aguardando intervenção do usuário. Outras opções estão listadas na documentação do comando, que pode ser acessada digitando-se ‘man hexdump’ na linha de comando do cygwin.

3.4.2 Comando dd

O comando dd efetua a cópia de ‘baixo nível’ de dados ‘brutos’. É uma ferramenta imprescindível quando se deseja copiar intervalos de dados a partir de endereços físicos específicos do dispositivo de memória, uma vez que o windows dificulta o acesso a determinadas porções dos dispositivos de armazenamento. Permite especificar diretamente o endereço inicial a ser gravado, o tamanho do bloco a ser copiado a cada vez, a quantidade de blocos, dentre outros. Pode ser utilizado também para apagar, gravando-se zeros, determinada porção do dispositivo de memória. Deve ser utilizado com cautela, uma vez que ao se gravar para o dispositivo errado, corre-se o risco de perder dados e/ou danificar o sistema. Outras opções estão listadas na documentação do comando, que pode ser acessada digitando-se ‘man dd’ na linha de comando do cygwin.

3.4.3 Comando tr

O comando tr (*translate*) permite que seja feita a substituição de determinado caractere (ou faixa de caracteres) por outro. Útil na conversão de um arquivo gerado com um código de página para outro ou na supressão de determinados caracteres de um arquivo. Por exemplo, a execução do comando abaixo resulta na cópia de ‘arquivo.entrada’ para ‘arquivo.saida’, porém com todas as ocorrências do caractere nulo removidas.

```
tr -d '\000' < arquivo.entrada > arquivo.saida
```

O parâmetro ‘\000’ indica qual caractere será procurado no arquivo, neste caso o caractere nulo. O parâmetro ‘-d’ especifica que estas ocorrências, quando encontradas, serão removidas. Outras opções estão listadas na documentação do comando, que pode ser acessada digitando-se ‘man tr’ na linha de comando do cygwin.

3.5 LABVIEW

O Labview, *Laboratory Virtual Instrument Engineering Workbench*, é um ambiente gráfico de programação onde utilizam-se ícones para criar programas. Os programas criados em Labview são chamados de instrumentos virtuais (*virtual instruments*). Os programas são

criados a partir de duas interfaces interligadas: o painel de controle e o diagrama em blocos. No painel de controle são especificadas as operações a serem executadas pelo programa, a partir de funções de bibliotecas do Labview ou de programas definidos pelo usuário. Uma ferramenta de 'fio' traça linhas na tela que conectam os dados entre os ícones. Os valores de variáveis presentes no diagrama em blocos podem ser mostrados no painel de controle (indicador), ou alterados (controle) pelo usuário através do painel de controle. Uma vez executado o programa, o usuário interage com o programa através do painel de controle.

Uma característica dos programas Labview consiste na possibilidade de representar um programa Labview por um ícone. Neste processo, conecta-se a terminais de um ícone as variáveis de entrada e/ou saída do programa que deseja-se ter acesso. Pode-se então incorporar o programa, representado pelo ícone, em outro programa.

4

DESENVOLVIMENTO DO PROJETO

Nesta seção descreve-se a implementação do projeto, detalhando-se o Sistema Caipora conforme o diagrama em blocos da Figura 4-1. O Sistema Caipora está organizado em módulos: Alimentação, Controle, Armazenamento de Dados, Entrada, Condicionamento de Sinais e Interface/Transmissão.

A partir da bateria, o Módulo Alimentação fornece energia ao sistema. Tem a função também de monitorar a tensão da bateria e as tensões geradas localmente no Módulo Alimentação. O Módulo Controle gerencia o processo de aquisição de dados, sendo responsável por iniciar a aquisição de dados, transferi-los para o cartão de memória e/ou transmiti-los para uma estação remota, através do Módulo Transmissão. O Módulo Armazenamento de Dados abriga o cartão de memória onde os dados medidos e de diagnóstico do sistema são salvos. O Módulo Entrada digitaliza os sinais analógicos e envia o resultado para o Módulo Controle. O Módulo Condicionamento de Sinais, ajusta a faixa de tensão ou corrente do sinal gerado pelo transdutor para a faixa de tensão do Módulo Entrada, além de fornecer alimentação destinada à excitação de sensores. O módulo Interface é responsável pelo envio dos dados para uma estação remota.

A placa de interconexão e as placas de circuito impresso dos módulos foram projetadas utilizando a versão gratuita do Eagle Layout Editor. Devido ao desenvolvimento do projeto em módulos, as restrições impostas pela versão gratuita não consistiram fatores limitantes do projeto. As placas foram confeccionadas utilizando equipamento de prototipagem rápida.

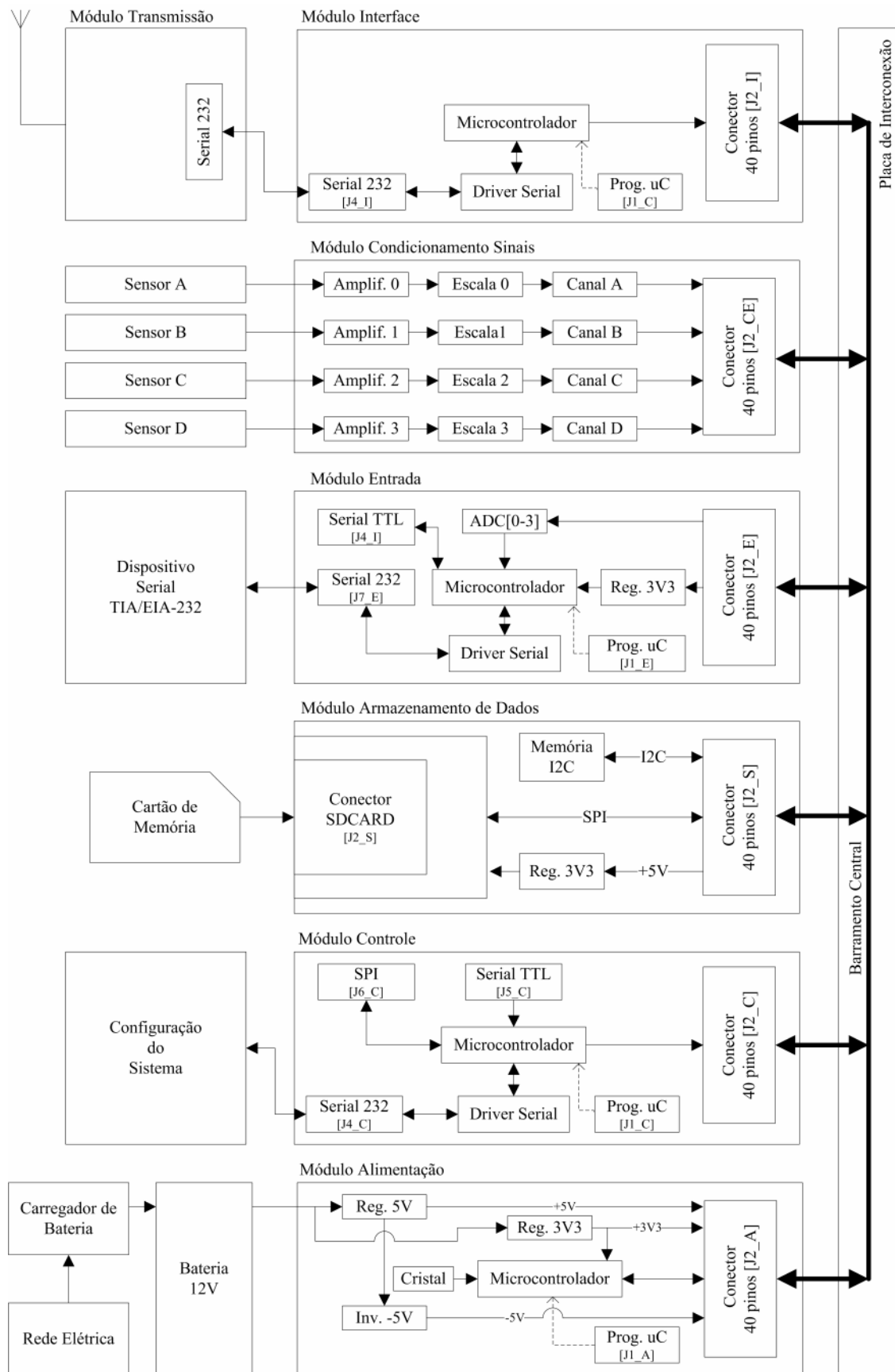


Figura 4-1 – Diagrama em blocos do Sistema Caipora.

Os módulos Alimentação, Controle e Entrada são microcontrolados. O microcontrolador MC9S08QG8, fabricado pela Freescale, foi escolhido devido às suas características de memória, suporte simultâneo às interfaces serial, SPI e I²C, flexibilidade das ferramentas de desenvolvimento, disponibilidade no Brasil e preço inferior às alternativas similares. O MC9S08QG8 apresenta apenas 16 pinos e está disponível em versão DIP e TSSOP, sendo este último de montagem em superfície. Embora tenha sido utilizada a versão DIP, há a possibilidade de reduzir as dimensões físicas das placas caso o encapsulamento TSSOP seja utilizado.

Desenvolveu-se *firmware* para cada microcontrolador, adequando-o às necessidades do respectivo módulo. Nesta etapa o compilador Codewarrior auxiliou na configuração por meio da seleção das interfaces de comunicação, origem da frequência de operação do microcontrolador, pinos de E/S, dentre outros. A Tabela 4-1 apresenta comparação da configuração dos pinos de cada microcontrolador.

Tabela 4-1– Comparação da função dos pinos do microcontrolador nos módulos Alimentação, Controle e Entrada.

PINO:	ALIMENTAÇÃO	CONTROLE	ENTRADA
1	#RST_ALIM (E)	#RST_CONT (E)	#RST_ENTRADA (E)
2	PTA4_BKGD (P)	PTA4_BKGD (P)	PTA4_BKGD (P)
3	+3V3		
4	GND		
5	EXTAL (X)	I2C_SCL (S)	I2C_SCL (E)
6	XTAL (X)	I2C_SDA (E/S)	I2C_SDA (E/S)
7	#RST_ENTRADA (S)	SPI_#SS1 (S)	Reservado ()
8	#RST_INTERF (S)	SPI_MISO (E)	Reservado ()
9	#RST_SDCARD (S)	SPI_MOSI (S)	Reservado ()
10	MON_5NEG (A)	SPI_SPCK (S)	Reservado ()
11	MON_3V3 (A)	TX_TTL3V_CONT (S)	TX_TTL3V_ENT (S)
12	#RST_CONT (S)	RX_TTL3V_CONT (E)	RX_TTL3V_ENT (E)
13	I2C_SCL (E)	SPI_#SS3 (S)	ADC3_ENT (A)
14	I2C_SDA (E/S)	SPI_#SS2 (S)	ADC2_ENT (A)
15	MON_VBAT (A)	SDCARD_DETECT (E)	ADC1_ENT (A)
16	MON_5V (A)	SDCARD_WP (E)	ADC0_ENT (A)

E=Entrada; S=Saída; E/S=Entrada/Saída; A=Sinal Analógico; X=Cristal; P=Programação

Excetuando-se os pinos de alimentação (3 e 4) e de programação (2), a função dos demais pinos é definida dentro do projeto Codewarrior. Os sinais EXTAL e XTAL destinam-se à conexão de um cristal oscilador; I2C_SDA e I2C_SCL referem-se à interface I²C; SPI_MISO, SPI_MOSI, SPI_SPCK e SPI_SS referem-se à interface SPI; TX_TTL3V e RX_TTL3V referem-se à interface TIA/EIA-232; ADCn_ENT são entradas analógicas; MON_VBAT, MON_5V, MON_5NEG e MON_3V3 destinam-se ao monitoramento das tensões de alimentação +5V, -5V, +3,3V e da bateria; #RST_x são sinais que controlam a reinicialização (*RESET*) dos diversos módulos.

Os módulos são conectados através do Barramento Central, que é a espinha dorsal do sistema. Sua implementação consiste de conector de 40 pinos, contendo as tensões de alimentação +5V, +3,3V, -5V e tensão da bateria; sinais dos barramento SPI, I²C e Serial; sinais de reinicialização dos módulos e os sinais referentes a quatro canais de ADC. Para referência, a listagem dos sinais do Barramento Central encontra-se na Tabela 4-2.

Uma placa de interconexão, de tamanho 10,9 x 8,9cm, foi projetada em duas camadas utilizando o programa Eagle. Os módulos desenvolvidos têm placa de tamanho 8,9 x 5,3 cm e se encaixam perpendicularmente à placa de interconexão. A Figura 4-14, na seção 4.7 mostra a disposição dos módulos montados na Placa de Interconexão. A placa de Interconexão tem a função de interconexão elétrica entre módulos através do Barramento Central e fornece rigidez mecânica para a fixação das placas de circuito impresso dentro do gabinete. Possui seis conectores fêmea de 40 pinos, destinados à fixação dos Módulos Alimentação, Controle, Armazenamento de Dados, Entrada, Condicionamento de Sinais e Interface. Uma vez que o Módulo Interface não foi implementado ainda, a transmissão de dados é feita através da porta Serial TIA/EIA-232 do Módulo Controle.

Tabela 4-2– Sinais do Barramento Central – conector de 40 pinos.

CLASSIFICAÇÃO	PINO(S)	SINAL	AL	C	AR	E	CS
ALIMENTAÇÃO	1	-12V	–	–	–	–	–
	3	-5V	S	–	–	–	E
	5	+5V	S	–	E	E	E
	7	+12V	E	–	–	–	–
	9	+3V3	S	E	–	–	–
I ² C	11	I2C_SDA	E/S	E/S	E/S	E/S	–
	12	I2C_SCL	E	S	E	E	–
RESET	13	#RST_CONT	S	E	–	–	–
	14	#RST_ENT	–	S	–	E	–
	16	#RST_SD	–	S	E	–	–
OUTROS	17	SDCARD_DETECT	–	E	S	–	–
	21	SDCARD_WP	–	E	S	–	–
SPI	18	SPI_MISO	–	E	S	–	–
	19	SPI_MOSI	–	S	E	–	–
	20	SPI_SPCK	–	S	E	–	–
	23	SPI_#SS3	–	S	E	–	–
SERIAIS	24	TX_TTL3V_CONT	–	S	–	–	–
	25	RX_TTL3V_CONT	–	E	–	–	–
	26	TX_TTL3V_ENT	–	–	–	S	–
	27	RX_TTL3V_ENT	–	–	–	E	–
ANALÓGICOS	37 a 40	ADC[0-3]	–	–	–	E	S
RESERVADOS	15, 22, 28, 29, 30, 31, 32, 33, 34, 35, 36	Sinais não implementados ainda.	–	–	–	–	–
TERRA	2, 4, 6, 8, 10.	GND					

Legenda: ‘Al’ = Alimentação; ‘C’ = Controle; ‘Ar’ = Armazenamento; ‘E’ = Entrada;
‘CS’ = Condicionamento de Sinais; ‘E’ = Entrada; ‘S’ = Saída; ‘I’ = Interface;
‘E/S’ = Bidirecional; ‘–’ = não utilizado pelo módulo;

4.1 MÓDULO ALIMENTAÇÃO

O Módulo Alimentação supre as tensões +5V, +3,3V e -5V, a partir da tensão da bateria, por meio de reguladores e inversores de frequência. Além disto, disponibiliza no Barramento Central a tensão da bateria (V_{bat}), caso aplicações futuras a necessitem. O microcontrolador presente neste módulo é responsável pelo sequenciamento durante a inicialização do sistema. Desta forma, módulos problemáticos podem ser identificados durante a inicialização. Abriga ainda o relógio-calendário do sistema e o sensor de temperatura ambiente TMP101.

O sistema é permanentemente alimentado por uma bateria. Na presença de rede elétrica, um carregador comercial mantém a bateria carregada, o que mantém o sistema operando por um intervalo de tempo mesmo que haja uma interrupção no fornecimento de energia. A alimentação +5V é obtida da tensão da bateria a partir de um regulador de tensão linear 78M05, capaz de fornecer até 500mA. Uma vez que para funcionar este regulador necessita que a tensão de entrada seja ao menos dois volts superior à tensão de saída, a tensão da bateria não deve apresentar tensão inferior a sete volts. A tensão de alimentação +3,3V é obtida por um regulador, 78M33, também a partir da tensão da bateria.

Utilizou-se o circuito integrado MAX660 para fornecer -5V a partir de +5V (circuito inversor). São poucos os circuitos alimentados por -5V uma vez que a saída MAX660 não é regulada e fornece corrente máxima de 100mA. Adicionalmente, a resistência de saída não nula do MAX660, estimada pela folha de especificação em 6.5Ω , introduz uma queda de tensão proporcional à corrente fornecida. A tensão de saída cai para cerca de -4.35V quando o MAX660 fornece corrente de 100mA. Durante a soldagem da placa escolhe-se a frequência de chaveamento, gerada pelo próprio MAX660, entre 10kHz ou 80kHz, através de um resistor de 0Ω , que funciona como um *jumper*. A operação a 10kHz oferece menor corrente quiescente. Optando-se pela frequência de 80kHz, obtêm-se menor flutuação na saída e a possibilidade de fornecer mais corrente [34]. A tensão -5V é necessária aos amplificadores empregados no Módulo Condicionamento de Sinais. A Figura 4-2 ilustra as derivações para a obtenção das tensões +3V3, +5V e -5V. Há ainda a possibilidade de soldar um regulador 7905 ao módulo, caso esteja disponível alimentação negativa inferior a -7V. Neste caso, deve-se optar entre o uso do MAX660 e o do regulador 7905 para a obtenção da tensão -5V.

O módulo foi projetado utilizando placa de circuito impresso face simples. Para melhorar a dissipação térmica dos reguladores de tensão, foi desenhada uma área estendida de cobre, conectada à aba de metal dos reguladores 78M05, 78M33 e 7905. A foto do Módulo Alimentação é mostrada na Figura 4-3.

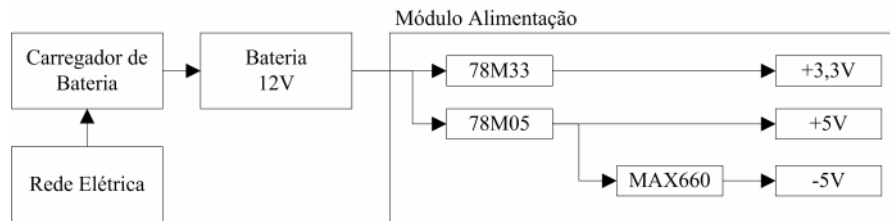


Figura 4-2 – Diagrama de derivação das tensões de alimentação.

O microcontrolador MC9S08QG8 deste módulo foi configurado para utilizar o cristal externo de 32,768Hz, com o circuito multiplicador por 512 habilitado. Uma vez que este módulo não necessita de muitos recursos de processamento, a frequência do circuito multiplicador, 16,6MHz, foi dividida por oito para gerar a frequência de *clock* do microcontrolador. Desta forma, a frequência do barramento interno, $f_{clk}/2$, é de cerca de 1MHz. Foram utilizados os módulos I²C, ADC, relógio-calendário e timer do *Processor Expert*.

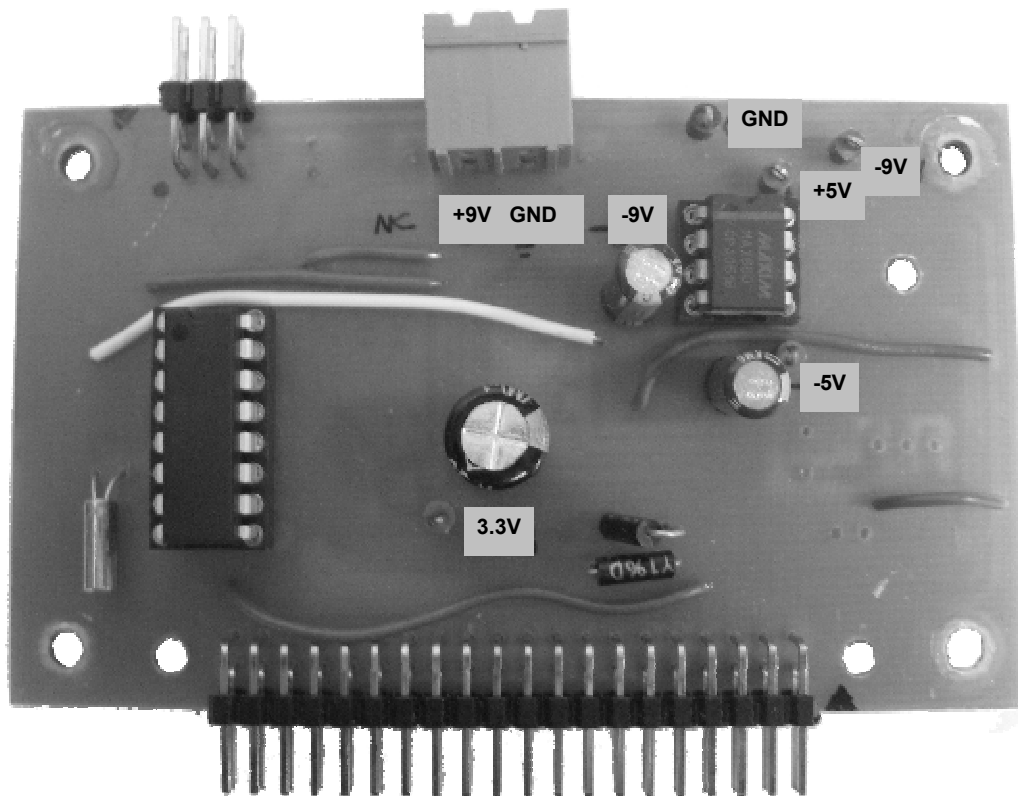


Figura 4-3 – Foto do Módulo Alimentação.

O programa desenvolvido no compilador CodeWarrior inicializa o Sistema Caipora conforme mostrado no fluxograma da Figura 4-4. Durante a inicialização do módulo Alimentação, as configurações do microcontrolador, definidas pelo usuário com o Processor Expert, são executadas. Em seguida, os sinais de *reset* dos módulos Controle, Entrada e Armazenamento de Dados são liberados sequencialmente, permitindo a execução das rotinas de inicialização destes módulos.

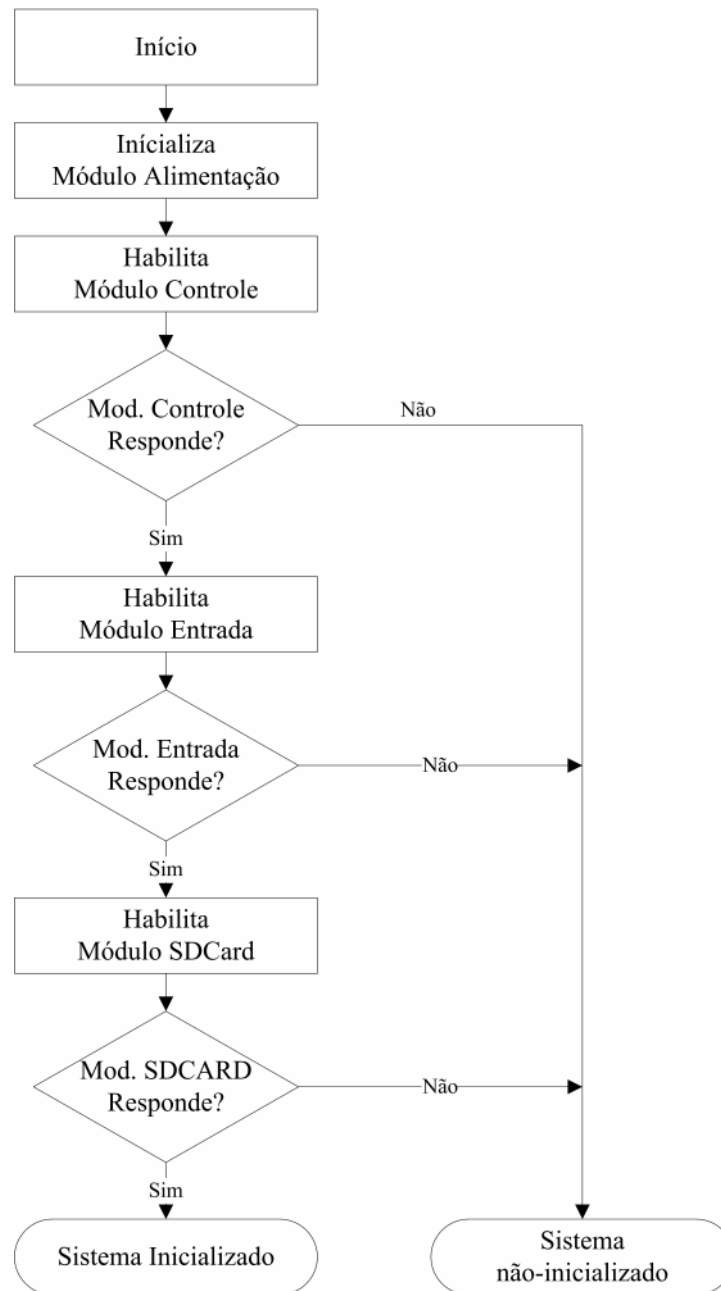


Figura 4-4 – Fluxograma da sequência de inicialização do Sistema Caipora.

A interface de comunicação I²C foi configurada no modo escravo, no endereço 0x39h. O relógio/calendário do módulo foi configurado para resolução de 100ms. O ADC interno foi configurado para resolução de 10 bits, com dois pinos destinados à leitura da tensão da bateria e da alimentação +5V, por meio de divisores de tensão. A referência de tensão bandgap interna foi habilitada. Os pinos de controle do *reset* dos módulos foram configurados como saídas digitais. O pino de reset do microcontrolador foi desabilitado e configurado como entrada digital, vinculada a uma interrupção.

4.1.1 Monitoramento da Tensão da Bateria e da Alimentação +5V

Divisores resistivos de tensão, Figura 4-5, são utilizados para monitorar a tensão da bateria e da alimentação +5V, baixando-as a níveis compatíveis com a entrada do ADC do microcontrolador do Módulo Alimentação. É importante monitorar a tensão da bateria do sistema, uma vez que nas baterias utilizadas, do tipo chumbo-ácido, a tensão em seus terminais decresce conforme a bateria é descarregada. Desta forma, pode-se estimar o nível de carga da bateria medindo-se esta tensão. O valor dos resistores R1 e R2 deve ser alto para minimizar a corrente drenada pelo divisor de tensão.

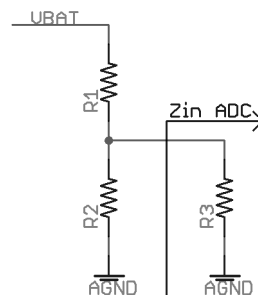


Figura 4-5 – Circuito para o monitoramento da tensão da bateria.

Entretanto, a impedância de entrada do ADC da família HCS08QG8 (modelada como R₃ na Figura 4-5) é relativamente baixa, de 5 a 7kΩ, de acordo com a folha de especificações do microcontrolador [11]. Isto introduz erros na medição da tensão da bateria caso R₂ seja relativamente alto. Ainda de acordo com a folha de especificações do microcontrolador, recomenda-se que a resistência série da fonte conectada ao ADC seja de no máximo 10kΩ.

4.1.2 *Monitoramento da Temperatura Ambiente do Sistema*

Incorporado ao Módulo Alimentação, localizado próximo ao microcontrolador e ao regulador de tensão +5V, há um sensor de temperatura TMP101, destinado a permitir que medidas de proteção do Sistema Caipora sejam executadas caso a temperatura ultrapasse o valor configurado. A localização do sensor foi escolhida devido à dissipação de energia por parte dos reguladores de tensão, sendo portando a região onde espera-se encontrar temperaturas maiores. O Módulo Controle, mestre no barramento I²C, configura o sensor durante a inicialização do sistema e periodicamente o lê. O sensor utiliza o endereço 0x49h do barramento I²C. O sinal de alerta do TMP101 está conectado a um pino de interrupção do microcontrolador do Módulo Alimentação. Durante a inicialização do sistema, os registradores T_{HIGH} e T_{LOW} do TMP101 são configurados respectivamente com a temperatura de disparo do alerta e com a temperatura de desabilitação do alerta. Caso a temperatura ultrapasse T_{HIGH}, o sinal de alerta torna-se ativo, informando a ocorrência deste evento por meio de interrupção ao Módulo Alimentação. O alerta somente é desabilitado quando a temperatura lida pelo sensor for inferior ao valor armazenado em T_{LOW}. A conexão do sinal de alerta a um pino de interrupção, ao invés de reinicializar imediatamente o microcontrolador, caso a função de *reset* do pino fosse habilitada. Isto permite que o módulo verifique se a temperatura realmente está alta antes de executar a ação correspondente.

Além do TMP101, há interno a cada microcontrolador um diodo destinado à medição da temperatura. Uma vez que a sensibilidade do sensor interno ao microcontrolador é de cerca de 1,8mV/°C, e a resolução do ADC é de $3.3/1024 = 3.22\text{mV}$, cada incremento no resultado do ADC corresponde a cerca de 1,8°C. Este poderia eventualmente substituir o TMP101. Porém, resta determinar se a menor resolução e a necessidade de o microcontrolador ser inicializado antes de se efetuar a primeira medição do sensor não impedem que seja utilizado em substituição ao TMP101. Além disto, a faixa de operação do TMP101 se estende de -40 a 125°C, enquanto a faixa de operação do sensor interno equivale à do microcontrolador, normalmente de -40 a 85°C.

4.1.3 *Hora do Sistema*

O microcontrolador do Módulo Alimentação utiliza um cristal de 32,768kHz como origem do *clock*, utilizado para implementar um relógio de tempo real via *software*. Utilizam-se as funções de relógio/calendário do módulo TimeDate fornecido pelo Codewarrior. Contabiliza-se desta forma ano, mês, dia, hora, minuto, segundo e centésimos de segundo,

válidos no intervalo de janeiro de 1998 a dezembro de 2099. Entretanto, no momento data e hora são ajustadas através do programador, sendo necessário implementar rotina para ajustar data e hora por meio mais prático, por exemplo através da interface serial.

4.2 MÓDULO CONTROLE

O módulo Controle tem a função de gerenciar a operação dos demais módulos, além de evitar que os dados armazenados no cartão de memória sejam acessados indevidamente. Embora os sinais do barramento SPI estejam disponíveis no barramento central, apenas o Módulo Controle e o Módulo Armazenamento estão eletricamente conectados à ele. As funções de acesso ao Módulo Armazenamento estão implementadas apenas no Módulo Controle. Assim, mesmo que fosse feita uma conexão indevida a outro módulo, por exemplo através da interface serial TIA/EIA-232 do Módulo Entrada, o conteúdo do cartão não seria acessado por não estarem disponíveis as funções de acesso e não haver conexão elétrica entre ambos. A placa de circuito impresso do Módulo Controle, de duas camadas, pode ser vista na Figura 4-6.

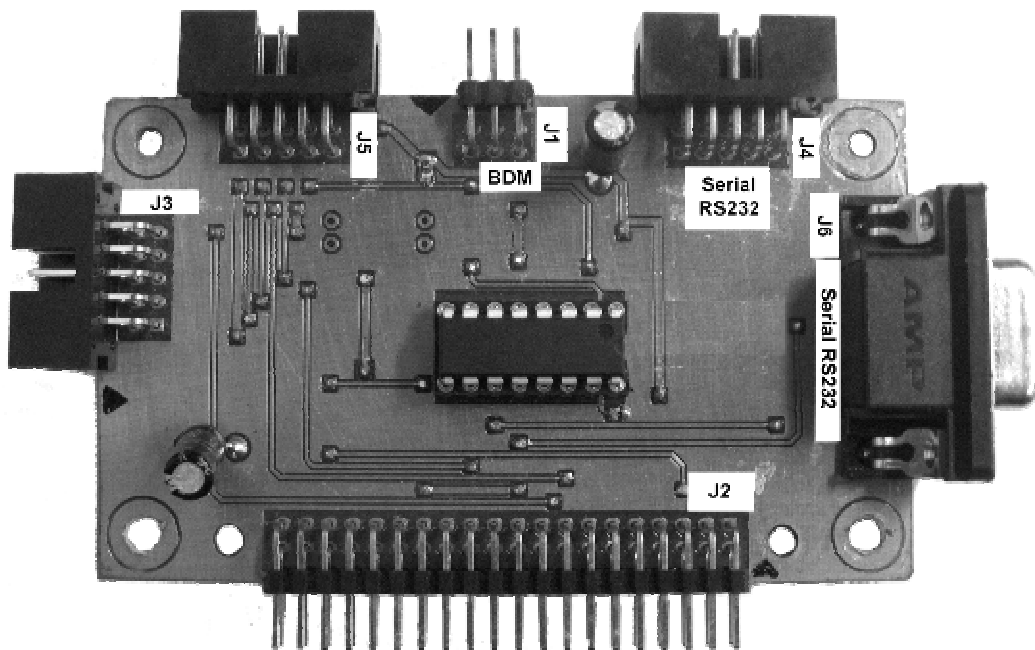


Figura 4-6 – Foto do Módulo Controle.

O código escrito no compilador Codewarrior para o Módulo Controle utiliza os módulos de *software* do Codewarrior I²C, SPI, Serial EIA/TIA-232, *timer*, dois pinos configurados como entrada para detecção de inserção do cartão de memória e de proteção contra escrita, além de pinos para seleção de dispositivos no barramento SPI.

A interface I²C foi configurada como mestre neste barramento, e portanto não teve um endereço atribuído. A frequência de *clock* I²C foi ajustada em cerca 88kHz, respeitando com certa margem a velocidade limite nominal de 100kHz.

A interface SPI foi configurada para ser inicializada com a frequência de *clock* de cerca de 230kHz. Esta frequência é utilizada somente durante o procedimento de inicialização do cartão de memória SDCard. Assim que o cartão é iniciado, a frequência é aumentada para cerca de 4,6MHz. O uso de frequências mais altas para acessar o cartão é encorajada pela redução no consumo de energia, uma vez que o cartão fica ativo por um período de tempo menor durante a escrita de um bloco.

O módulo timer foi configurado para gerar uma interrupção a cada 100ms. O intervalo de amostragem é configurado como um múltiplo deste intervalo. Assim, para configurar a aquisição de dados para uma amostra por minuto, basta configurar o intervalo de amostragem com o valor 600. Uma vez que a variável que armazena o valor do intervalo de amostragem é do tipo inteiro, sem sinal, de 16 bits, o intervalo máximo de amostragem é de cerca de 110 minutos.

A sequência de inicialização do Módulo Controle, mostrada na Figura 4-7, é iniciada quando o Módulo Alimentação libera o pino de *reset* do microcontrolador deste módulo. A etapa de configuração do microcontrolador consiste de rotina gerada pelo CodeWarrior que inicializa os periféricos de acordo com a seleção do usuário. Procede-se inicializando o sensor de temperatura presente no Módulo Alimentação, através da interface I²C, configurando a resolução utilizada e a temperatura de ativação da função Alerta.

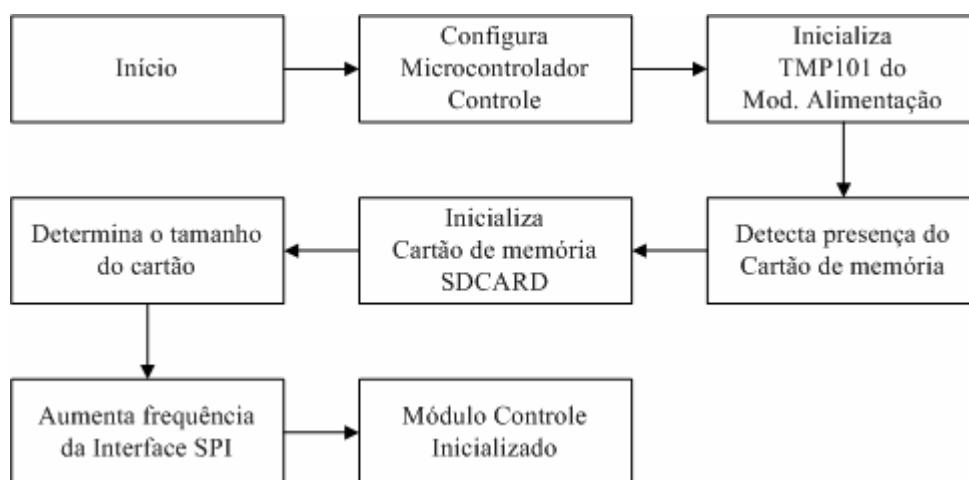


Figura 4-7 – Sequência de inicialização do Módulo Controle.

Em seguida, verifica-se, através da leitura dos pinos do microcontrolador conectados às chaves do conector do cartão, se o cartão de memória está inserido e se a chave de proteção contra a escrita do cartão está em posição que habilite a gravação. Neste caso, executa-se o procedimento de inicialização do cartão de memória no modo SPI. Em caso de sucesso, procede-se lendo o registrador CSD do cartão de memória e a partir deste calcula-se o endereço limite do cartão, ou seja, o tamanho do cartão. Ao fim deste procedimento, o Módulo Controle está inicializado.

4.2.1 Comandos de Configuração e Controle do Sistema Caipora

Assim que o Módulo Controle é inicializado, a interface serial é continuamente monitorada, aguardando o recebimento de um caractere. No programa desenvolvido para o microcontrolador deste módulo, são associadas determinadas funções a determinados caracteres, que ao serem recebidos resultam na execução de uma ação. A Tabela 4-3 apresenta alguns comandos válidos, recebidos pela interface serial do Módulo Controle e a correspondente ação do comando.

Caso o comando que inicia a aquisição de dados seja recebido, o Módulo Controle passa a executar também a máquina de estados do processo de aquisição de dados.

Tabela 4-3 – Lista parcial dos comandos implementados no Módulo Controle.

CARACTERE DE COMANDO	AÇÃO EXECUTADA
'+'	Incrementa o endereço e retorna através da interface serial o conteúdo (caracteres imprimíveis) do bloco (512 <i>bytes</i>).
'_'	Decrementa o endereço e retorna através da interface serial o conteúdo (caracteres imprimíveis) do bloco (512 <i>bytes</i>).
'C'	Habilita o envio de mensagens de resposta aos comandos através da interface serial (<i>Echo On</i>).
'c'	Desabilita o envio de mensagens de resposta aos comandos através da interface serial (<i>Echo Off</i>).
'D'	Habilita o envio de uma cópia dos dados gravados no cartão através da interface serial.
'd'	Desabilita o envio de uma cópia dos dados gravados no cartão através da interface serial.

Tabela 4-4 – Lista parcial dos comandos implementados no Módulo Controle [continuação].

CARACTERE DE COMANDO	AÇÃO EXECUTADA
‘E’	Retorna o endereço atual, em hexadecimal, através da interface serial, no formato E=xxxxxxxxh.
‘e’	Ajusta o endereço atual. Após a digitação do comando, deve-se enviar oito caracteres em hexadecimal, contendo o novo endereço de 32 bits.
‘g’	Lê o registrador CSD do cartão de memória e calcula o tamanho do cartão.
‘H’	Inicia a aquisição de dados, armazenando-os no cartão de memória, com o intervalo de amostragem configurado.
‘h’	Interrompe a aquisição de dados.
‘I’	Verifica o status do cartão de memória: se está inserido e se está protegido.
‘j’	Inicializa o cartão de memória SD no modo SPI.
‘5’	ajusta intervalo de amostragem (intervalo de amostragem = valor ajustado * 100ms).
‘.’ (ponto)	Lê o sensor TMP101 e retorna o valor da temperatura.

4.2.2 Processo de Aquisição de Dados

O processo de aquisição de dados ocorre segundo uma máquina de estados, implementada por *software*, executada no microcontrolador do Módulo Controle. O diagrama de transição de estados é mostrado na Figura 4-8. O estado atual do processo é dado pela valor da variável `status_modulo_controle`, sendo o próximo estado uma função do estado atual e dos respectivos valores de outras variáveis de entrada.

Ao receber o comando ‘H’ (ver Tabela 4-3) através da interface serial, o Módulo Controle envia através da interface I²C um caractere solicitando ao Módulo Entrada o envio de dados. O Módulo Entrada, ao receber este comando, lê os quatro canais analógicos, além da referência de tensão bandgap e do sensor de temperatura internos, e gera uma *string* contendo o resultado da medição. O Módulo Controle aguarda por um período de tempo, até que o Módulo Entrada inicie a transmissão, também pela interface I²C, da string gerada. Ao

receber os dados do Módulo Entrada, o Módulo Controle concatena-os com outros dados. Se a opção de enviar o resultado através da serial estiver habilitada, a string é enviada através da serial. Em seguida o Módulo Controle checa o endereço atual, verificando se encontra-se na faixa de memória determinada durante a inicialização do cartão de memória. Caso o endereço seja inválido, o Módulo Controle é travado, com vistas a proteger o conteúdo do cartão de memória. Se o endereço for válido, inicia-se o procedimento de escrita de um bloco no cartão de memória com o conteúdo do bloco de dados SDCard.

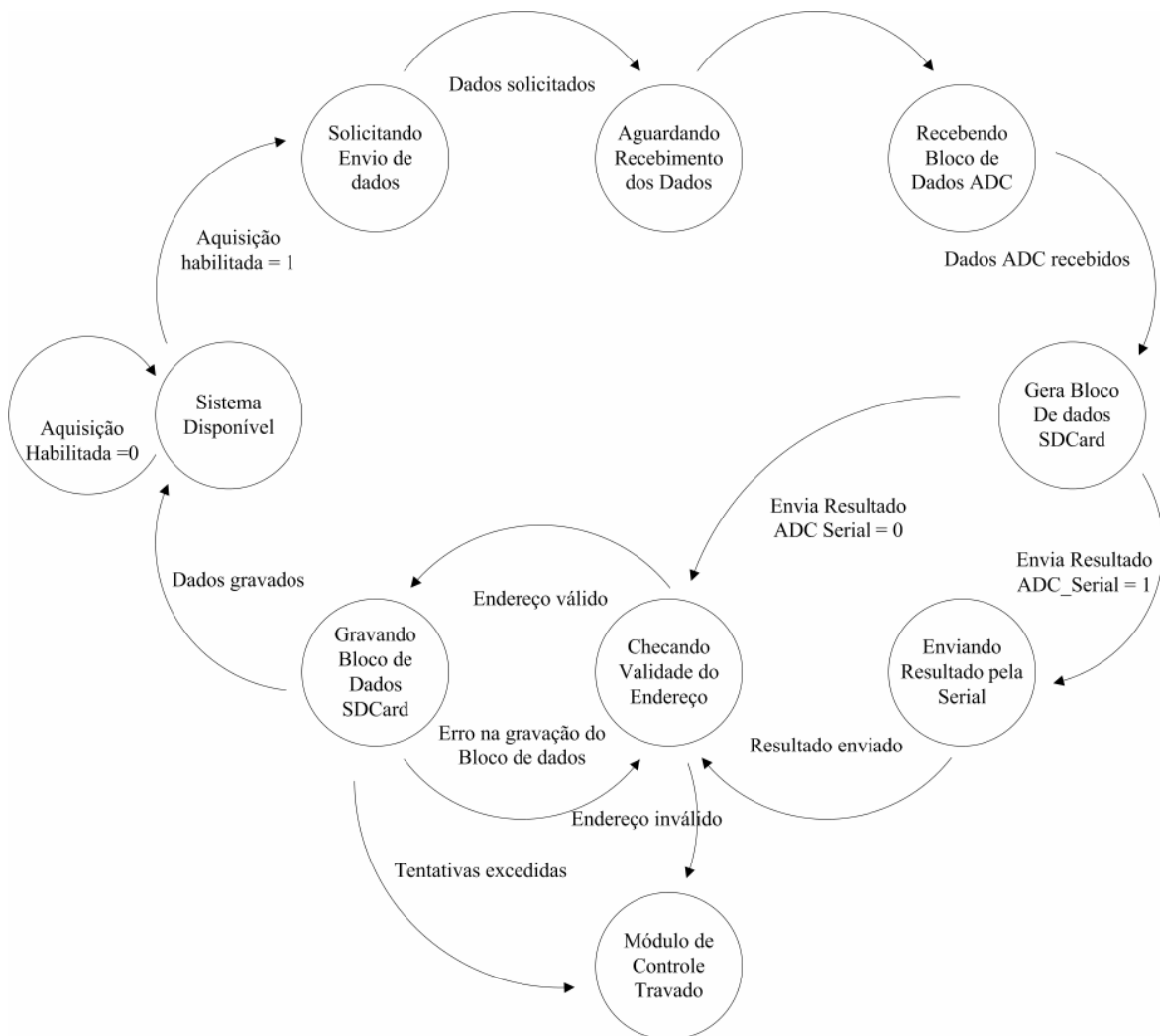


Figura 4-8 – Diagrama de transição de estados do processo de aquisição de dados

Caso haja um erro na gravação, o Módulo Controle verifica novamente o endereço e tenta gravar o bloco de dados novamente. Se as tentativas sem sucesso ultrapassarem um valor pré-configurado, o sistema é travado. Caso contrário, o bloco é gravado e o sistema está disponível para uma nova aquisição.

4.3 MÓDULO ARMAZENAMENTO DE DADOS

O Módulo Armazenamento de Dados fornece meios de acesso ao cartão de memória SD Card. O acesso é feito no modo SPI, uma vez que o microcontrolador suporta esta interface e a aplicação não exige altas taxas de transferência de dados. Embora o barramento caipora apresente 40 pinos, apenas os sinais do barramento SPI, da alimentação +5V e três pinos de E/S são utilizados pelo Módulo Armazenamento. Cartões padrão de até 2GB são suportados pelo sistema. Cartões do tipo SDHC e de capacidade superior não são suportados. A Figura 4-9 mostra a foto do Módulo Armazenamento de Dados.

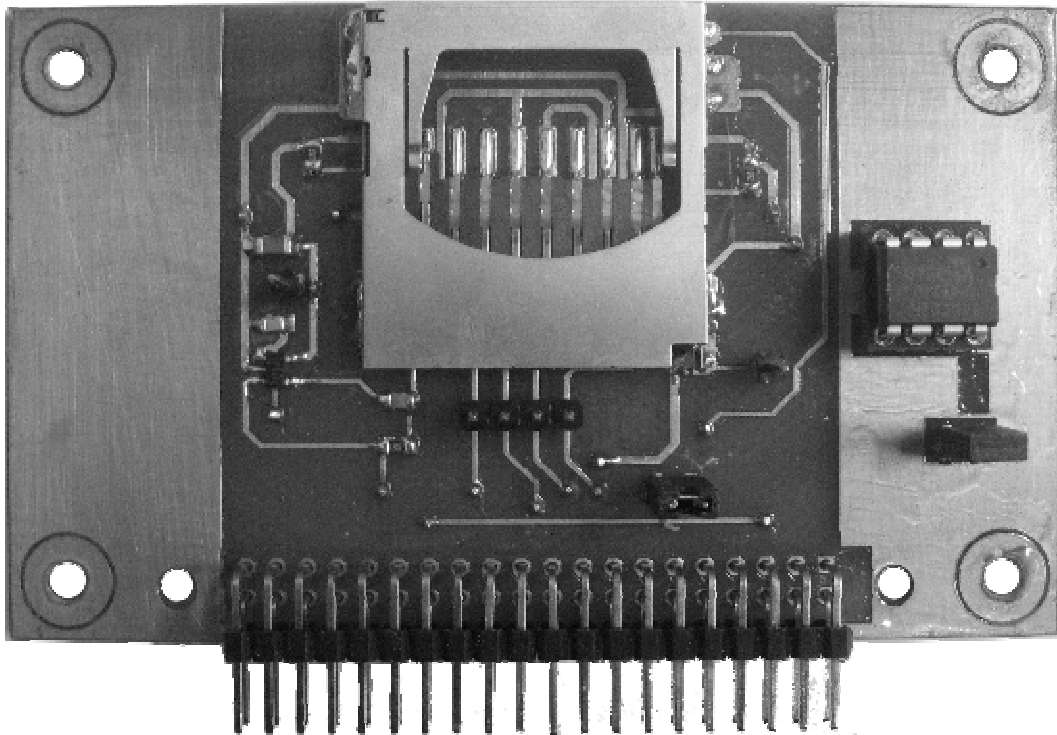


Figura 4-9 – Foto do Módulo Armazenamento de Dados.

Um regulador de tensão de 3,3V, incorporado ao módulo, alimenta o cartão de memória. O pino de habilitação do regulador está conectado a um pino de saída do Módulo Alimentação, que controla o fornecimento de energia ao cartão. Isto permite que a inserção/remoção do cartão seja simulada, procedimento que pode ser necessário para restabelecer a comunicação com o cartão nos casos eventuais em que uma condição de erro resulte na interrupção da comunicação com o cartão.

Uma vez que o módulo conecta-se ao sistema através do barramento Central, aproveitou-se os sinais I²C disponíveis no barramento para incorporar no Módulo Armazenamento um soquete destinado a uma memória do tipo 24LC. Embora esta memória

não apresente grande capacidade de armazenamento, permite que informações acerca do estado de operação do sistema sejam armazenadas. Por exemplo, o número de série do cartão de memória em uso e o valor do último endereço utilizado poderiam ser armazenados nesta memória para agilizar um algoritmo que determine qual endereço está livre, e qual o espaço disponível no cartão de memória.

4.3.1 Procedimento de Inicialização do Cartão de Memória SDCard

Durante a inicialização, o Módulo Controle verifica se o cartão está inserido e desprotegido. Isto é feito através de duas chaves mecânicas presentes no conector do cartão de memória e conectadas a entradas digitais do microcontrolador. Adicionalmente, caso o cartão seja removido, é gerada uma interrupção, a qual sinaliza ao Módulo Controle a necessidade de atualizar o status do cartão de memória. Assim que possível, o Módulo Controle verifica se o cartão realmente está inserido e disponível para gravação.

Procede-se chamando a rotina que determina o máximo bloco acessível do cartão de memória, i.e., o tamanho do cartão. Este limite é armazenado em uma variável na memória RAM do microcontrolador, de forma a impedir tentativas de leitura/escrita a blocos inválidos.

4.3.2 Estrutura do Bloco de Dados

Os dados são armazenados no cartão de memória em formato ASCII, com o caractere ‘espaço’ como delimitador dos campos. Cada registro é iniciado por um caractere ‘G’, seguido dos campos de dados. Cada registro ocupa um bloco inteiro de 512 *bytes* do cartão de memória. São armazenadas as informações de data, hora, número do registro, resultado da conversão do ADC para os canais A a D, leitura da referência de tensão e temperatura internas do microcontrolador dos módulos entrada e alimentação e temperatura ambiente medida pelo TMP101. Um caractere ‘ponto e vírgula’ seguido do caractere CR (‘0x13’) indicam o fim do registro. O restante do bloco é preenchido com caracteres nulos (0x00).

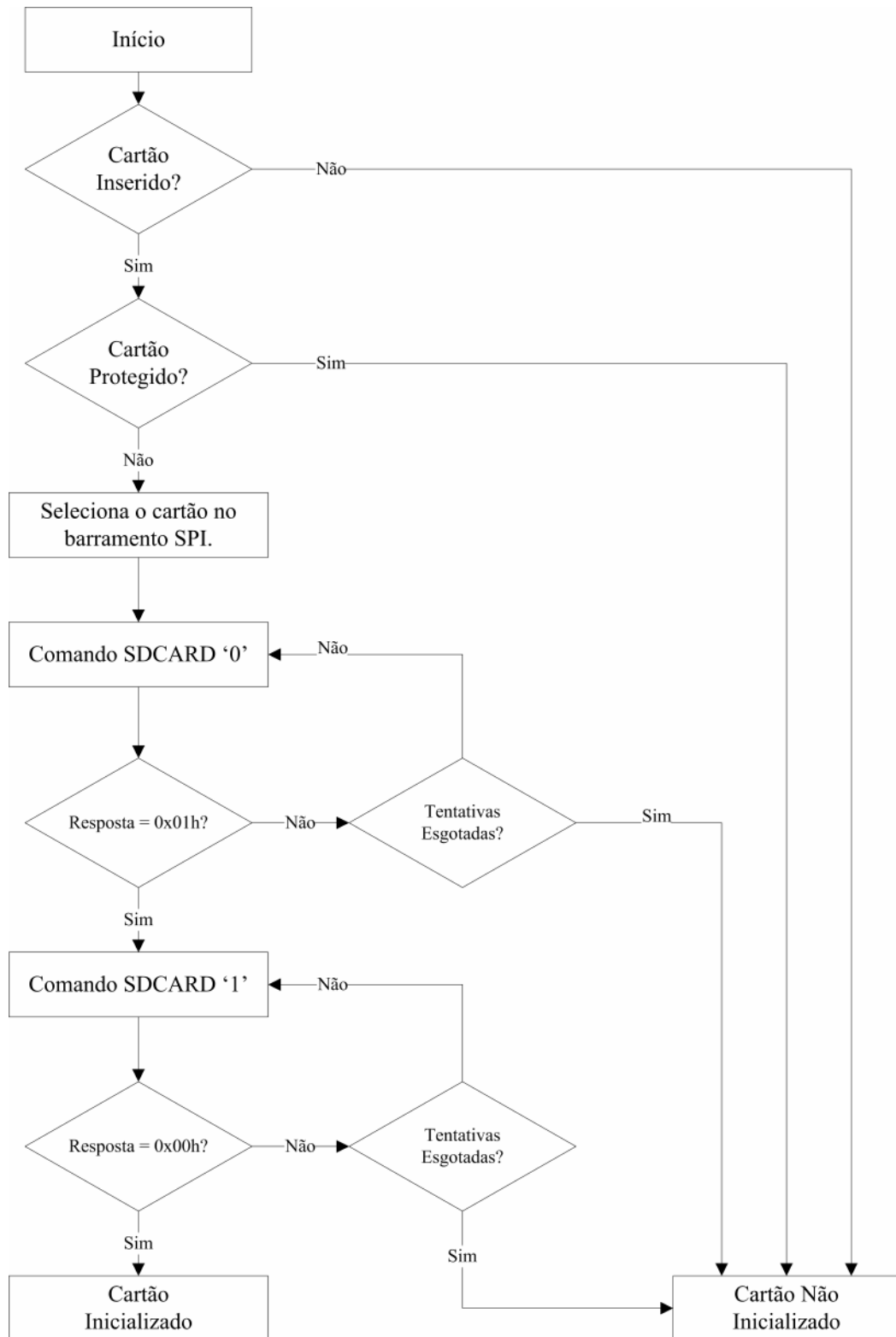


Figura 4-10 – Diagrama de fluxo da inicialização do cartão de memória.

4.3.3 Leitura do SDCard em um Microcomputador

Após a aquisição de dados, o conteúdo do cartão de memória pode ser lido diretamente em um microcomputador, mediante adaptador para leitura de cartões de memória.

Uma vez que os dados são armazenados sem utilizar um sistema de arquivos, não é possível copiar os dados utilizando o gerenciador de arquivos. Torna-se necessário ter acesso diretamente ao cartão de memória, o que é possível ao se utilizar o ambiente cygwin. O procedimento consiste em utilizar o comando `hexdump` para verificar o ponto de montagem do dispositivo removível (por exemplo, `/dev/sda`). Em seguida utiliza-se o comando `dd` para copiar o conteúdo de um intervalo de endereços do cartão de memória para um arquivo no computador. Entretanto, o arquivo criado será uma cópia binária do intervalo de memória especificado, incluindo as seções não utilizadas de cada bloco do cartão. Desta forma, executa-se o comando `tr` no arquivo copiado do cartão de memória para remover os caracteres nulos do arquivo de destino.

4.4 MÓDULO CONDICIONAMENTO DE SINAIS

O Módulo Condicionamento de Sinais tem função de ajustar a faixa de tensão/corrente de saída do sensor à faixa de tensão de entrada do ADC do microcontrolador. O módulo foi projetado para converter as faixas 0 a 3,3V, 0 a 5V, 0 a 10V, -1,65V a +1,65V, -2,5V a +2,5V, -5V a +5V, -10V a +10V e 0 a 20mA para a faixa 0 a 3,3V utilizada pelo conversor ADC do microcontrolador do Módulo Entrada.

A Figura 4-11 apresenta o esquema de um canal do Módulo Condicionamento de Sinais. O circuito de condicionamento consiste de dois estágios: o primeiro, inversor, dimensiona o sinal de entrada para a amplitude de 3,3V (-3,3V a 0V para sinais de entrada unipolares; -1,65V a +1,65V para sinais bipolares). O segundo estágio, também inversor, de ganho unitário, restaura a polaridade original do sinal. Se o canal estiver configurado para sinais bipolares (JP3 na posição 1-2) o segundo estágio soma 1,65V ao sinal, deslocando-o de modo que a saída esteja na faixa de 0 a 3,3V. A tensão 1,65V é gerada no próprio Módulo por um regulador ajustável, a partir da tensão 3,3V. As tensões de alimentação dos amplificadores operacionais, +5V e -5V, foram ocultas na Figura 4-11 para simplificar o esquema.

É necessária uma margem de cerca de 1,5V entre a tensão de alimentação e a máxima tensão de saída devido à incapacidade de o amplificador operacional utilizado excursionar a saída até as tensões de alimentação.

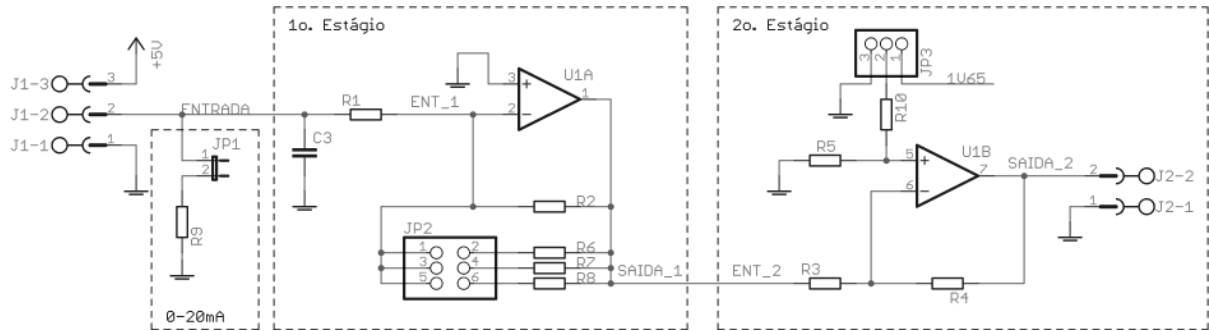


Figura 4-11 – Esquema de um canal do Módulo Condicionamento de Sinais.

A faixa é selecionada através dos *jumpers* JP2, que alteram o ganho G_1 do primeiro estágio, definido pela equação (4-1). A equação (4-2) apresenta o ganho do segundo estágio. Utiliza-se $R_3=R_4$ para que o ganho G_2 deste estágio seja unitário. A equação (4-3) apresenta a relação da tensão de saída do Módulo com a tensão de entrada do Módulo.

$$G_1 = \frac{V_{SAIDA_1}}{V_{ENTRADA}} \quad \therefore \quad G_1 = - \frac{R_1}{R_2 \parallel R_6 \parallel R_7 \parallel R_8} \quad (4-1)$$

De acordo com JP2

$$G_2 = \frac{V_{SAIDA_2}}{V_{ENT_2}} \quad \therefore \quad V_{SAIDA_2} = - \underbrace{\frac{R_4}{R_3}}_{G_2} \cdot \left(V_{ENT_2} + \underbrace{1,65V}_{\substack{\text{Somente se JP3} \\ \text{estiver instalado} \\ \text{na posição [1-2]}}} \right) \quad (4-2)$$

$$V_{SAIDA_2} = \left(\frac{R_1}{R_2 \parallel R_6 \parallel R_7 \parallel R_8} \right) \cdot V_{ENTRADA} + \underbrace{1,65V}_{\substack{\text{Somente se JP3} \\ \text{estiver instalado} \\ \text{na posição [1-2]}}} \quad (4-3)$$

A Tabela 4-5 sumariza os ganhos utilizados para cada faixa. A coluna ‘JP2’ indica a configuração do *jumper* JP2 para o respectivo ganho. Uma vez que o 1º e o 2º estágio são inversores, os ganhos apresentados são negativos.

Para utilizar a faixa 0 a 20mA, insere-se o ‘*Jumper*’ JP1, que conecta o resistor R9 (ver Figura 2-5) de 147Ω do terminal positivo da entrada para o terra. Deve-se preferencialmente selecionar a faixa de entrada de 0 a 3,3V. No modo 0 a 20mA o resistor de entrada R9 está efetivamente em paralelo com a impedância de entrada do circuito subsequente. Caso o resistor R9 fosse conectado diretamente à entrada do ADC do microcontrolador, cuja impedância de entrada é de cerca de $5k\Omega$, o paralelo destas duas impedâncias resultaria numa redução de cerca de 3% na impedância de entrada equivalente. Isto se traduziria em uma leitura também inferior, sensível à impedância de entrada do ADC, o que é uma característica indesejada. Entretanto, a conexão do resistor R9 à entrada do primeiro estágio do circuito de condicionamento de sinais, cuja impedância de entrada de cerca de $100k\Omega$ (valor de R1), torna pequeno o erro devido a este fator ($\approx 0.2\%$), uma vez que $100k\Omega \gg 150\Omega$.

Tabela 4-5 – Faixas de tensão de entrada do Módulo Condicionamento de Sinais.

FAIXA DE ENTRADA:	G_1	JP2	1o. ESTÁGIO	G_2	OFFSET	2o. ESTÁGIO
0 a 3,3V	-1,00	-	-3,3V	-1,00	0,00	0 a 3,3V
0 a 5V	-0,64	[1:2]	a			
0 a 10V	-0,32	[3:4]	0V			
-1,65V a +1,65V	-1,00	-	+1,65V		+1,65	
-2,5V a +2,5V	-0,64	[1:2]	a			
-5V a +5V	-0,32	[3:4]	-1,65V			
-10V a +10V	-0,15	[5:6]				

A Figura 4-12 mostra a foto do Módulo Condicionamento de Sinais. Na parte inferior vê-se o conector do Barramento Central (20 x 2 pinos), na parte superior os conectores de entrada para os sensores, referentes aos canais A-D.

O sinal condicionado de saída pode ser disponibilizado tanto nos conectores localizados nas bordas esquerda e direita, como diretamente no Barramento Central, desde que os respectivos *jumpers* estejam instalados.

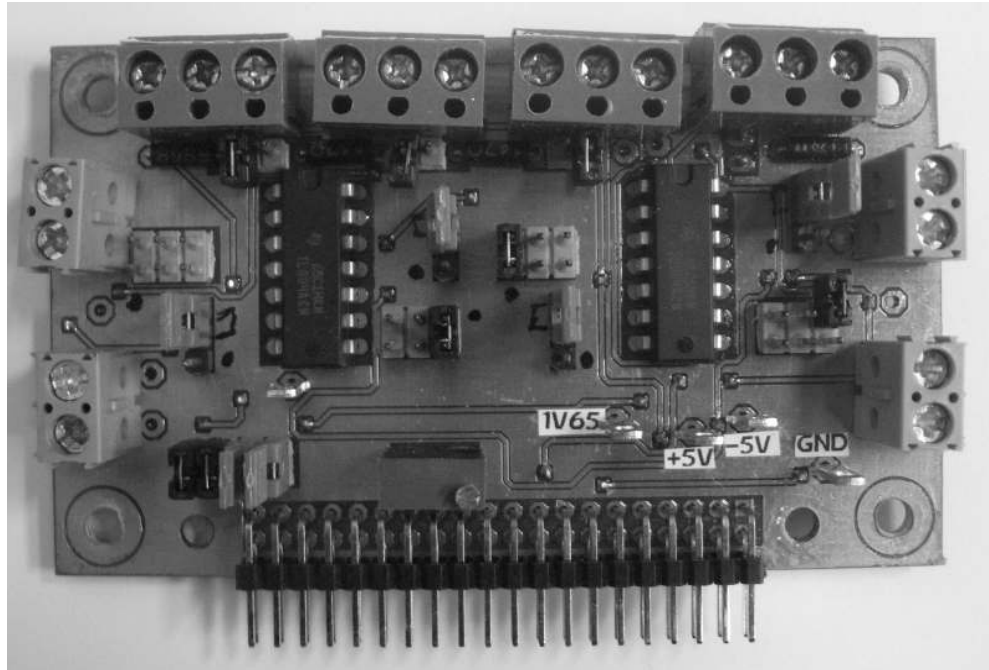


Figura 4-12 – Foto do Módulo Condicionamento de Sinais.

4.5 MÓDULO ENTRADA

O Módulo Entrada é responsável pela digitalização dos sinais analógicos de saída do Módulo Condicionamento de Sinais e por receber sinais através de interface serial EIA/TIA-232, destinada à conexão, por exemplo, de um GPS.

O Módulo Condicionamento de Sinais disponibiliza no Barramento Central suas quatro saídas, na faixa 0 a 3,3V. Cada canal é ligado a um pino de ADC do microcontrolador do Módulo Entrada através de um *jumper*. Uma vez que a referência de tensão utilizada pelo ADC é a própria alimentação do microcontrolador, utilizou-se um regulador de tensão exclusivo para este módulo, com o intuito de reduzir interferências e flutuações introduzidas pela operação dos demais módulos, especialmente módulos que apresentem circuitos digitais operando durante a digitalização dos sinais.

No projeto da placa de circuito impresso deste módulo, a referência de terra foi subdividida em terra analógico e terra digital. A seção analógica foi roteada, utilizando um plano

de terra no entorno dos quatro canais. A seção digital foi roteada separadamente, e as duas seções foram conectadas num único ponto [35]. Estas medidas têm por objetivo reduzir correntes na malha de terra, que resultam em flutuações na referência de terra, e portanto nas medições feitas com o ADC. A foto do módulo é mostrada na Figura 4-13.

O código escrito no compilador Codewarrior para o Módulo Entrada utiliza os módulos Codewarrior I²C, Serial EIA/TIA-232, ADC e timer. O endereço do módulo no barramento I²C foi configurado como 0x59h. Uma vez que o módulo não utiliza um oscilador a cristal, a interface I²C foi configurada para utilizar estes pinos.

O módulo Codewarrior Serial EIA/TIA-232 foi configurado para dados de oito bits, um bit de parada, sem paridade. No momento da compilação, pode-se escolher as velocidades típicas no intervalo de 300 a 115200bps. O módulo Codewarrior ADC é utilizado para ler a tensão gerada pela referência bandgap e pelo sensor de temperatura internos ao microcontrolador.

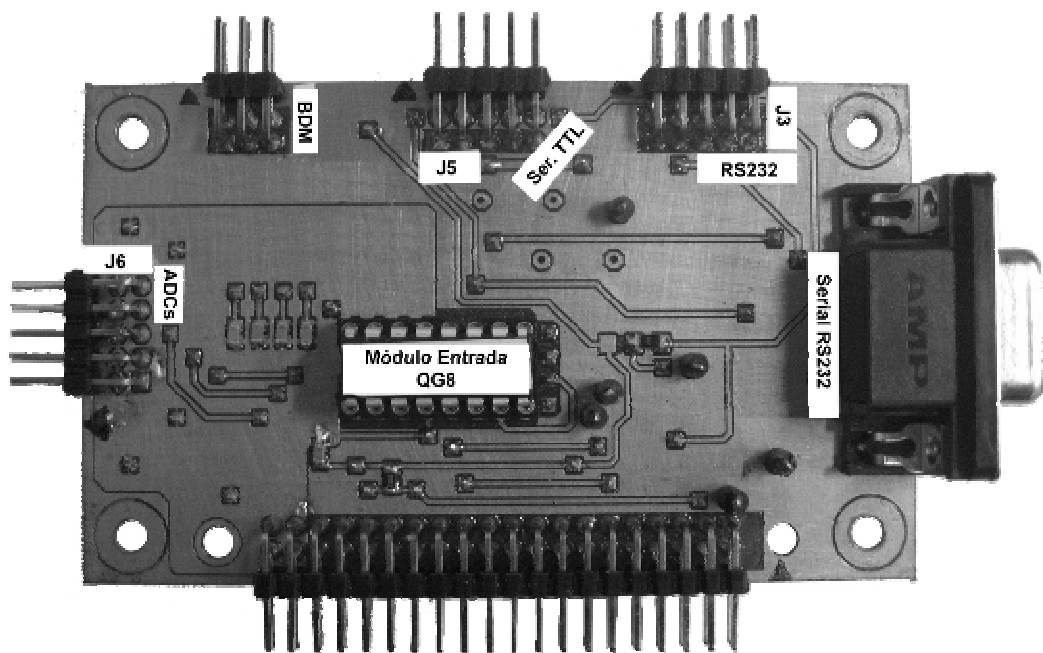


Figura 4-13 – Foto do Módulo Entrada.

4.6 MÓDULO INTERFACE / MÓDULO TRANSMISSÃO / MODEM GPRS

O Módulo Interface destina-se a enviar os dados medidos para uma estação remota. Entretanto, este módulo ainda não foi implementado. Pretende-se utilizar um modem terminal GPRS, que utiliza a infraestrutura celular para transmitir dados e receber comandos. Uma vantagem do uso da tecnologia GPRS consiste na possibilidade de a tarifação tomar por base o volume de dados transmitidos. Alternativamente pode-se utilizar mensagens curtas de celular, SMS, para transmitir eventos específicos e receber comandos da estação remota. Neste caso, a tarifação baseia-se na quantidade de mensagens enviadas. O Sistema Caipora poderia enviar os dados medidos mais recentes, ou ainda, informações estatísticas acerca das medições de determinado período em intervalos pré-configurados de tempo. Ao receber um comando via SMS, retornar para a estação remota as informações solicitadas. Entranto, estas funções ainda não foram implementadas, e portanto fazem parte de desenvolvimentos futuros.

Como alternativa utilizou-se um microcomputador, conectado ao Sistema Caipora através da interface serial do Módulo Controle. Neste caso, quando decorre o período de tempo configurado no intervalo de amostragem, os dados medidos, assim como demais parâmetros, são enviados através desta interface serial. O programa de visualização recebe estes dados, efetua a conversão de unidades e disponibiliza para o usuário sob a forma de gráficos e de mostradores digitais.

4.7 MONTAGEM DO SISTEMA CAIPORA

A Figura 4-14 mostra a foto dos Módulos Alimentação (placa inferior), Controle, Armazenamento de Dados, Entrada e Condicionamento de Sinais (placa superior) conectados à Placa de Interconexão (ao fundo). Espaçadores, atravessados por parafusos fornecem rigidez mecânica.

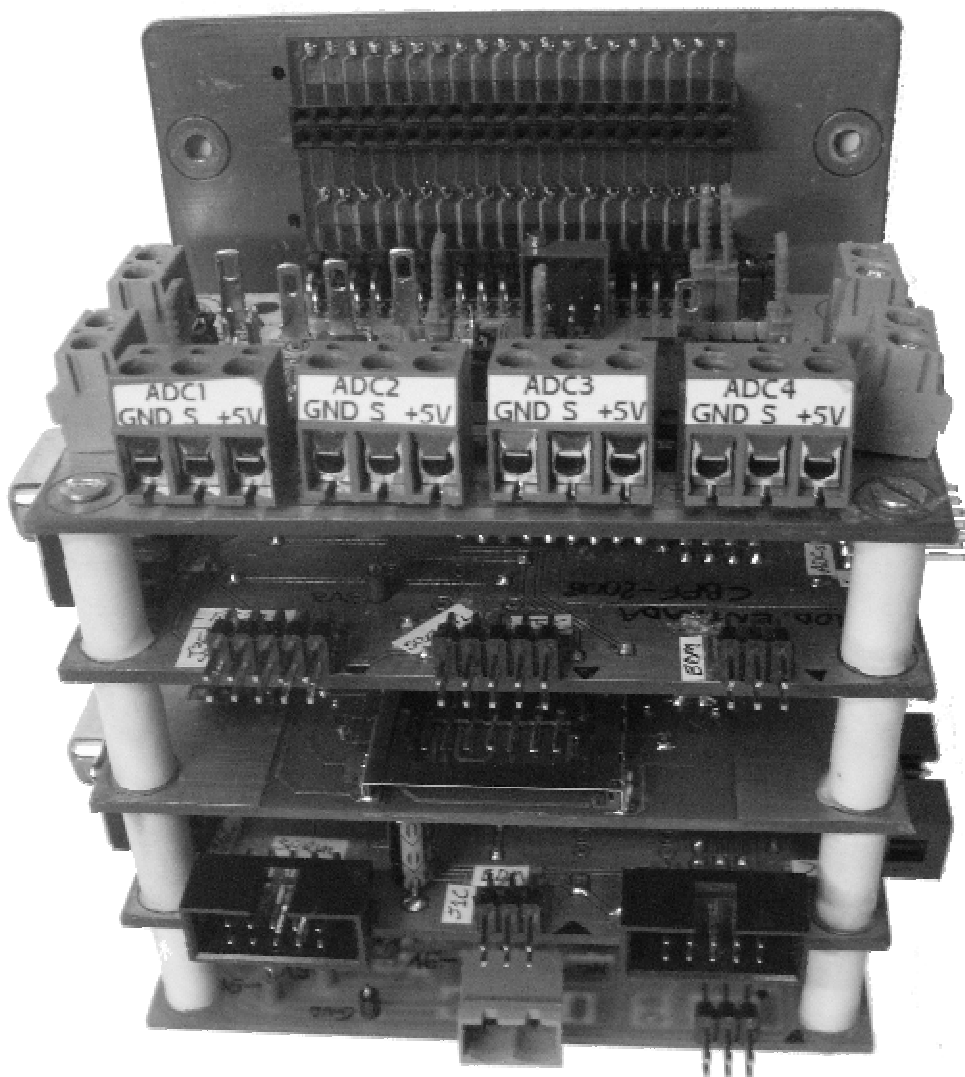
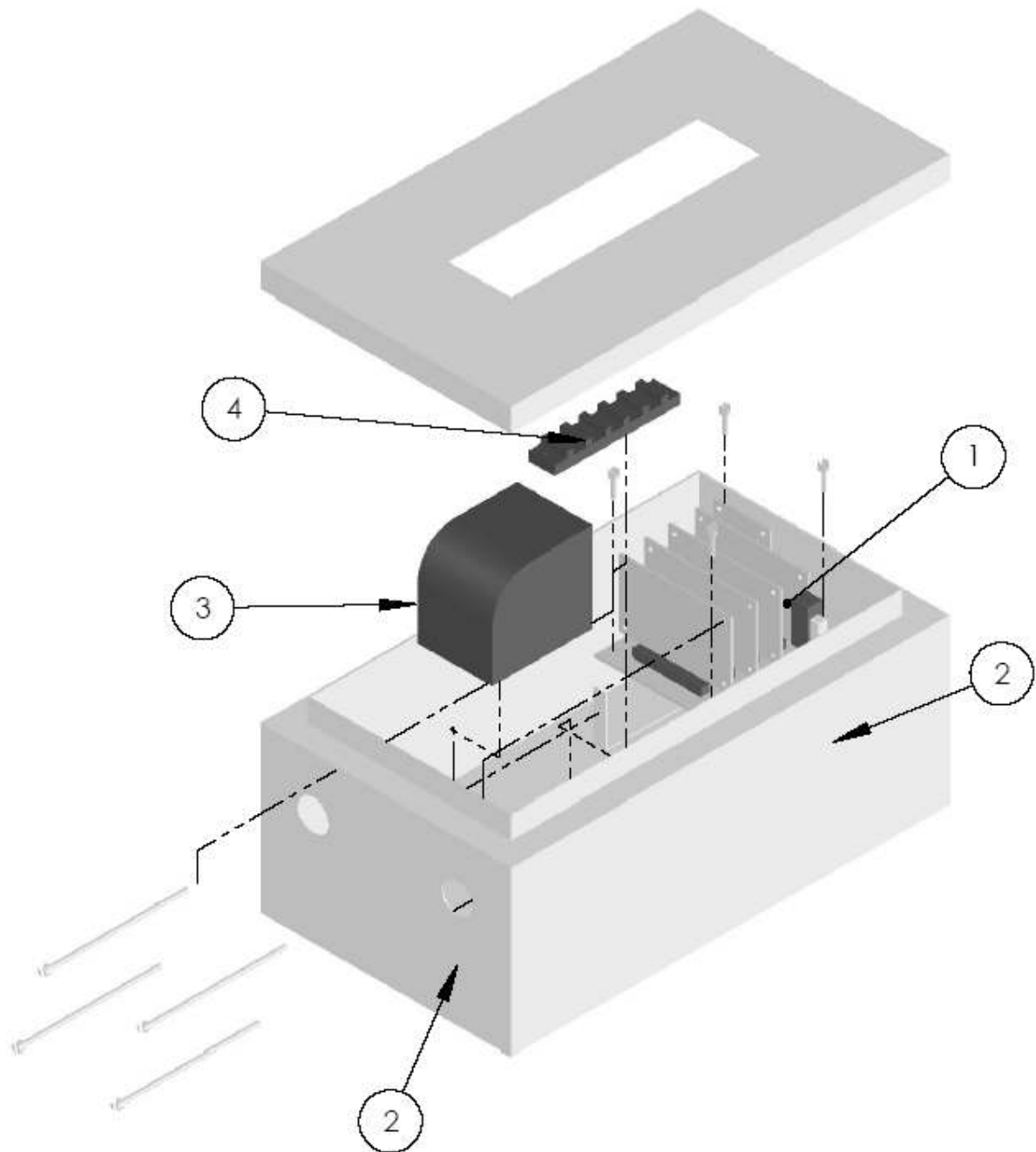


Figura 4-14 – Foto dos módulos conectados através da Placa de Interconexão.

A Figura 4-15 apresenta vista explodida da disposição dos diversos componentes do Sistema Caipora, acondicionados em caixa hermética de aço inox. Aparecem destacados os diversos módulos do sistema (1), conforme foto da Figura 4-14; o gabinete em inox (2); o carregador da bateria (3) e os bornes de conexão elétrica (4).



Fonte: Desenho elaborado por Marcelo Simões.

Figura 4-15 – Concepção da montagem do sistema.

A Figura 4-16 mostra a foto do interior do Sistema Caipora. À esquerda estão os módulos alimentação, controle, armazenamento, entrada e condicionamento de sinais. Na porção superior direita a barra de terminais com as conexões de entrada da rede elétrica e bateria. À direita são vistos os interruptores da energia elétrica da rede e da bateria do sistema, e os fusíveis de proteção do sistema. Sob estes, oculto na foto, está o carregador de bateria. As baterias, também ocultas na foto, estão localizadas sob a placa de fixação. O espaço livre, ao centro, destina-se ao módulo transmissão, que pode ser, por exemplo, um modem GPRS.

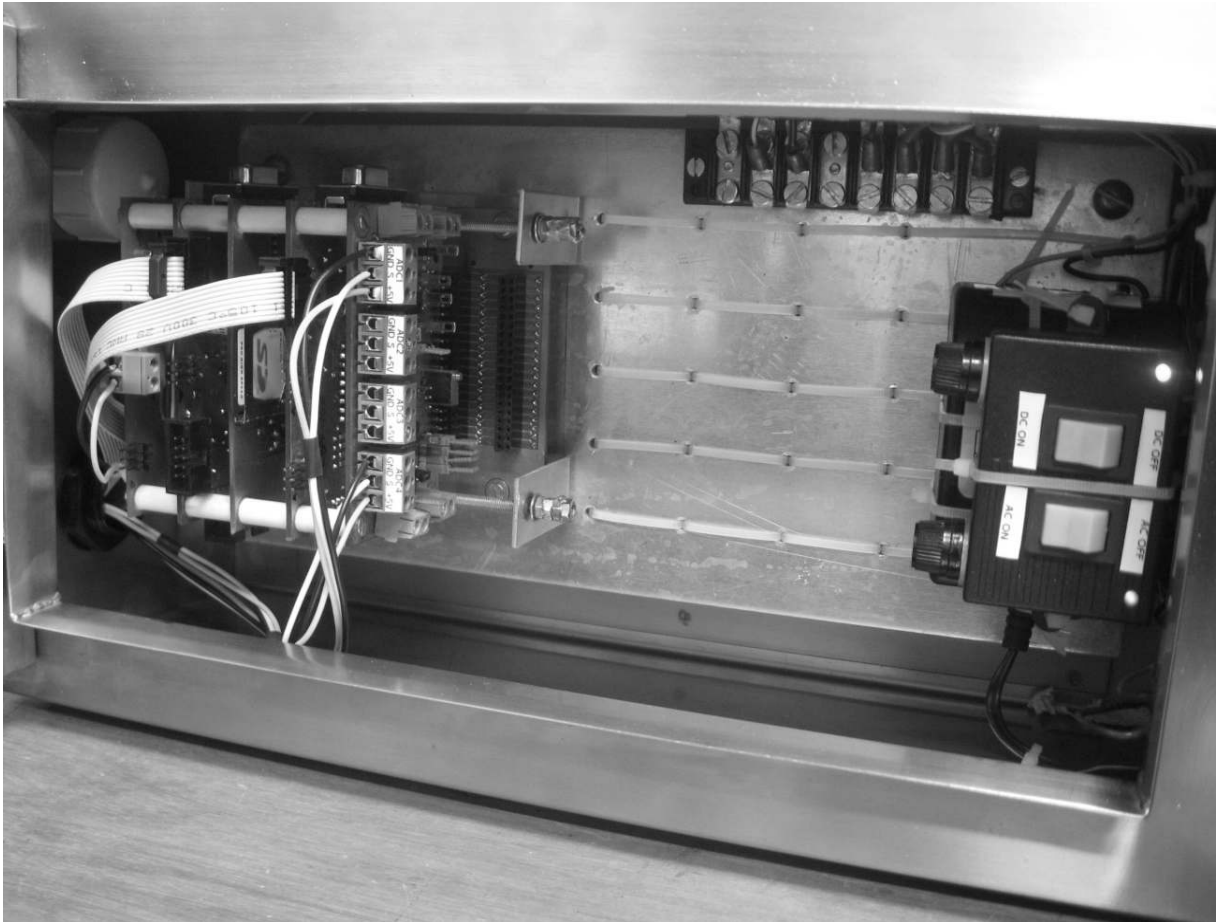


Figura 4-16 – Vista interna do Sistema Caipora.

A Figura 4-17 mostra a foto do Sistema Caipora (esquerda) operando conectado a um microcomputador (direita).



Figura 4-17 – Foto do sistema operando conectado a um microcomputador.

4.8 PROGRAMA DE VISUALIZAÇÃO DE DADOS

O programa de visualização de dados, desenvolvido em linguagem gráfica Labview, permite a visualização dos dados conforme são adquiridos pelo sistema. O computador executando o Labview é conectado ao Módulo Controle do sistema através da interface serial TIA/EIA 232. Neste caso, o computador funciona como o módulo transmissão mostrado na Figura 4-1. A Figura 4-18 mostra o painel de configuração da interface serial, através da seleção da porta e do *baud rate*. O programa configura ainda a porta serial EIA/TIA-232 para dados de 8 bits, sem paridade e sem controle de fluxo, sem apresentar opção para alterar estes parâmetros.

Cada vez que um registro é gravado no cartão de memória, o Módulo Controle envia uma *string* semelhante à armazenada pela interface serial. O software de visualização de dados recebe esta *string*, armazena-a em um arquivo no microcomputador e a decodifica. A partir destes dados recebidos, os valores medidos são convertidos para tensão, de acordo com a faixa selecionada, resolução e tensão de referência do ADC. Para cada canal de ADC

(referente aos dados do Módulo Entrada) é gerado um gráfico, conforme mostrado na Figura 4-19.

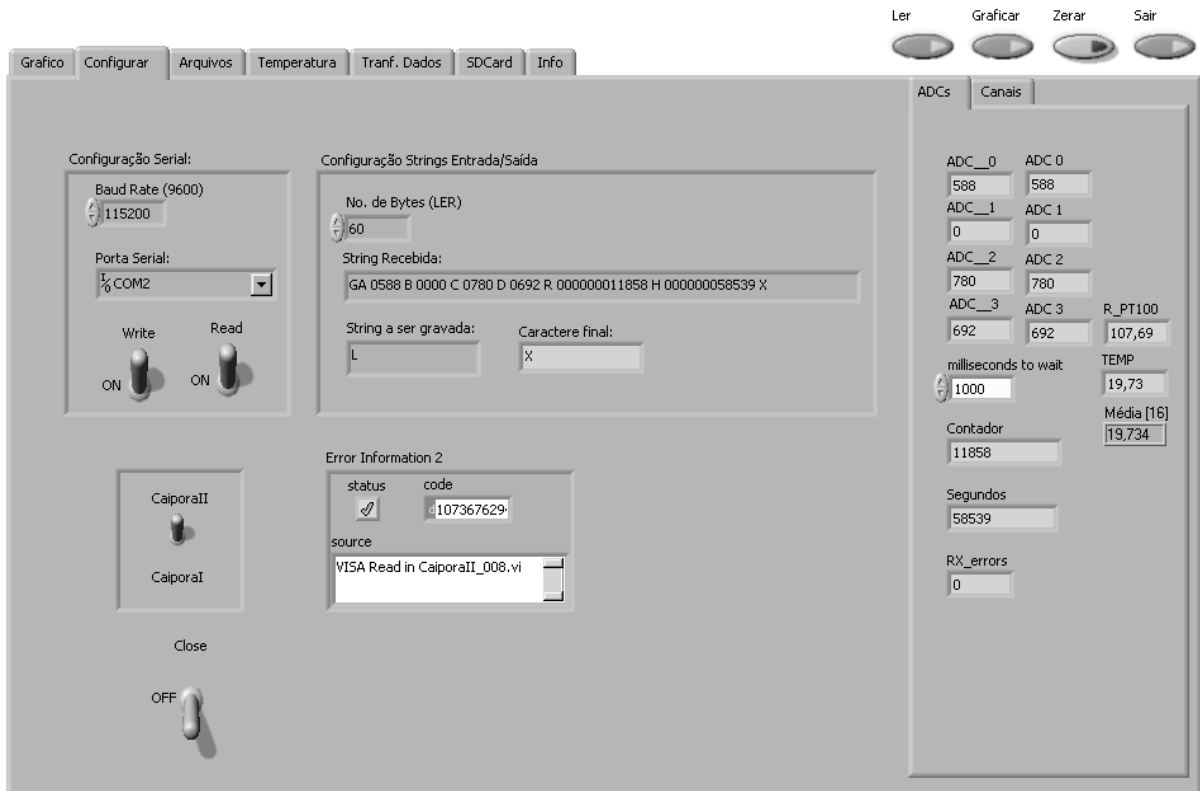


Figura 4-18 – Painel de configuração da interface serial.

O programa também converte o valor da temperatura ambiente do sistema, lido pelo sensor TMP101, do formato hexadecimal recebido para temperatura [°C]. Outros parâmetros que são visualizados com o programa são o endereço do cartão de memória onde os dados foram salvos e a data e hora da medição. O programa permite ainda salvar as *strings* recebidas, contendo os dados medidos, diretamente em um arquivo no computador.

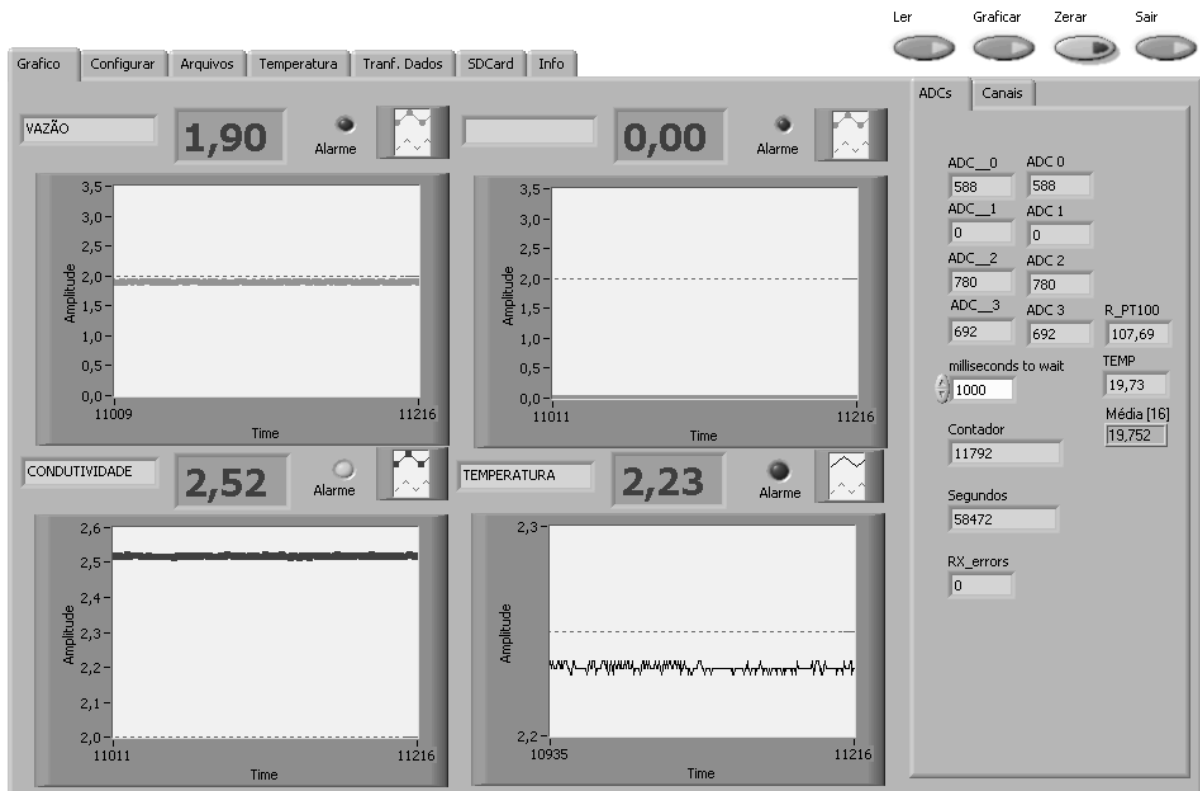


Figura 4-19 – Painel de visualização dos dados.

4.9 CONSUMO DE ENERGIA

A Tabela 4-6 apresenta uma estimativa do consumo de energia do sistema. Os módulos com maior consumo de energia são o de condicionamento de sinais e o cartão de memória. No Módulo Condicionamento de Sinais isto se deve à alta corrente quiescente por canal do amplificador operacional TL084. A substituição dos CIs por equivalentes de baixa energia, por exemplo LP324 e LP2902, permitiria reduzir o consumo em cerca de 10mA, uma vez que apresentam corrente quiescente típica por canal de 0.3mA, contra 1.4mA do TL084.

A coluna ‘Corrente Quiescente Estimativa’ apresenta a o valor de corrente estática consumida pelos módulos. O valor em parênteses para o SDCard refere-se ao consumo do módulo durante leitura/escrita. A coluna ao lado, ‘Corrente Quiescente Medida’ apresenta os valores das respectivas correntes medidas individualmente. O valor entre parênteses para os módulos Entrada e Controle refere-se ao consumo quando o cabo serial está conectado aos módulos. Isto se deve à função *autoshtutdown*, que habilita os drivers da porta serial somente quando níveis válidos TIA/EIA–232 são detectados na porta. A medição foi feita diretamente a partir da alimentação fornecida pela bateria ao Sistema. Por fim, a coluna ‘Máximo Calculado’ apresenta a corrente máxima prevista para o módulo. Destaca-se nesta coluna o

fato de o valor máximo calculado para o Módulo Condicionamento de Sinais ser inferior à corrente quiescente medida. Isto deve-se, provavelmente, a consumo por parte dos amplificadores operacionais acima do valor típico publicado na folha de especificação.

Tabela 4-6 – Estimativa do consumo de energia.

MÓDULO	CORRENTE QUIESCENTE ESTIMATIVA [mA]	CORRENTE QUIESCENTE MEDIDA [mA]	MÁXIMO CALCULADO [mA]
Alimentação	10	9,4	22
Controle	4	4,1 (7,2) ^e	10
Entrada	4	4,3 (7,6) ^f	10
Condic. Sinais	11,2	28,2	23 ^g
SDCard	5 (75)	1,6	104
Total:	34	47 (54)^e	169

O cartão de memória do Módulo Armazenamento, embora apresente alto consumo de energia, permanece habilitado apenas por uma pequena fração de tempo. Por exemplo, a gravação de um bloco de 512 *bytes* demora cerca de 4500 ciclos de clock da interface SPI, incluindo envio do comando de escrita do bloco especificando o endereço de destino, 512 *bytes* de dados do bloco, dois bytes de CRC, e um tempo variável até que o cartão esteja pronto para gravar os dados. Se a frequência do barramento SPI for de 4MHz, a escrita do bloco dura cerca de 1.1ms. Supondo que o módulo escreva um bloco por segundo, e consuma 75mA durante a gravação e 1,6mA quando disponível, o consumo médio será de apenas 1,68mA, conforme equação (4-4).

$$I_{\text{média}} = \frac{75mA \cdot 1,1ms + 1,6mA \cdot 998,9ms}{1000ms} \quad \therefore I_{\text{média}} = 1,68mA \quad (4-4)$$

^e Conectando-se a porta serial do Módulo Controle ao computador.

^f Conectando-se a porta serial do Módulo Entrada ao computador.

^g Desconsiderando operação no modo 4-20mA.

5

RESULTADOS E DISCUSSÕES

5.1 CARACTERIZAÇÃO DO MÓDULO CONDICIONAMENTO DE SINAIS

A caracterização do Módulo Condicionamento de Sinais foi feita utilizando-se gerador de funções Agilent 33120A, conectado à entrada de cada canal, medindo-se a tensão de saída do respectivo canal com multímetro Keithley 175A. O gerador de funções foi configurado para impedância de saída alta. O multímetro foi ajustado na escala 0-20VDC. O módulo foi alimentado com fonte ICEL PS-7000, com tensão de saída ajustada em 8,00V. A tensão medida para alimentação -5V, gerada pelo Módulo Alimentação através do CI MAX660, foi de -4,792V; a de +5V foi de 5,006V. Nesta etapa utilizou-se o Módulo de forma isolada, sem estar conectado ao restante do sistema.

Efetuuou-se com o gerador de funções, para cada canal, a varredura da tensão de entrada de -3,3 a +3,3V (ganho unitário), -5,0 a +5,0V (ganho 0,643), -10,0V a +10,0V (ganhos 0,320 e 0,152), de modo de que a faixa de tensão de saída do circuito estivesse compreendida entre -3,3V e +3,3V.

A Tabela 5-1 reúne os resultados do ajuste linear para cada canal de acordo com o ganho selecionado.

Tabela 5-1 – Resultados da caracterização do Módulo Condic. de Sinais – Modo tensão.

FAIXA	GANHO (PREVISTO)	CANAL A		CANAL B		CANAL C		CANAL D	
		Ganho [V/V]	Offset [mV]	Ganho [V/V]	Offset [mV]	Ganho [V/V]	Offset [mV]	Ganho [V/V]	Offset [mV]
3.3V	1,0000	1,0000	-0,72	0,9999	-5,88	0,9969	1,32	1,0018	-3,08
5V	0,6429	0,6435	-0,68	0,6438	-6,24	0,6423	0,68	0,6456	-3,40
10V	0,3197	0,3201	-0,26	0,3200	-5,96	0,3193	0,56	0,3216	-4,07
20V	0,1525	0,1532	-0,63	0,1531	-6,33	0,1526	-0,30	0,1534	-3,93

Para a caracterização da faixa 0 a 20mA utilizou-se o mesmo gerador de funções e multímetro. Entretanto, em série com o gerador de função foi inserido um multímetro Keithley 199, configurado para medir a corrente fornecida pelo gerador de funções. O *jumper* do Módulo Condicionamento de Sinais que insere o resistor de 147Ω na entrada do canal foi instalado, configurando-o para a faixa de corrente 0 a 20mA. Ajustou-se a tensão de saída do gerador de funções até que multímetro indicasse na tela a corrente desejada. Mediu-se a tensão de saída para as correntes 0 a 24mA, em passos de 2mA. Todos os canais foram testados na posição de ganho unitário. Não recomenda-se utilizar ganhos inferiores ao unitário para o modo de corrente, uma vez que a dissipação máxima do encapsulamento do resistor do Módulo Condicionamento de Sinais limita a corrente a aproximadamente 40mA. Entretanto, pode-se utilizar um resistor externo em conjunto com outras faixas do módulo para medir outras faixas de corrente. A Tabela 5-2 apresenta os resultados do ajuste linear para os quatro canais.

Tabela 5-2 – Resultados da caracterização do Módulo Condic. de Sinais – Modo corrente.

GANHO (PREVISTO)	CANAL A		CANAL B		CANAL C		CANAL D	
	Ganho [V/A]	Offset [mV]	Ganho [V/A]	Offset [mV]	Ganho [V/A]	Offset [mV]	Ganho [V/A]	Offset [mV]
147,0	147,3	-9,9	146,9	-12,3	146,3	-7,8	147,5	-11,7

5.2 RESULTADOS DE MEDIÇÃO COM O SISTEMA

As medições a seguir foram feitas entre os dias 21/set/2008 e 7/out/2008. Utilizou-se cartão de memória de 1GB, suficiente para armazenar cerca de 1,9 milhão de medições. O intervalo de amostragem foi configurado para cerca de 30s, resultando em autonomia de cerca de 670 dias. O sistema foi conectado a um microcomputador, executando o programa de visualização de dados, através da interface serial do Módulo Controle.

O gerador de funções Agilent 33120A, ajustado para $1,650V_{dc}$, foi conectado ao canal A do Módulo Condicionamento de Sinais. Um sensor de temperatura PT100 foi conectado ao canal B, porém estes dados foram desconsiderados uma vez que não se utilizou em paralelo um sistema de medição de temperatura que pudesse servir de referência na análise posterior. A saída de um circuito divisor de tensão, formado por dois resistores de valor nominal 8k25, foi conectado ao canal C. O divisor de tensão foi conectado à tensão de alimentação +5V e à referência de terra. O Canal D foi aterrado (conectado a 0V). Os canais A e D utilizaram a faixa unipolar 0 a 3.3V, enquanto os canais B e C utilizaram a faixa 0 a 5V.

Também foram medidas e armazenadas, com o mesmo intervalo de amostragem, as tensões da bateria, tensão +5V, tensão das referências *bandgap* internas aos microcontroladores dos módulos alimentação e entrada, temperatura interna do microcontrolador dos módulos entrada e alimentação e a temperatura ambiente do sistema.

A leitura da tensão de referência 1,65V, conectada ao canal A, fornece uma estimativa da flutuação da referência de tensão utilizada pelo conversor analógico digital. A Tabela 5-3 apresenta as frequências dos resultados das medições do canal A. O histograma correspondente pode ser visto na Figura 5-1. A Figura 5-2 apresenta o resultado das medições da tensão 1.65V entre os dias 21/set/2008 e 7/out/2008.

Tabela 5-3 – Frequências dos resultados da conversão da tensão de Referência 1,65V.

VALOR LIDO:	TENSÃO:	CONTAGENS	%
510	1,643	28	0,1
511	1,646	2416	5,6
512	1,650	39745	92,3
513	1,653	864	2,0
TOTAL:		43053	100,0

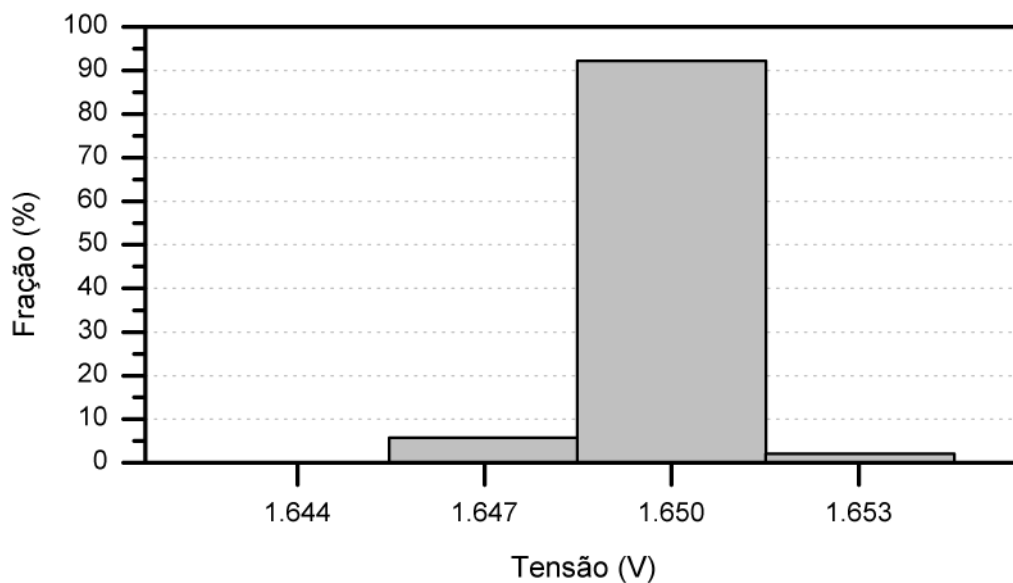


Figura 5-1 – Histograma da tensão de referência 1.65V.

O gerador de funções utilizado, modelo Agilent 33120A, apresenta exatidão de 2% da medida, somada de 2mV. O equipamento foi ligado uma hora antes do início das medições, respeitando o período de aquecimento especificado no manual, e operado dentro da faixa de temperatura ambiente de 18°C a 28°C, intervalo onde as especificações apresentadas são válidas. O canal utilizado, de acordo com a caracterização, apresenta cerca de -0,7mV de *offset*.

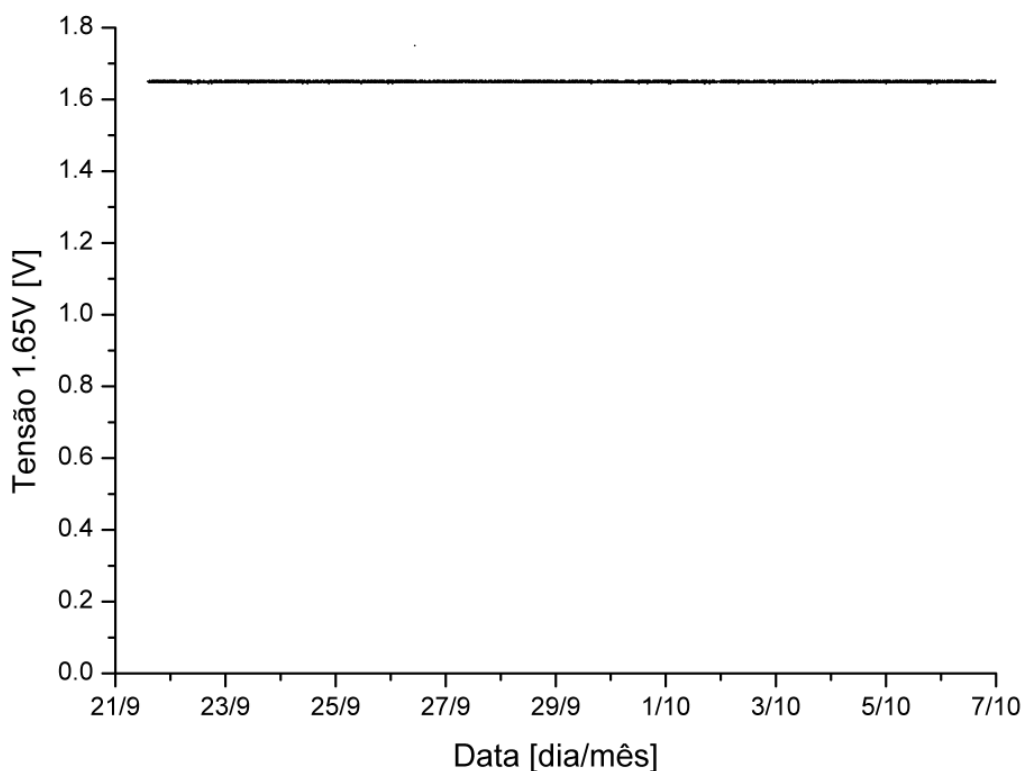


Figura 5-2 – Tensão de referência 1.65V, entre dias 21/set/2008 e 07/out/2008.

Um resultado de medição foi removido da análise, uma vez que decorreu da alteração não intencional do valor ajustado no gerador de funções de 1,65V para 1,75V. O evento foi anotado, e a medida, coerente com a data e a hora anotadas, foi desconsiderada. Para o canal A, os valores obtidos pelo ADC concentraram-se no valor 512 (1,65V), com ocorrências significativas uma contagem acima e uma contagem abaixo. A Figura 5-3 mostra um trecho da Figura 5-2, referente ao intervalo entre 14h e 16h do dia 23/set/2008. Observa-se que a predominância do resultado 1,650V, com oscilação no bit menos significativo do conversor analógico digital, resultando em leituras 1,646V e 1,653V.

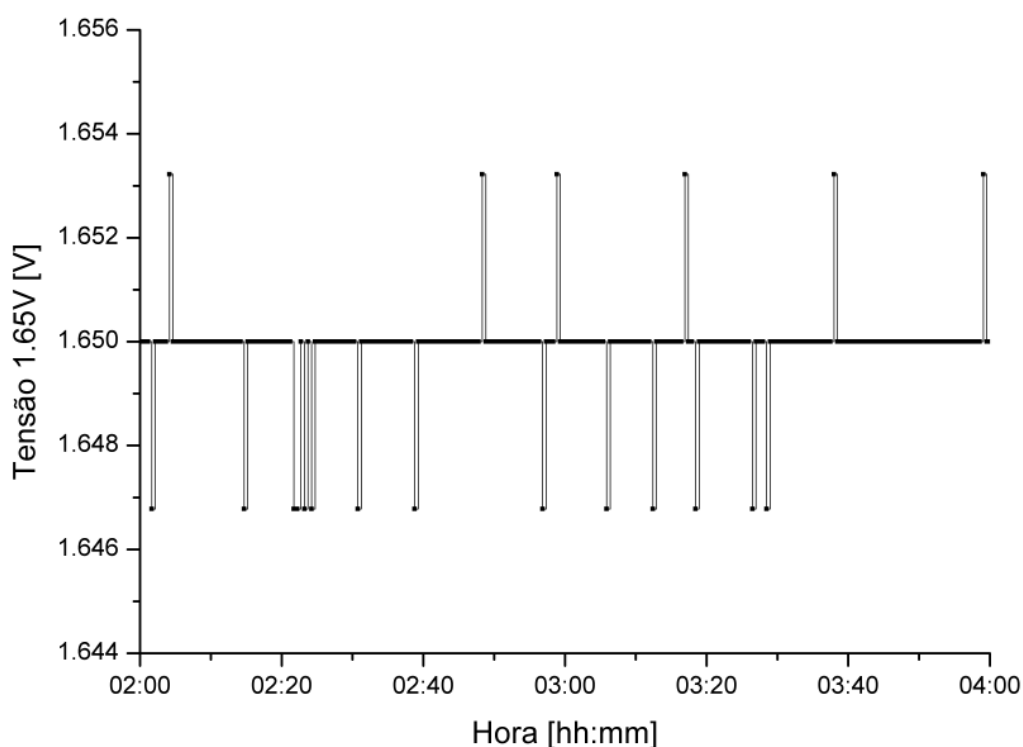


Figura 5-3 –Tensão de referência 1,65V, dia 23/set/2008, 14h a 16h.

A leitura do divisor de tensão conectado ao canal C fornece uma estimativa da flutuação da alimentação +5V. Uma vez que utilizou-se dois resistores semelhantes, estima-se que os coeficientes de variação da resistência com a temperatura sejam próximos e no mesmo sentido, fornecendo uma tensão proporcional à tensão que alimenta o divisor com pequena sensibilidade à variação da temperatura. Os resultados do canal C são apresentados na Tabela 5-4. Todas as medições do canal D, conectado à referência de terra retornaram o valor zero.

Tabela 5-4 – Frequências dos resultados da conversão do divisor de tensão.

VALOR LIDO:	TENSÃO:	CONTAGENS	%
474	2,376	211	0,5
475	2,381	12619	29,3
476	2,386	30176	70,1
477	2,391	48	0,1
TOTAL:		43054	100,0

Os resultados da leitura das referências de tensão bandgap internas dos microcontroladores dos Módulos Entrada e Alimentação são apresentados na Tabela 5-5. Observa-se que as leituras do Módulo Entrada apresentam dispersão menor em torno do valor predominante. Apenas três eventos desviaram mais de um lsb no módulo entrada, ao passo no Módulo Alimentação cerca de 2% das medições desviaram dois lsb, havendo um evento em que o desvio correspondeu a quatro lsb. Esta diferença pode ser atribuída ao uso de um regulador dedicado no Módulo Entrada.

Tabela 5-5 – Frequências dos resultados da conversão da referência interna bandgap do microcontrolador dos módulos Entrada e Alimentação.

VALOR LIDO:	TENSÃO: [V]	MÓDULO ENTRADA		MÓDULO ALIMENTAÇÃO	
		CONTAGENS	%	CONTAGENS	%
368	1,186	0	0,0	1	0,0
369	1,189	0	0,0	29	0,1
370	1,192	0	0,0	812	1,9
371	1,196	3	0,0	9029	21,0
372	1,199	1316	3,1	32003	74,4
373	1,202	31307	72,7	1111	2,6
374	1,205	10428	24,2	0	0,0
TOTAL:		43054	100,0	42985	100,0

A Figura 5-4 e a Figura 5-5 apresentam o resultado da medição da tensão de bandgap do microcontrolador do Módulo Alimentação e do Módulo Entrada, respectivamente, no período de 14h às 16h do dia 23/set/2008.

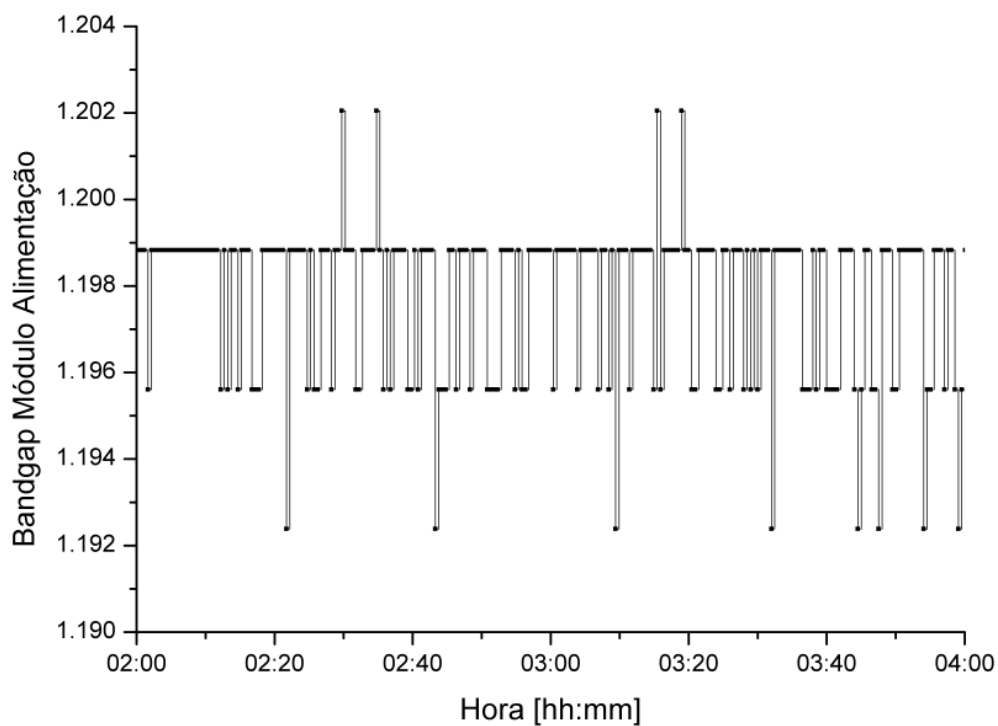


Figura 5-4 –Tensão bandgap do Módulo Alimentação, dia 23/set/2008, 14h a 16h.

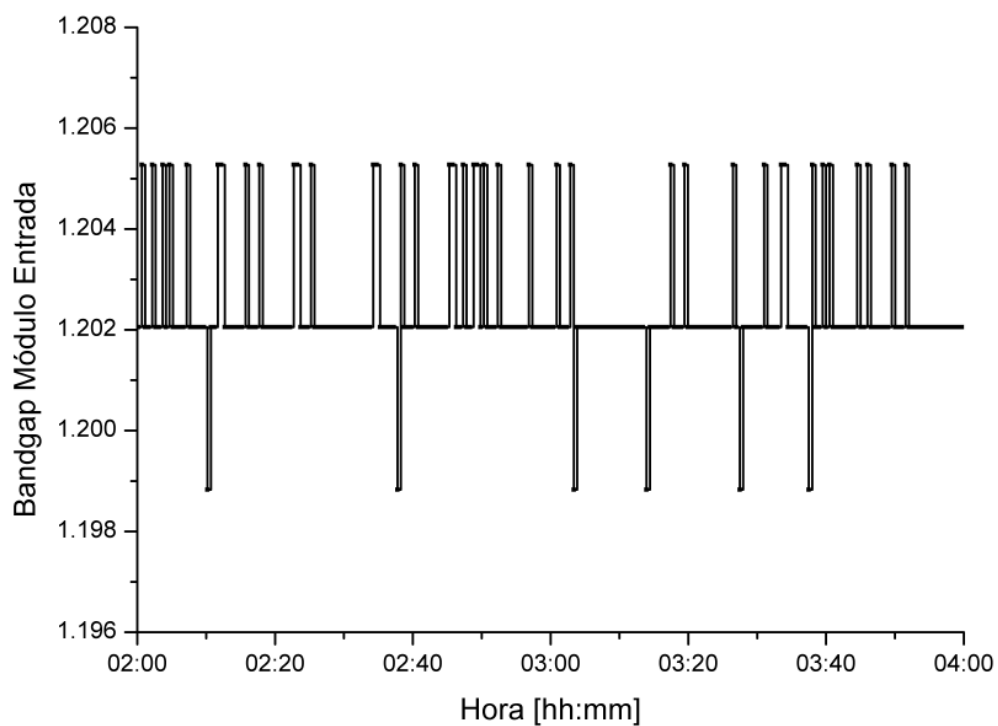


Figura 5-5 –Tensão bandgap do Módulo Entrada, dia 23/set/2008, 14h a 16h.

5.3 MEDIÇÃO DA ESTABILIDADE DO RELÓGIO DE TEMPO REAL

O procedimento para medir a estabilidade do relógio de tempo real do sistema consistiu em sincronizar o relógio de um computador com um servidor de tempo, através da internet. Ao fim deste procedimento, o relógio do computador está sincronizado com o do servidor, com erro de cerca de um segundo. Confronta-se então o tempo informado pelo sistema com o tempo do computador. Estima-se que este novo procedimento apresente erro adicional de um segundo. Desta forma, este procedimento permite medir apenas desvios acima de dois segundos.

O cristal utilizado, XT38T fabricado pela Vishay, apresenta tolerância de $\pm 20\text{ppm}$. O coeficiente parabólico para este cristal é $B = -0,042\text{ppm}/(\Delta^\circ\text{C})^2$ e a temperatura de referência é $T_i = 25^\circ\text{C}$. A temperatura média dentro do gabinete, medida pelo sensor TMP101, foi de cerca de 35°C . Substituindo-se estes valores na Eq. (2-1), obtém-se um desvio estimado de $-4,2\text{ppm}$, conforme Eq. (5-1).

$$\Delta\text{freq} = -0,042\text{ppm} \cdot (35 - 25)^2 \quad \therefore \quad \Delta\text{freq} = -4,2\text{ppm} \quad (5-1)$$

Ao longo de 12 dias e 5 horas o relógio do sistema atrasou $29\text{s} \pm 2\text{s}$, correspondendo a $-27,5\text{ppm} \pm 1,9\text{ppm}$. O desvio foi superior ao esperado, uma vez que cristal utilizado apresenta tolerância de $\pm 20\text{ppm}$ a 25°C . Mesmo levando-se em consideração o desvio devido à diferença entre a temperatura de operação e a temperatura T_i de referência, calculado com a Eq. (5-1), o valor obtido ficou acima do previsto.

5.4 AUTONOMIA DO SISTEMA QUANDO OPERADO SOMENTE POR BATERIAS

Uma estimativa da autonomia da bateria do Sistema pode ser obtida a partir da capacidade da bateria, expressa em amperes-hora (Ah). Desta forma, de acordo com a Eq. (5-2), a bateria de $1,3\text{Ah}$ teria autonomia de cerca de 26 horas.

$$\text{Autonomia} = \frac{\text{Capacidade}}{\text{Corrente}} \quad \therefore \quad \text{Aut} = \frac{1,3\text{A} \cdot \text{h}}{0,05\text{A}} \quad \therefore \quad \text{Aut} = 26\text{h} \quad (5-2)$$

O ensaio para determinar o intervalo de tempo que a bateria mantém o Sistema em operação consistiu em desligar a chave da alimentação da rede elétrica durante a operação do sistema. Desta forma, simulou-se a falta de energia e pôde-se verificar a validade dos dados

armazenados no cartão de memória. Após a interrupção da alimentação, o Sistema armazenou cerca de 6200 medições no cartão de memória, equivalente a mais de 51 horas de aquisição. Entretanto, após cerca de 5900 medições a tensão da bateria atingiu 6.6V. Uma vez que o regulador de tensão precisa de uma margem para operar, a tensão +5V do Sistema começou a baixar a partir deste ponto. Desta forma, embora a tensão de referência do microcontrolador ainda estivesse estável, as medições posteriores a este ponto não são confiáveis pois os sensores utilizados podem necessitar da alimentação +5V.

Entretanto, para evitar a redução da vida útil da bateria, deve-se evitar descarregá-la abaixo de cerca de 11V. Utilizando este critério, teriam sido feitas cerca de 5400 medidas. Uma vez que o intervalo utilizado foi de uma medida a cada 30s, a autonomia obtida seria de cerca de 45h. A diferença entre a autonomia esperada de 26h e a medida de 45h deve-se à diferença entre a corrente utilizada para especificar a capacidade da bateria (cerca de 1A) e a corrente drenada pelo sistema, cerca de 0,05A. A eficiência da bateria é maior quando descarregada com correntes menores.

6

CONCLUSÃO

Neste trabalho descrevemos a proposta de um meio alternativo para o monitoramento da poluição ambiental, baseado na medição, registro, transmissão e monitoramento contínuos de parâmetros físico-químicos indicadores da emissão de poluentes ou atmosféricos, ou solo ou da água. Com este intuito, desenvolveu-se um dispositivo capaz de amostrar, armazenar e transmitir sinais analógicos ou digitais provenientes de transdutores certificados. As atividades concentraram tarefas de projeto e construção de circuitos eletrônicos, programação de microcontroladores e de processamento e interface com o usuário. Foram desenvolvidos os Módulos Alimentação, Controle, Entrada, Condicionamento de Sinais, Armazenamento, além da placa de Interconexão dos módulos. O emprego do sistema em aplicações de monitoramento ambiental permitirá às autoridades exercer um maior controle acerca da emissão de substâncias nocivas à vida humana e ao meio ambiente.

Os módulos desenvolvidos desempenharam corretamente as funções projetadas. Nesta primeira etapa de implementação, o Módulo Condicionamento de Sinais se mostra limitado para sinais unipolares. O Sistema apresentou autonomia de cerca de 45 horas quando alimentado apenas pela bateria, valor adequado para aplicações onde haja disponibilidade de infraestrutura de rede elétrica. Caso haja a necessidade de utilizar o Sistema em locais em que a rede elétrica não esteja disponível, faz-se necessário empregar um meio alternativo para carregar as baterias, por exemplo painéis solares e/ou baterias com maior capacidade de carga.

Num sistema de aquisição de dados o instante em que a amostragem do sinal foi feita tem importância comparável à da própria medida. Desta forma, foi dada atenção especial ao

circuito de relógio do sistema, incorporado no Módulo Alimentação. O desvio medido do relógio da ordem de 30s em 12 dias mostra-se fora dos limites especificados pelo fabricante do cristal, elemento central do relógio. Caso venha a ser acoplado um GPS ao sistema, a informação de data e hora do Sistema pode ser obtida com elevada precisão diretamente a partir deste.

A calibração do Módulo Condicionamento de Sinais mostrou pequena variação entre o ganho projetado e o medido para os quatro canais. Os canais apresentaram offset entre 1 e 6mV, que podem ser compensados posteriormente no tratamento dos dados, sabendo-se de antemão o canal utilizado. O histograma referente à tensão do bandgap interno do microcontrolador do Módulo Entrada apresentou dispersão inferior ao do microcontrolador do Módulo Alimentação, sugerindo redução do ruído associado à medição devido ao uso de uma referência dedicada neste módulo.

Uma vez que o Sistema terá a responsabilidade fiscal do meio-ambiente, é imprescindível que os dados sejam armazenados de forma confiável e inviolável. Neste sentido, o fato de os dados armazenados no cartão de memória corresponderem aos dados enviados pelo Sistema através da interface serial atestam a confiabilidade do sistema. Por outro lado, a distribuição do processamento entre diversos microcontroladores garante que apenas o Módulo Controle tenha acesso ao Módulo Armazenamento, contribuindo para a inviolabilidade dos dados armazenados.

Ao aliar as funções propostas de medição, registro e transmissão de dados com a atratividade econômica, o sistema qualifica-se para ser amplamente empregado em atividades de monitoramento e fiscalização ambiental. Pretende-se, após aprimoramentos, que o sistema seja homologado frente às instituições competentes e sirva de modelo de referência, abrindo caminho para o monitoramento e fiscalização efetivos de atividades poluidoras do meio ambiente.

A cooperação entre o Instituto Nacional de Tecnologia – INT e o Centro Brasileiro de Pesquisas Físicas – CBPF, ambas unidades do Ministério da Ciência e Tecnologia – MCT, constitui sinergia voltada para o desenvolvimento de uma solução nacional para um problema tecnológico.

A partir de uma demanda da Secretaria da Receita Federal ao Instituto Nacional de Tecnologia no âmbito do Sistema de Medição de Vazão – SMV, decorreram desdobramentos do projeto original que resultaram em alternativas interessantes às instituições. O desenvolvimento de ferramentas tecnológicas pode se mostrar indutor de políticas públicas e de processos de fiscalização.

Embora o sistema implementado incorpore funções de medição, armazenamento e transmissão dos dados, diversos aprimoramentos são necessários. No sistema em operação os dados são armazenados no cartão de memória, e uma cópia é enviada através da interface serial para um dispositivo remoto. A implementação de um Módulo Interface permitiria que, por exemplo, um modem GPRS fosse conectado ao sistema. Desta forma, ao receber um comando através do modem GPRS, o Módulo Interface solicitaria os dados ao Módulo Controle. Os dados seriam lidos pelo Módulo Controle e repassados para o Módulo Interface, que os transmitiria.

Para aplicações em que haja deslocamento físico do sistema faz-se necessário efetuar a leitura da posição geográfica do sistema, a qual pode ser fornecida por um GPS (*Global Positioning System*). O GPS seria conectado à porta TIA/EIA-232 do Módulo Entrada, sendo necessário alterar o programa executado no microcontrolador deste módulo para selecionar os dados desejados.

O circuito implementado do Módulo Condicionamento de Sinais precisa ser expandido para permitir a conexão de sensores com sinal de saída bipolar. Outra função útil seria identificar automaticamente qual a configuração de cada canal do Módulo Condicionamento de Sinais (unipolar, bipolar, faixa de tensão de entrada (ganho), modo tensão ou corrente, etc) no início de cada aquisição. Estes dados poderiam ser armazenados num cabeçalho da aquisição, o que simplificaria o processo de tratamento dos dados.

Para que o sistema seja efetivamente utilizado em aplicações de monitoramento ambiental faz-se necessário utilizá-lo em conjunto com sensores de dióxido de carbono, monóxido de carbono, pH, temperatura, umidade, dentre outros, homologados por instituições de referência. Caso seja necessário converter o sinal analógico com resolução maior, um Módulo Entrada alternativo poderia ser desenvolvido, incorporando ADC com resolução maior ou com suporte a medições diferenciais. O Módulo Entrada calcularia algumas funções estatísticas, por exemplo máximo, mínimo, média, para os canais de entrada analógica.

Uma vez que atualmente o sistema armazena os dados em formato de texto, há a possibilidade de aumentar a quantidade de dados armazenados caso fosse utilizado o formato binário. Para tanto, bastaria alterar o código executado no microcontrolador do Módulo Controle. Outra melhoria consistiria em alterar o módulo controle para que sejam utilizados todos os 512 *bytes* de cada bloco do cartão de memória. Para aumentar segurança e sigilo dos dados, estes seriam codificados e/ou criptografados.

Outra área que permite melhorias consideráveis reside no consumo de energia. A substituição de componentes com consumo elevado por outros com função semelhante mais

econômicos, além da habilitação, quando possível, dos modos de baixo consumo nos dispositivos empregados no sistema, aumentaria a autonomia do sistema quando operado diretamente a partir da bateria.

REFERÊNCIAS

- [1] BRASIL. Secretaria da Receita Federal. Instrução normativa SRF nº 265, de 20 de dezembro de 2002. Dispõe sobre a instalação de equipamentos medidores de vazão e condutivímetros de que trata o art. 36 da Medida Provisória nº 2.158-35, de 24 de agosto de 2001. *Diário Oficial*, Brasília, DF, 24 dez. 2002.
- [2] MEDINA, Humberto. Leão guloso: arrecadação em 2005 cresce 5,7% e chega a R\$ 372 bilhões. *Jornal Folha de São Paulo*, São Paulo, 20 de jan. 2006. Folha dinheiro.
- [3] BRASIL. Ministério do Meio Ambiente. Resoluções do CONAMA. Brasília, DF, 2006.
- [4] BRASIL. Conselho Nacional do Meio Ambiente. Resolução CONAMA nº 5 de 15/jun/1989. Dispõe sobre o Programa Nacional de Poluição do AR. *Diário Oficial*, Brasília, DF, 25/ago/1989, p. 14713-14714.
- [5] BRASIL. Conselho Nacional do Meio Ambiente. Resolução CONAMA nº 357 de 17 de março de 2005. Dispõe sobre a classificação dos corpos de água e diretrizes ambientais para o seu enquadramento, bem como estabelece as condições e padrões de lançamento de efluentes, e dá outras providências. *Diário Oficial*, Brasília, DF, 18/mar/2005, seção 1, p. 58-63.
- [6] BRASIL. Conselho Nacional do Meio Ambiente. *Resolução CONAMA nº 3 de 28 de junho de 1990*. Dispõe sobre padrões de qualidade do ar, previstos no PRONAR. *Diário Oficial*, Brasília, DF, 22/ago/1990, p. 15937-15939.
- [7] RIO DE JANEIRO. Fundação Estadual De Engenharia Do Meio Ambiente. Relatório anual de qualidade do ar 2007. Rio de janeiro, 2008?.
- [8] BRASIL. Agência Nacional de Energia Elétrica. Resolução ANEEL nº 396 de 4 de dezembro de 1998. Estabelece as condições para implantação, manutenção e

operação de estações fluviométricas e pluviométricas associadas a empreendimentos hidroelétricos. *Diário Oficial*, Brasília, DF, 07/dez/1998, seção 1, p. 47, v. 136, n 234 – E.

- [9] BARRET, Steven; PACK, Daniel. *Microcontrollers fundamentals for engineers and scientists*. Morgan & Claypool publishers, 2006.
- [10] PEREIRA, Fábio. *Microcontroladores HCS08: teoria e prática*. São Paulo, Ed. Érica, 2005.
- [11] FREESCALE. *Folha de especificações do microcontrolador MC9S08QG8*. Freescale Semiconductor, versão 4, 2008.
- [12] SAFARI, Ahmad et al. Piezoelectric transducers and sensors. In: WEBSTER, John. *The Measurement, Instrumentation and Sensors Handbook CRCnetBase 1999*. CRC press LLC, 1999. p. 143-155.
- [13] VISHAY. *Folha de especificações do cristal Vishay XT38T*. Tunning Fork Crystal. Vishay Intertechnologh Inc., out. 2007.
- [14] SPANSION. *Flash Memory, an overview*. Rev. A0. Application note. November 2005.
- [15] SD GROUP. *SD specifications Part 1: Physical layer simplified specification*. Versão 2.00, Setembro de 2006.
- [16] TANEMBAUM, Andrew. *Computer networks*. Prentice Hall, 2nd ed., 1988.
- [17] ISO/IEC. *ISO/IEC 7498-1:1994(E)*. Information technology – open systems interconnection: Basic Reference Model: The Basic Model. 1996.
- [18] CROVELLA, Robert M. Sensors Networks and Communication. In: WEBSTER, John. *The Measurement, Instrumentation and Sensors Handbook CRCnetBase 1999*. CRC press LLC, 1999. p. 2331-2348.
- [19] JEDEC. Terms, definitions, and letter symbols for microelectronic devices. JESD99B. Maio de 1997.
- [20] MAXIM-DALLAS. Maxim AN83 – *Fundamentals of RS-232 serial communications*. 2001.
- [21] MAXIM-DALLAS. Maxim AN2141 – *Determining clock accuracy requirements for UART communications*, aug. 2003.
- [22] FREESCALE. *Serial peripheral interface (SPIV3) block description*. Freescale Semiconductor, versão 3.06, 2001.

- [23] CATSOULIS, John. *Designing Embedded Hardware*. O'Reilly, 2005. p. 160-173.
- [24] NXP SEMICONDUCTOR. *I2C-bus specification and user manual*. NXP Semiconductor. Versão 03, Junho de 2007.
- [25] SCHERZ, Paul. *Practical electronics for inventors*. McGraw-Hill Professional, 2006.
- [26] PROAKIS, J. G., MANOLAKIS, D.. *Digital signal processing, principles, algorithms, and applications*. Prentice Hall International, 1996.
- [27] KOSONOCKY, S., XIAO, P. Analog to digital conversion architectures. In: MADISSETTI, V.; WILLIAMS, D. *Digital signal processing handbook*. CRC Press LLC, 1999. p. 106-120.
- [28] HOLANDA, Aurélio Buarque de. *Novo dicionário Aurélio da língua portuguesa*. 2. ed. rev. aum. Rio de Janeiro: Nova Fronteira, 1986.
- [29] FRANK, Randy. Semiconductor junction thermometers. In: WEBSTER, John. *The Measurement, Instrumentation and Sensors Handbook CRCnetBase 1999*. CRC press LLC, 1999. p. 1024-1037.
- [30] TEXAS INSTRUMENTS. *Digital temperature sensor with I2C support*. Folha de especificações TMP101. Texas Instruments. 2002.
- [31] INSTITUTO NACIONAL DE METROLOGIA, NORMALIZAÇÃO E QUALIDADE INDUSTRIAL. *Vocabulário internacional de termos fundamentais e gerais de metrologia: portaria Inmetro nº 029 de 1995*. 5ª Ed. Rio de Janeiro: Ed. Senai, 2007.
- [32] HOROWITZ, Paul; HILL, Winfield. *The art of electronics*. Cambridge University Press, 2nd. Edition. 1989.
- [33] MICREL. *Folha de especificações do regulador MIC5205*. Fevereiro de 2006.
- [34] MAXIM-DALLAS. *Maxim application note 669*. Maxim Integrated Products. 2001.
- [35] BROOKS, Douglas. *Signal integrity issues and printed circuit board design*. New Jersey: Prentice Hall PTR, 2003.

APÊNDICE

APÊNDICE A – ESPECIFICAÇÕES ELÉTRICAS

A.1 ALIMENTAÇÃO:

Rede elétrica:	127VAC
Bateria:	12VDC

A.2 MÓDULO ENTRADA:

A.2.1 *ADC interno ao microcontrolador.*

Entrada de tensão com faixa entre	0 a 3,3VDC, referenciado ao terra.
Resolução:	10 bits – LSB = 3.22mV.
Acurácia inicial (Vref):	1%
Coef. de variação de Vref vs T:	40ppm / °C
Zin ADC do uC (típico):	5kΩ // 4.5pF
Impedância da fonte ligada ao ADC:	5kΩ

Compatível com padrão 4-20mA (0-22mA), mediante resistor de 147Ω.

A.3 MÓDULO CONDICIONAMENTO DE SINAIS:

A.3.1 *Conexão dos sensores^h:*

Seção mínima do fio:	0,2mm ²
Seção máxima do fio:	2,5mm ²
Comprimento desencapado do fio:	5mm

A.4 MÓDULO ARMAZENAMENTO DE DADOS:

Cartão de memória:	SDCard, tamanho padrão (24mm x 32mm), capacidade até 2GB (não SDHC).
--------------------	--

^h Definidos pela especificação do conector 'Phoenix Contact'.

A.5 ESTIMATIVA DO CONSUMO DE ENERGIA

A.5.1 *Módulo Alimentação:*

Tabela A.1 – Consumo de corrente Módulo Alimentação.

DISPOSITIVO	CORRENTE QUIESCENTE	MÁXIMO	SHUTDOWN
78M05	4,5 mA	6 mA	-
78M33	4,5 mA	6 mA	-
TMP101	45 μ A	75 μ A	1 μ A
MC9S08QG8 (cristal 32kHz, 3V3, Bus=1MHz, ADCoff)	1mA	10mA	35 μ A
MAX660	0,12mA	0,5mA	-
Pullups I ² C			
Total:	10mA	22mA	

A.5.2 *Módulo Controle:*

Tabela A.2 – Consumo de corrente Módulo Controle.

DISPOSITIVO	CORRENTE QUIESCENTE	MÁXIMO	SHUTDOWN
MC9S08QG8 (ICS, 3V3, Bus=9MHz ADCon)	4mA	10mA	35 μ A
MAX3223 (Autoshutdown habilitado)	1 μ A	10 μ A	-
Total:	4mA	10mA	-

A.5.3 *Módulo Entrada:*

Tabela A.3 – Consumo de corrente Módulo Entrada.

DISPOSITIVO	CORRENTE QUIESCENTE	MÁXIMO	SHUTDOWN
MC9S08QG8 (ICS, 3V3, Bus=9MHz ADCon)	4mA	10mA	35 μ A
MAX3223 (Autopowerdown habilitado)	1 μ A	10 μ A	-
MIC5205-3.3	125 μ A	150 μ A	5 μ A
Total:	4mA	10mA	

A.5.4 *Módulo Condicionamento de Sinais:*

Tabela A.4 – Consumo de corrente Módulo Condicionamento de Sinais.

DISPOSITIVO	CORRENTE QUIESCENTE	MÁXIMO	SHUTDOWN
TL084 x 8	1,4mA x 8	2,8 x 8	1,4mA x 8
MIC5205-ADJ	125 μ A	150 μ A	5 μ A
Total:	11,2mA	22,4mA	11,2mA

A.5.5 *Módulo Armazenamento:*

Tabela A.5 – Consumo de corrente Módulo Armazenamento.

DISPOSITIVO	CORRENTE QUIESCENTE	MÁXIMO	SHUTDOWN
TPS73033	170 μ A	250 μ A	1 μ A
24LC02B	1 μ A	1mA (leitura) 3mA (escrita)	1 μ A
SDCard	65mA (leitura) 75mA (escrita)	100mA	250 μ A
Total:	75mA	104mA	250 μ A