

Sistema de Gestão de Energia Elétrica - DataLogger

João Pedro Moreira da Cunha

Dissertação apresentado à
Escola Superior de Tecnologia e de Gestão
Instituto Politécnico de Bragança

para obtenção do grau de Mestre em
Engenharia Industrial - Ramo Engenharia Eletrotécnica

Este trabalho foi efetuado sob orientação de:
Professor Orlando Manuel de Castro Ferreira Soares
Professor José Luís Sousa de Magalhães Lima

Outubro de 2016

Sistema de Gestão de Energia Elétrica- DataLogger

João Pedro Moreira da Cunha

Relatório Final de Dissertação apresentado na
Escola Superior de Tecnologia e de Gestão
Instituto Politécnico de Bragança

para obtenção do grau de Mestre em
Engenharia Industrial - Ramo Engenharia Eletrotécnica

Este trabalho foi efetuado sob orientação de:
Professor Orlando Manuel de Castro Ferreira Soares
Professor José Luís Sousa de Magalhães Lima

Outubro 2016

Certifico que li este relatório e que na minha opinião, é adequado no seu conteúdo e forma como demonstrador do trabalho desenvolvido no âmbito da UC de Projeto.

<Nome do Orientador> Orientador

Certifico que li este relatório e que na minha opinião, é adequado no seu conteúdo e forma como demonstrador do trabalho desenvolvido no âmbito da UC de Projeto.

<Nome do Coorientador> Coorientador

Certifico que li este relatório e que na minha opinião, é adequado no seu conteúdo e forma como demonstrador do trabalho desenvolvido no âmbito da UC de Projeto.

<Nome do Arguente> Arguente

Aceite para avaliação da UC de Projeto

Agradecimentos

A realização desta dissertação marca o fim de uma importante etapa da minha vida, contudo contou com importantes apoios e incentivos e pelos quais estarei eternamente grato.

Aos Professores Doutores Orlando Soares e José Lima, pelas indicações, sugestões e correções que contribuíram para o desenvolvimento e conclusão da dissertação.

Ao Engenheiro António Morais pela disponibilidade, conhecimento e auxílio prestado.

Aos amigos pela amizade e companheirismo durante todo o meu percurso académico.

À Isabel pelo afeto e todo o apoio que me concedeu.

Por último um agradecimento especial aos meus pais e irmã pelos conselhos, estima e confiança que sempre depositaram em mim.

Obrigado a todos!

Resumo

No decorrer dos últimos anos, tem-se verificado um aumento significativo de dispositivos e de novas propostas para a implementação de edifícios inteligentes, com a necessidade de registrar dados provenientes desses mesmos dispositivos foram desenvolvidos mecanismos analógicos e, mais recentemente digitais para suprir essas necessidades, vulgarmente designados por DataLoggers, de forma a facilitar o registo e recolha de dados.

O DataLogger consiste em receber dados de um ou vários dispositivos, processar esses mesmos dados e armazenar em formato digital.

Esta dissertação tem como objetivo, a implementação de um DataLogger num Sistema Inteligente de Gestão de Energia Elétrica, que passa pela obtenção de dados provenientes do sistema inteligente, é inserida uma “etiqueta temporal” para que o utilizador possa tomar conhecimento quando alguma ação por parte dos dispositivos terá ocorrido. No final são armazenados num cartão SD genérico.

O trabalho não é focado para uma aplicação específica, contudo o DataLogger desenvolvido terá uso no projeto de investigação denominado por SIGEE proposto pela empresa Nanoquantum – Energias Renováveis Lda.

Palavras-chave: Datalogger, Arduino, SPI, Cartão SD, Armazenamento de dados.

Abstract

Over the past few years, there has been a significant increase in devices and new proposals for the implementation of intelligent buildings with the need to register their data with the development of analog mechanisms devices and more recently digital devices to address these needs, DataLoggers commonly called, to facilitate the registration and data collection.

The DataLogger propose is to receive data from one or more devices, process and store such data in a digital format.

This dissertation aims the implementation of a DataLogger in Intelligent Energy Management System, which obtains data from the intelligent system where is inserted a "time tag" for the user to be aware when an action by part of the devices has occurred. At the end that data is stored in a generic SD card.

The work is not focused to a specific application, yet the DataLogger developed will be used in the research project called for SIGEE proposed by Nanoquantum company - Renewable Energy Lda.

Keywords: DataLogger, Arduino, Card SD, Data Storage.

Índice

1	Introdução.....	17
1.1	Enquadramento	17
1.2	Objetivos e motivação.....	18
1.3	Organização do trabalho	18
2	Sistema de Gestão de Energia Elétrica numa habitação.....	19
2.1	Casa Residencial Inteligente	19
2.1.1	Objetivos das casas residências inteligentes	22
2.1.2	Vantagens das habitações inteligentes	23
3	Estrutura do Trabalho	25
3.1	Arduíno	25
3.2	Real Time Clock	27
3.3	Conversor DC/DC bidirecional.....	27
3.4	Cartão SD.....	29
4	Conceção do DataLogger	31
4.1	Implementação do circuito.....	31
4.2	Explicação do código	33
5	Simulação e Resultados obtidos	35
5.1	Resultados Obtidos	35
5.2	Simulação.....	37
5.2.1	Simulação Micro-ondas.....	38
5.2.2	Simulação Aquecedor	40
6	Conclusão e desenvolvimentos futuros	43
6.1	Conclusão do trabalho.....	43
6.2	Trabalhos futuros	44
7	Referencias bibliográficas	45
A	Código de Programação.....	1

Lista de Figuras

Figura 1 - Arquitetura de uma casa inteligente	22
Figura 2 - Arquitetura do Hardware do Arduino [5]	26
Figura 3 - Real Time Clock - DS1307 [6]	27
Figura 4 - Conversor DC/DC 3,3V e 5V	28
Figura 5 - Micro SD e adaptador de tamanho SD original	29
Figura 6 - Módulo SD Card	30
Figura 7 - Circuito DataLogger no Fritzing	32
Figura 8 - Primeira vez no <i>serial port</i>	35
Figura 9 - Modificação da data e hora no serial port	36
Figura 10 - Após modificação da data e hora	36
Figura 11 - Ficheiro de texto criado no Cartão SD	37
Figura 12 - Informação no Ficheiro de texto	37
Figura 13 - Sincronização com o RTC	38
Figura 14 - Informação de tomadas desligadas e respetiva gravação	38
Figura 15 - Tomada 1 (Micro-ondas) ligada	39
Figura 16 - Tomada 1 Desligada (Micro-ondas desligado)	39
Figura 17 - Informação no ficheiro de texto	39
Figura 18 - Após sincronização com RTC, tomadas desligadas	40
Figura 19 - Tomada 2 (Aquecedor) ligada	40
Figura 20 - Subida da temperatura, com tomada 2 ligada	41
Figura 21 - Descida da temperatura, com a tomada 2 desligada e tomada 3 (TV) ligada	41
Figura 22 - Temperatura estagnada	41
Figura 23 - Informação da simulação no ficheiro de texto no Cartão SD	42

Lista de Abreviações

SIGEE- Sistema Inteligente de Gestão de Energia Elétrica

IBI - Intelligent Buildings Institute

SPI – Serial Peripheral Interface

I²C – Inter-Integrated Circuit

RTC – Real Time Clock

SD – Secure Digital

SIGEE - Sistema Inteligente de Gestão de Energia Elétrica

MISO – Master in Slave Out

MOSI – Master Out Slave IN

SCK – Serial Clock

CS – Chip Select

SDA – Data line

SCL – Clock Line

RAM – Random Access Memory

EEPROM – Electrically Erasable Programmable Read Only Memory

HV – High Voltage

LV – Lower Voltage

USB – Universal Serial Bus

FAT – File Allocation Table

IDE - Integrated Development Environment

PCB – Printed Circuit Board

1 Introdução

1.1 Enquadramento

O ser humano depende das mais variadíssimas formas de energia tanto renováveis, como não renováveis sendo a energia elétrica aquela que mais suscita relevo para a sociedade atual. Para isso é preciso encontrar novos meios e métodos inovadores para que a utilização da energia seja feita de forma o mais sustentável possível.

Na questão do consumo energético, os edifícios são responsáveis por cerca de 40% da energia utilizada na maioria dos países [1]. Esta área de investigação tem tido um rápido crescimento, os edifícios habitacionais ao adquirir um carácter inteligente que podem dar um grande contributo para a regressão das alterações climáticas e promover a eficiência energética nos edifícios. Este tema poderá ter uma maior influência num futuro próximo, pois existe atualmente o conhecimento e tecnologia para reduzir a utilização de energia nas casas residenciais, melhorando em simultâneo os níveis de conforto [1]. Contudo, a resposta não deve depender apenas de uma alteração da produção, mas que se introduzam novos hábitos de consumo, principalmente no sentido de atenuar situações de pico ou tirar proveito da disponibilidade de fontes de energia renováveis, como por exemplo a energia solar fotovoltaica. Para isso o consumidor tem que ser informado e/ou persuadido a contribuir para este efeito, nomeadamente com a deslocação de consumos para períodos de menor procura ou instalar sistemas de energias renováveis. Mesmo assim só serão esperadas reações dos clientes com ganhos tangíveis em termos de poupança na fatura.

Para este problema surge como resposta o projeto, Sistema Inteligente de Gestão de Energia Elétrica para habitações. Este projeto tem como objetivo controlar os mais diferentes eletrodomésticos de uma habitação.

1.2 Objetivos e motivação

Este projeto consiste num Sistema Inteligente de Gestão de Energia Elétrica, numa habitação, criando assim um sistema inteligente que controle eletrodomésticos em períodos de menor procura e/ou períodos em que exista produção de energia a partir das fontes de energia renovável, para que a eficiência seja a maior possível, mesmo que o utilizador não esteja na sua habitação. No âmbito deste projeto foi-me proposto a criação de um DataLogger para armazenar, num Cartão SD, os vários dados provenientes do sistema, em que neste caso será feito o registo da mudança de estado das tomadas (ligada/desligada), como registar o consumo, a hora e a temperatura em que foi feita essa mudança a fim do utilizador tomar conhecimento.

1.3 Organização do trabalho

O presente relatório está organizado em 6 capítulos. No capítulo um é feita uma breve introdução acerca do projeto SIGEE, como também é feita a apresentação dos objetivos propostos para este trabalho (DataLogger). No capítulo dois é apresentado um estudo generalizado sobre o conceito de edifícios inteligentes que estão enquadrados no tema deste projeto.

O capítulo três consiste numa breve revisão sobre a tecnologia utilizada para a realização do DataLogger.

O capítulo quatro faz-se uma descrição do trabalho, explicando com maior detalhe como é constituído DataLogger a nível de hardware e software.

O capítulo cinco será apresentado os resultados, como também comentar os resultados obtidos da simulação.

O capítulo seis será destinado a conclusão do trabalho e será feita uma abordagem a trabalhos futuros.

2 Sistema de Gestão de Energia Elétrica numa habitação

Quando se fala em sistemas de Gestão de Energia Elétrica numa habitação a primeira ideia que suscita no pensamento será certamente “casas inteligentes”, que são definidas como sendo edifícios de habitação que possuem uma rede de comunicação entre todos os dispositivos da mesma, permitindo o controlo, a monitorização e o acesso remoto a todas as aplicações e serviços do sistema de gestão. O sistema de gestão deve incluir funções avançadas, como a previsão meteorológica, comunicação com sistema de contagem de energia (gás, água e luz), comunicação com eletrodomésticos através de tomadas inteligentes e por fim o registo de dados.

2.1 Casa Residencial Inteligente

Hoje em dia, o custo das construções e a conservação das casas urbanas tem custos muito elevados. Esta mesma construção e conservação vão trazer custos para as empresas que as possuem ou as utilizam. Como tentativa de reduzir esses custos o objetivo principal é a racionalização do projeto e a exploração dos edifícios.

No começo dos anos 60, nasceram os primeiros sistemas de controlo focado nos edifícios, com especial impacto em equipamentos de climatização. Já nos inícios dos anos 70, com o aparecimento de microprocessadores aumentou o poder de aplicação dos sistemas de controlo que permitiam a automação e o controlo de equipamentos mais complexos e em maior número [2].

Um dos fatores que contribuiu para o aumento da implementação destes sistemas foi a crise petrolífera. Dando assim maior importância a questões relacionadas com uma gestão energética mais sensata [2] [3].

Já nos anos 80 surgiu a conceção de casa residencial (casa inteligente), como resposta a necessidade de diminuir dos gastos da construção e de exploração. Surgiram então os designados sistemas de gestão técnica.

Juntamente com os sistemas de gestão técnica das habitações surgiram 2 fenómenos: surgiu o conceito de serviço ou funções realizadas por vários equipamentos de gestão técnica- o serviço iluminação, o serviço de controlo de acessos ou serviço de deteção de incêndios. Por outro lado, surgiu a necessidade de junção dos serviços, com o objetivo de obter novas funcionalidades resultantes das suas interações.

Ainda nos anos 80 apareceram novas exigências de conforto, segurança, flexibilidade dos locais de trabalho e novas necessidades de serviços de telecomunicações e de processamento de informação. Todas estas necessidades levaram ao aparecimento de três sistemas nas habitações [3]:

- O sistema de automação e gestão de edifícios, tendo como função o controlo das instalações técnicas, deteção de incêndios, gestão energética, controlo da iluminação, climatização, entre outros;
- O sistema de telecomunicações, abrangendo comunicações de voz e de dados, a comunicação com a parte exterior do edifício, entre outras.;
- O sistema computacional, de que fazem parte sistemas de informação, escritório eletrónico, sistemas de ajuda à decisão, automação de procedimentos administrativos, entre outras.

No ano 1986 foi fundada nos Estados Unidos da América a organização *Intelligent Buildings Institute*, IBI, tendo como função promover e ajudar em todos os aspetos relativos as casas residenciais inteligentes.

Um dos principais objetivos da organização era a criação de uma definição sólida para o conceito:

"Uma Casa Residencial inteligente é aquela que oferece um ambiente produtivo e que é economicamente racional, através da otimização dos seus quatro elementos básicos - estrutura, sistemas, serviços e gestão - e das inter-relações entre eles. As casas residenciais inteligentes ajudam os seus proprietários, gestores e ocupantes a atingir os seus objetivos sob as perspetivas do custo, conforto, adequação, segurança, flexibilidade no longo prazo e valor comercial" - (IBI,1986)

Com o conceito apresentado posteriormente, podemos apresentar vários aspetos importantes a ter em conta [3][4]:

- O conceito de inteligência durante todo o tempo de vida de habitação, tendo importante relevo na fase de projeto e de conceção.

- A estrutura e organização dos edifícios tem enorme importância, devendo então prever uma forma simples e fácil de organização do espaço.
- A casa residencial deverá adaptar a novas formas de utilização e a novas necessidades no futuro.
- A ideia de casa residencial inteligente não é exclusiva a escritório, podendo ser implementado em hospitais, escolas, hotéis, espaços comerciais, entre outros espaços.
- O grau de inteligência da habitação, esta associado a forma de como as necessidades e requisitos das organizações lá instaladas são satisfeitas.
- Na casa residencial não se deve dar apenas importância a aspetos com o controlo, automação e supervisão, mas também aos sistemas informáticos e às comunicações,
- A casa residencial deve proporcionar espaços que motivem o trabalho do utilizador, apoiando também tarefas criativas ou administrativas,
- Na casa residencial, deve ser permitido aos trabalhadores intervir no ambiente de trabalho de forma a que este satisfaça as suas necessidade e preferências.
- Diversos sistemas na casa residencial como automação, comunicação e processamento de informação, devem se relacionar e entreajudar entre si, possibilitando assim novos níveis de gestão e supervisão, levando a um melhor aproveitamento dos recursos acessíveis no edifício.

Já na década de 90 apareceu o conceito mais aprofundado de casa residencial inteligente, em que a incorporação de serviços desempenha uma função muito importante. O serviço de controlo de acessos interage com o serviço de apoio à portaria, isto é, obtém a informação e envia-a para o serviço de vigilância, e assim repetidamente.

Com o passar do tempo, o Homem tem apostando em novas tecnologias na sua habitação. Nomeadamente o aumento da segurança, tornar a habitação mais confortável para o utilizador e gestão de energia sustentável,

Com a evolução das habitações destacam-se principalmente as instalações técnicas, tornando-se cada vez mais complexas.

A arquitetura de uma casa inteligente nos dias de hoje encontra-se presente na Figura 1.

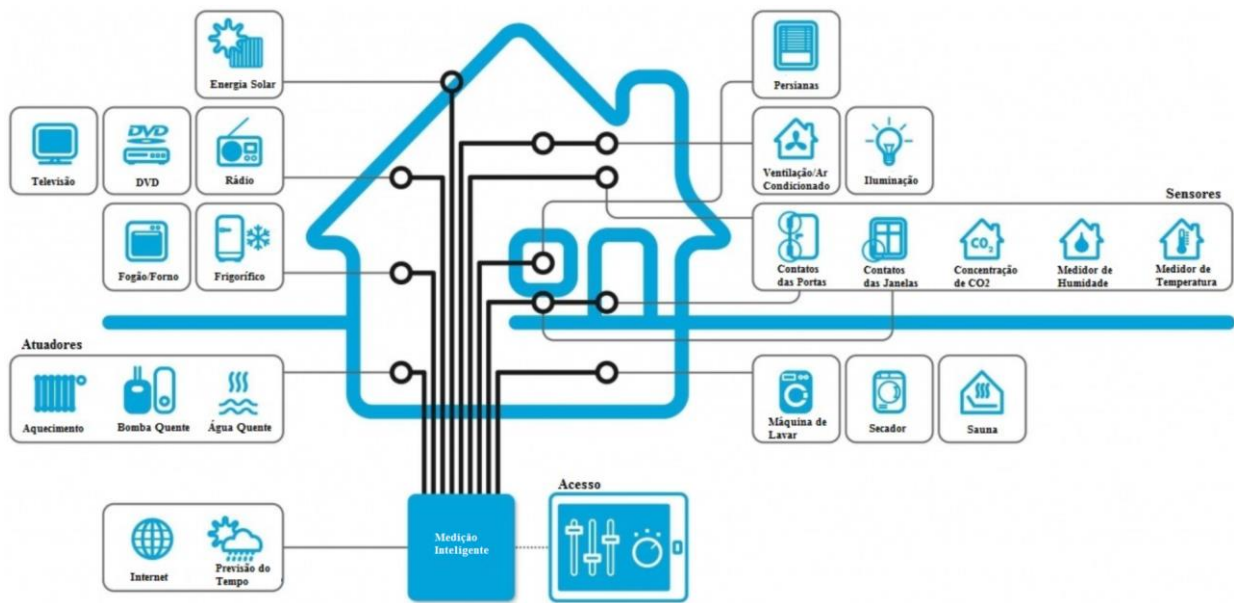


Figura 1 - Arquitetura de uma casa inteligente

2.1.1 Objetivos das casas residências inteligentes

De forma a responder as necessidades, agrupou-se os objetivos das casas inteligentes segundo a arquitetura, tecnologia, questões ambientais e económicos dos quais se realçam:

Arquitetónicos:

- Responder as necessidades do utilizador;
- A agilidade quer dos sistemas, como também das estruturas e dos serviços;
- Processo arquitetónico ajustado e certo;
- Aumento de estimulação no trabalho;
- A finalidade da habitação;
- Aumento da comodidade e proteção do utilizador;

Tecnológicos;

- Disponibilidade de meios técnicos avançados de telecomunicações;
- A automatização das instalações;
- Integração de serviços;

Ambientais:

- A criação de um edifício sustentável;

- A gestão energética;
- Preservar o meio ambiente;

Económicos:

- A redução dos custos da operação e manutenção das casas residenciais;
- Vantagens económicas para os clientes;
- Aumento da durabilidade das habitações;
- A possibilidade de cobrar preços mais altos pela renda ou venda de espaços;
- A relação custo/benefício.

2.1.2 Vantagens das habitações inteligentes

Conforme as casas residenciais tornam-se mais dispendiosas e complexas, como também crescem em número e sofisticação a nível de sistemas tecnologias que possuem, torna-se então cada vez mais importante gerir de forma mais capaz as habitações e a sua respetiva tecnologia.

Esta integração está interligada à capacidade dos diversos sistemas comunicarem entre si, trocarem informação e entreajuda para atingir objetivos em comuns.

Tendo em consideração os domínios tecnológicos mais essenciais das casas residenciais, sendo eles a automação, computação e comunicações, estes têm uma noção de integração que necessita de ser aplicada no interior de cada domínio e entre domínios diferentes. Isto é a integração deve ser o mais leve possível.

A melhor solução corresponde a sobrepor de forma total os vários domínios, de forma a que para o utilizador não seja possível distinguir os sistemas específicos de forma isolada ou independentes [3][4].

A noção de integração tem um papel fundamental no âmbito das casas inteligentes. Tal deve-se ao conjunto de vantagens e capacidades que estas podem oferecer, entre elas:

- Maior aproveitamento dos recursos disponíveis e uma maior qualidade na utilização;
- Novas utilidades, como mérito da interação e interligação entre os vários sistemas/aplicações;
- Reações organizadas e rápidas;

- Soluções com uma melhor relação funcionalidade/custo.
- A capacidade de relacionar informação, processar e de aprimorar decisões;
- Fácil acesso aos sistemas a partir do mesmo ponto, tornando-o mais simples, flexível e eficaz;
- Maior produtividade, facilidade na execução de tarefa complexas entre vários sistemas;

3 Estrutura do Trabalho

No projeto SIGEE existe um conjunto de tarefas a serem realizadas, desde o estudo de perfis de consumos até à criação das bases de dados. Este trabalho tem como objetivo o armazenamento de dados provenientes de todo o SIGEE. Assim sendo, nesta fase, a escolha de ferramentas e material a ser utilizado é o primeiro passo para a criação de um DataLogger.

3.1 Arduíno

Um trabalho como este que envolve eletrónica, analógica ou digital, e que por sua vez é necessário torná-lo inteligente/autómato, sabe-se logo à partida que é necessário a utilização de microcontroladores. Um microcontrolador possui uma memória interna e uma unidade de processamento, mas, porém, possui um desempenho inferior à de um processador. Hoje em dia os microcontroladores mais utilizados, em geral, são os PIC da Microchip e os Atmega da ATMEL.

Face ao problema, foi de rápida decisão a escolha do Arduíno, devido a ser uma plataforma que permite o desenvolvimento de controlo de sistemas, de baixo custo e que utiliza os microcontroladores da ATMEL (Atmega). [5]

Com o Arduíno também é possível enviar e receber informações de praticamente qualquer outro sistema eletrónico. Outra característica importante é que todo o material é open-source (hardware, software, bibliotecas) sendo a plataforma composta por hardware e software. Trata-se de uma placa PCB de programação de código aberto, baseado numa simples placa microcontrolador e um ambiente de desenvolvimento para escrever o código para a respetiva placa. A linguagem de programação utilizada no Arduíno é a linguagem C++ sendo uma das linguagens mais conhecidas. A arquitetura do Hardware do Arduíno encontra-se presente na Figura 2.

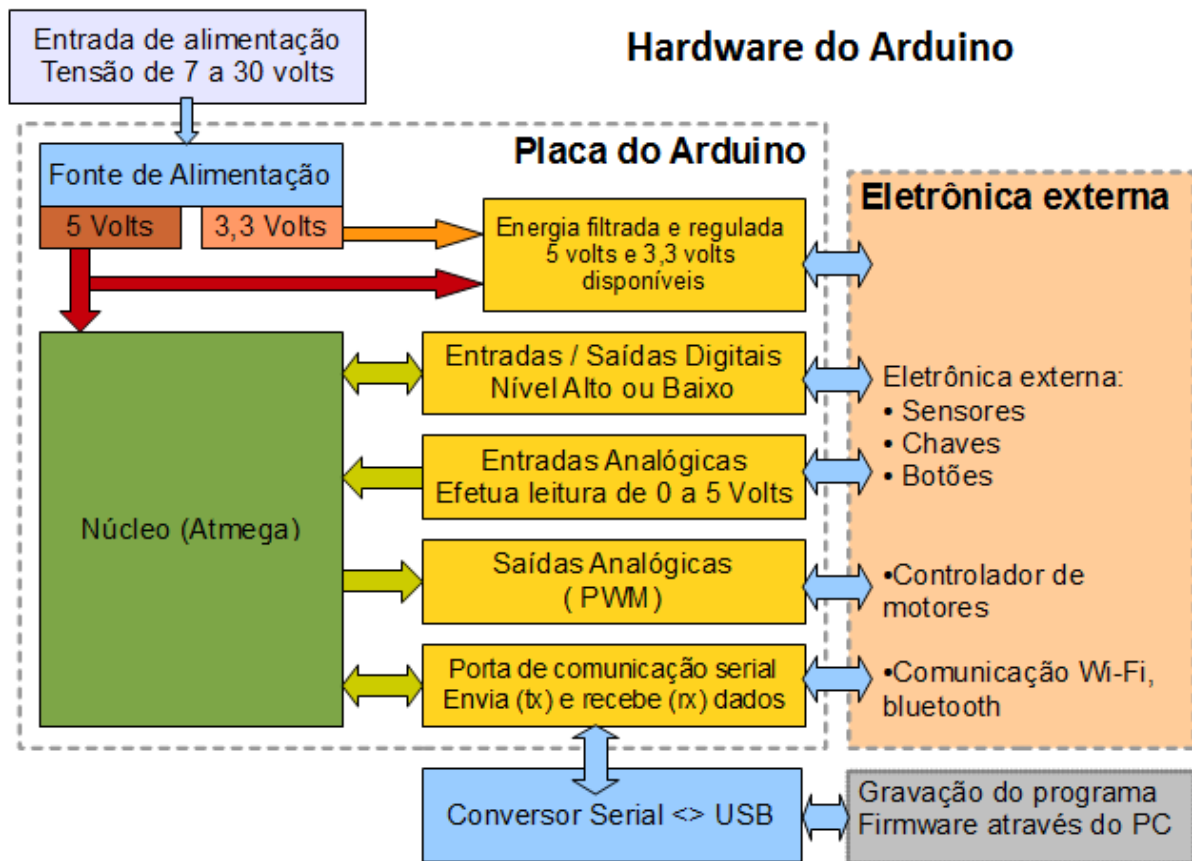


Figura 2 - Arquitetura do Hardware do Arduino [5]

Os vários tipos de Arduino têm uma constituição muito semelhante e são compostos por blocos:

- Fonte de Alimentação – Recebe energia externa, filtra e converte em duas tensões reguladas e filtradas (5 e 3,3 Volts).
- Núcleo CPU – Um microcontrolador responsável pelo processamento da informação.
- Entradas e Saídas – O CPU vem completo com diversos dispositivos embutidos dentro do chip.
- Pinos com funções especiais – Alguns pinos possuem hardware embutido para funções especiais.
- Firmware – Programa que é carregado para o CPU com o código de programação

Por isto, o Arduino tem tudo o que é necessário para a realização do trabalho, permitindo, caso seja necessário, o acoplamento de circuitos externos através de pinos de conexão em posições padronizadas.

3.2 Real Time Clock

Neste trabalho, desenvolvimento do Datalogger, é necessário armazenar a informação, mas não faz sentido armazenar as informações das ações provenientes do SIGEE sem dar a conhecer ao utilizador em que dia e hora foi feita essa mesma ação/armazenamento. Com isto optou-se por utilizar um RTC que basicamente é um relógio, com calendário, que é capaz de fornecer informações como: segundo, minuto, dia, mês e ano. O RTC tem um circuito que deteta falhas de energia acionando assim automaticamente a bateria sem se perder dados. As informações são transferidas via protocolo I²C. Tem um baixo consumo de energia, liberta o sistema de tarefas mais críticas e costuma ser mais preciso que outros métodos. [6]

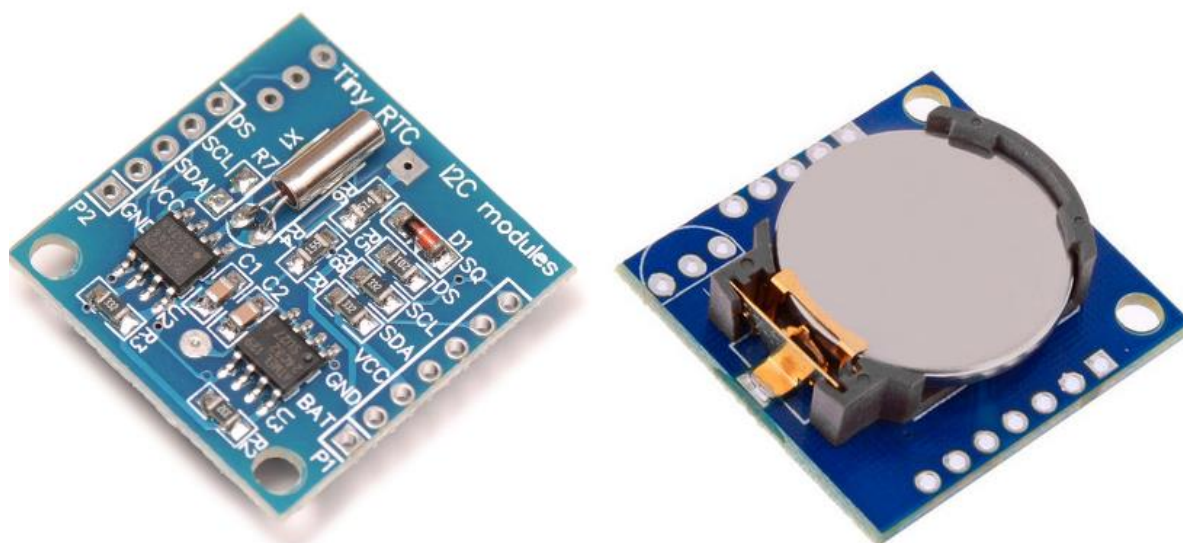


Figura 3 - Real Time Clock - DS1307 [6]

3.3 Conversor DC/DC bidirecional

Conectar um dispositivo de 3,3V a um sistema de 5V pode ser um desafio. O conversor DC/DC é um pequeno dispositivo que converte com segurança de 5V para 3.3V e vice-versa. Este conversor tem a capacidade de conversão até 4 pinos, tanto da tensão mais alta como da tensão mais baixa. Possui também duas entradas e duas saídas que servem para fornecer os 3,3 e 5V. Este conversor pode ser usado através de uma comunicação em série normal, SPI ou I²C.

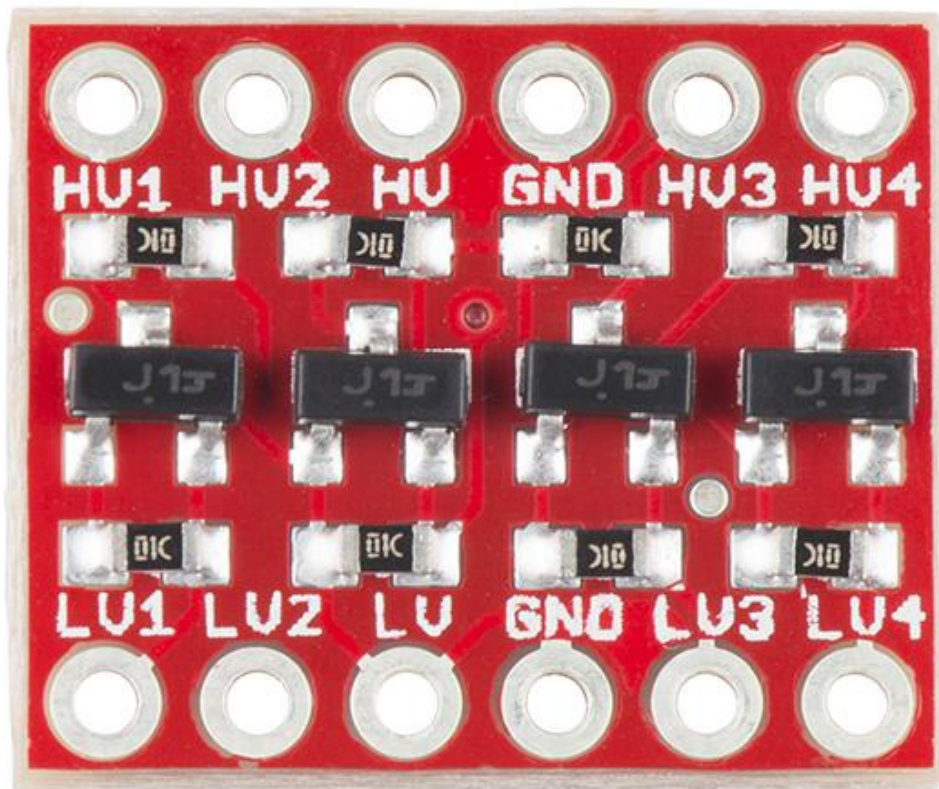


Figura 4 - Conversor DC/DC 3,3V e 5V

Repare-se que na Figura 4, a placa terá de ser alimentada a partir de duas fontes de tensão (alta tensão e baixa tensão). Por exemplo, a alta tensão referente aos 5V no pino HV e a baixa referente aos 3,3V, quanto os pinos GND são conectados à massa. Quanto aos pinos de conversão, estes estão identificados como entradas e saídas, sendo esta identificação relativa à placa, ou seja, um sinal digital que entre no pino HV1 com tensão de 5V vai ser convertido e surgirá no pino LV1 com a tensão de 3,3V. Caso se pretenda fazer o inverso, converter uma tensão de 3,3 para 5V, é necessário realizar o inverso.

Outra solução para este problema seria optar por utilizar um divisor de tensão, que faria o mesmo trabalho, contudo nem sempre fica tão robusto e confiável, além do espaço ocupado ser superior.

3.4 Cartão SD

Para o armazenamento de informação proveniente do projeto SIGEE, utilizou-se um cartão SD, que também é conhecido como cartão de memória flash. Este dispositivo de armazenamento de dados com memória flash é utilizado por todo o tipo de tecnologia em que seja necessário guardar dados/informações (telemóveis, câmaras digitais, consolas).

A memória flash é a tecnologia responsável pelo armazenamento e respetivo acesso aos dados, é um chip do tipo EEPROM, o que significa que nele a gravação e eliminação de dados é feita eletricamente sem a necessidade de uso de equipamentos especiais para a realização destas tarefas. Trata-se de uma tecnologia “não volátil”, ou seja, onde as informações ficam armazenadas por grandes períodos de tempo sem que seja necessário o uso de fontes de energia. [7]



Figura 5 - Micro SD e adaptador de tamanho SD original

Sendo o cartão SD facilmente encontrado em todo o mercado e com preços acessíveis à maior parte da população, faz com que o utilizador tenha livre escolha nos cartões quanto a sua capacidade ou qualidade, além de ser de fácil porte devido ao seu tamanho. Por estas razões foi escolhido para o armazenamento, como mostra a Figura 5. Contudo poderiam ser usados outros dispositivos como por exemplo, uma *pendrive* USB.

Para estabelecer uma comunicação entre o Arduino e o cartão SD, é necessário a utilização de um módulo Cartão SD Card, este módulo permite a leitura e a escrita no Cartão. O módulo é alimentado com tensões de 3,3 ou 5V, suporta formatos de arquivo FAT16 e FAT32. O FAT16 utiliza 16 bits para o endereçamento de dados, o que na prática, significa que o sistema de arquivos pode trabalhar até 65536 clusters, isto é, se cada cluster tiver até 32 kB de tamanho é capaz de trabalhar com discos ou partições até 2 GB. Quanto ao FAT32, é uma evolução do FAT16, utiliza 32 bits para cada cluster, permitindo apenas clusters de 4 kB. [8] Quanto a comunicação é realizada através do protocolo SPI através dos pinos MOSI, CSK, MISO, CS, o nível de tensão à entrada é de 3,3V.

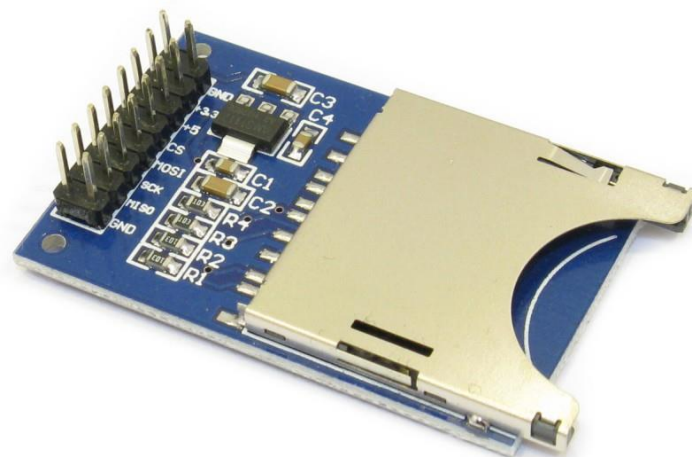


Figura 6 - Módulo SD Card

4 Conceção do DataLogger

Neste capítulo são descritos todos os passos necessários para a elaboração do DataLogger. A primeira parte será a implementação do circuito em Hardware, em seguida será feita uma análise aos pontos chave do software.

4.1 Implementação do circuito

Na implementação do circuito é necessário fazer a conexão entre o Arduíno e os outros componentes. Primeiramente estabelece-se conexão com o RTC, que por sua vez irá transmitir informações relativamente à hora, dia, mês e ano para o Arduíno. Como tem uma bateria própria o RTC mesmo desligado manterá a contagem das horas, isto para caso de existir uma falha de energia e/ou o Arduíno não seja alimentado por algum período de tempo, a contagem continuará a ser feita. Em seguida será efetuada a conexão com o Modulo de cartões SD, esta ligação tem como objetivo estabelecer uma ligação entre o Arduíno e o armazenamento de dados (Cartão SD), porém é necessário a utilização de um conversor dc/dc bidirecional devido aos pinos do módulo trabalharem com 3,3V e a tensão proveniente do Arduíno ser de 5V, contudo o módulo é alimentado pela tensão 5V. Por fim será implementado um LM35, este sensor tem como função fornecer informação ao utilizador da temperatura a que se encontra uma certa divisão da habitação.

A Figura 7 mostra uma melhor perceção do trabalho, quanto às conexões existentes no trabalho.

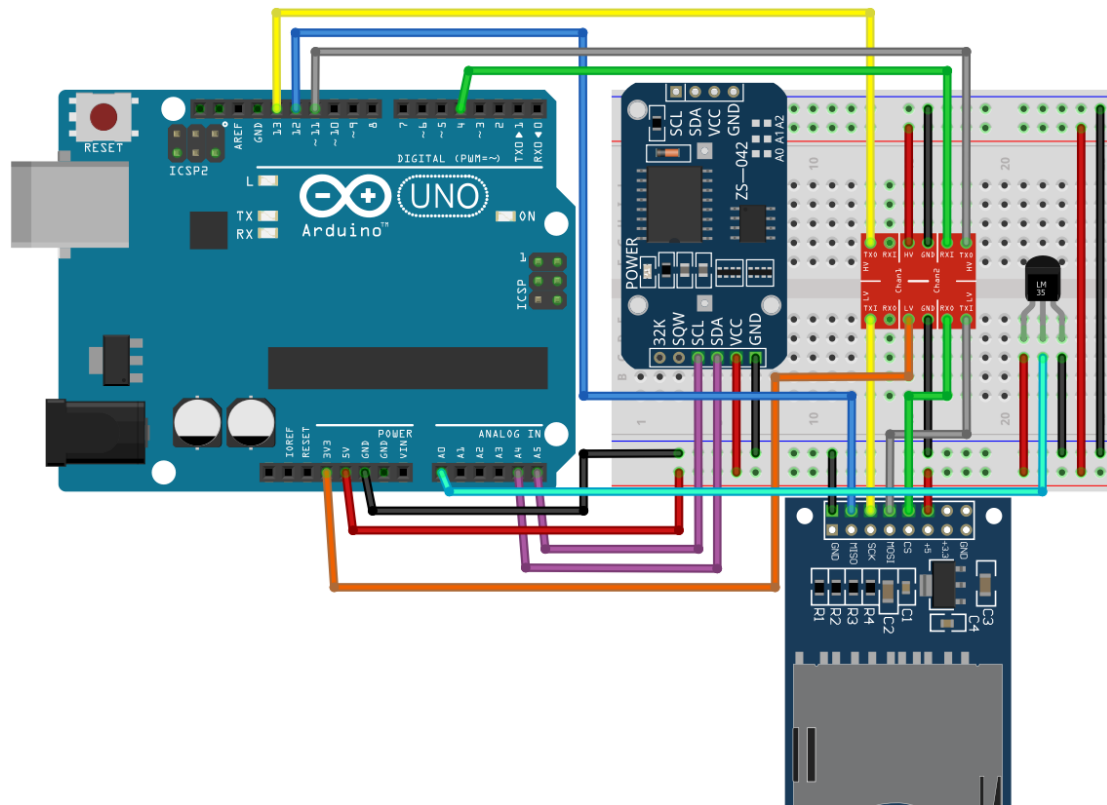


Figura 7 - Circuito DataLogger no Fritzing

Depois é necessário efetuar a montagem entre o Arduíno e o RTC, sendo esta ligação feita através dos pinos da interface I²C (A4 e A5), que estão ligados aos pinos SDA e SCL do RTC.

Em seguida será feita a ligação ao módulo do Cartão SD - este módulo é alimentado pela tensão de 5V, mas os pinos restantes trabalham com tensões de 3,3V. Por isso será utilizado um conversor DC/DC bidirecional, que irá reduzir a tensão de 5V proveniente do Arduíno para a tensão suportada do módulo, sendo esta ligação efetuada em duas partes. A primeira parte será alimentar a placa a partir de duas fontes de tensão. A alta tensão, proveniente do Arduíno (5V), para o pino HV, e a baixa tensão que também pode ser alimentada a partir do Arduíno (3,3V), para o pino LV, enquanto a massa será ligada aos pinos GND. A segunda será a ligação dos pinos do Arduíno ao conversor DC/DC bidirecional e em seguida do conversor ao módulo do Cartão SD. Os pinos digitais do Arduíno utilizados (13,12,11,4) irão transmitir informação para os pinos do módulo do cartão SD (SCK, MISO, MOSI, SDCS) em que esta comunicação é realizada através do protocolo SPI.

Também foi introduzido um sensor de temperatura, LM35, para registarmos a temperatura na habitação. Este sensor é constituído por 3 pinos - o primeiro é ligado aos 5V do circuito, o do meio ao Arduíno (A0) e o último à massa. Poderia ter sido usado o sensor de temperatura que geralmente vem com RTC, DS18B20, este componente deve ser conectado aos 3 pinos do

canto superior esquerdo do RTC, mas como neste caso o sensor de temperatura não veio juntamente com o RTC, foi utilizado o LM35.

Por fim será necessário a utilização de um cabo USB para efetuar ligação do Arduíno ao computador pessoal.

4.2 Explicação do código

Com o hardware implementado é necessário um software por detrás para que os dados sejam gravados no cartão SD. Neste ponto serão explicadas as partes com maior importância no programa do DataLogger. Contudo pode ser consultado o programa completo no Anexo A.

Neste programa são utilizadas as bibliotecas SPI, Wire e Sd, além destas são utilizadas as bibliotecas Time, Streaming e DS3232RTC que não vêm incluídas na IDE do Arduíno.

Quanto às bibliotecas, estas têm diferentes propósitos, a SPI é um protocolo de dados em série que sincroniza os microcontroladores para comunicar rapidamente com um ou mais dispositivos periféricos e também pode ser usado na comunicação entre dois microcontroladores.

A biblioteca Wire de permitir a comunicação através de I²C com outros dispositivos. Quanto à biblioteca SD, tem como função ler e escrever em cartões SD. A biblioteca Time tem como propósito adicionar a capacidade do Arduíno fazer a contagem do tempo com ou sem Hardware externo (RTC).

A Streaming funciona como um print normal, mas permitindo ao programador fazer um print em menor extensão (exemplo um print com oito linhas pode ser facilmente feito numa) sem consumir recursos.

Por fim a biblioteca DS3232RTC é necessária para comunicação com o RTC destinada a ser usada com a Time, além disso, esta é compatível com o RTC DS1307.

Portanto os pontos mais importantes no programa são:

A Sincronização entre a biblioteca Time e o RTC, para obtenção do tempo caso o Arduíno fique sem energia.

```
setSyncProvider(RTC.get);  
Serial << F("Sincronizando com o RTC...");  
if (timeStatus () != timeSet) Serial << F(" Falha!");  
Serial << endl;
```

A fórmula da temperatura utilizada no LM35 para ser fornecida a temperatura ao utilizador.

```
cel = ((4.4 * analogRead(tempPin) * 110) / 1024.0);
```

O sensor será alimentado por uma tensão de 5V, como as entradas analógicas do Arduino tem uma resolução de 10 bits, ou seja, $2^{10}=1024$ e como cada grau corresponde a 10 mV então a expressão de temperatura em função do valor lido na entrada analógica, será a equação acima referida. O `analogRead(tempPin)` será necessário para ler essa tensão enviada por parte do LM35.

Para verificação se um horário novo foi introduzido, é feito um teste na *serial port* para que sejam postos os dígitos corretos relativamente ao ano e para que o programador introduza o horário correto pela primeira vez. O formato da introdução é ano, mês, dia, hora, minuto, segundo.

```
if (Serial.available() >= 12) {
  int y = Serial.parseInt();
  if (y >= 100 && y < 1000)
    Serial<<F ("Erro: Ano deve ter dois ou quatro digitos!") <<endl;
  ...
  Serial << endl;
  while (Serial.available() > 0) Serial.read();
```

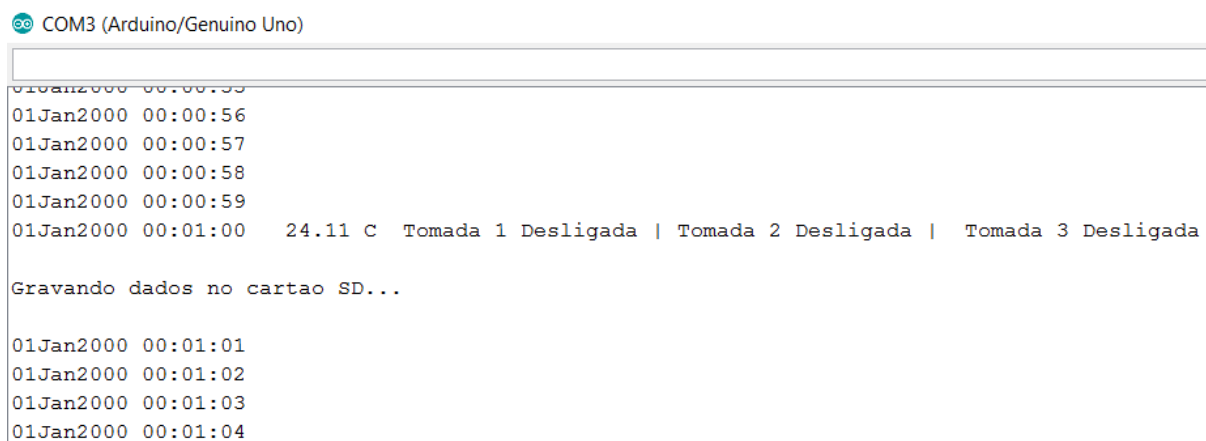
Na gravação dos dados no cartão SD os dados são gravados de minuto em minuto

```
void grava_SD (time_t, t)
{
  abre_arquivo_gravacao("data.txt");
  file.print("Data: ");
  file.print(day(t));
  ...
  fecha_arquivo();
}
```

5 Simulação e Resultados obtidos

5.1 Resultados Obtidos

Com o hardware e software preparados é altura de testar o Datalogger. Para isso, irá ser realizado o upload para o Arduíno e em seguida será executado o *serial port* do Arduíno. O *serial port* é mais destinado ao programador para que na primeira utilização seja definida a data e hora surgindo assim com o formato apresentado na Figura 8. O *serial port* serve também para que o programador possa comparar os dados que são armazenados no Cartão SD com os que são visualizados no *serial port*.



```
COM3 (Arduino/Genuino Uno)
01Jan2000 00:00:55
01Jan2000 00:00:56
01Jan2000 00:00:57
01Jan2000 00:00:58
01Jan2000 00:00:59
01Jan2000 00:01:00 24.11 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Desligada
Gravando dados no cartao SD...
01Jan2000 00:01:01
01Jan2000 00:01:02
01Jan2000 00:01:03
01Jan2000 00:01:04
```

Figura 8 - Primeira vez no *serial port*

Como se repara na Figura 8, no *serial port* o programador pode visualizar a data, hora, temperatura e o estado das tomadas (ON/OFF). Contudo, na Figura 8 a data e hora encontra-se com valores padrão por ser a primeira utilização e por isso deverão ser definidas, como se vê na Figura 10.

COM3 (Arduino/Genuino Uno)

```
2016,09,25,15,07,30|
01Jan2000 00:02:09
01Jan2000 00:02:10
Horario modificado para: 25Sep2016 15:07:30
25Sep2016 15:07:30
25Sep2016 15:07:31
25Sep2016 15:07:32
25Sep2016 15:07:33
25Sep2016 15:07:34
```

Figura 9 - Modificação da data e hora no serial port

Para efetuar a modificação de horas deverá ser introduzido no comando de texto do *serial port* o ano, mês, dia, hora, minuto e segundos, como demonstra a Figura 9.

COM3 (Arduino/Genuino Uno)

```
25Sep2016 15:07:34
25Sep2016 15:07:55
25Sep2016 15:07:56
25Sep2016 15:07:57
25Sep2016 15:07:58
25Sep2016 15:07:59
25Sep2016 15:08:00 24.11 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Desligada

Gravando dados no cartao SD...

25Sep2016 15:08:01
25Sep2016 15:08:02
25Sep2016 15:08:03
```

Figura 10 - Após modificação da data e hora

Após modificação da data e hora, o DataLogger encontra-se pronto para ser utilizado sem que seja necessário mais a intervenção do programador, além disso o *serial port* não precisa de ser executado mais nenhuma vez, apenas se o programador necessitar de fazer simulações. Se for necessário visualizar em tempo real o que está a acontecer no sistema, em vez de estar a retirar o cartão do Datalogger, é introduzir num computador pessoal. Vê-se que na Figura 10, a data e a hora já é a correta, além disso consegue se visualizar quando os dados são gravados no Cartão SD.

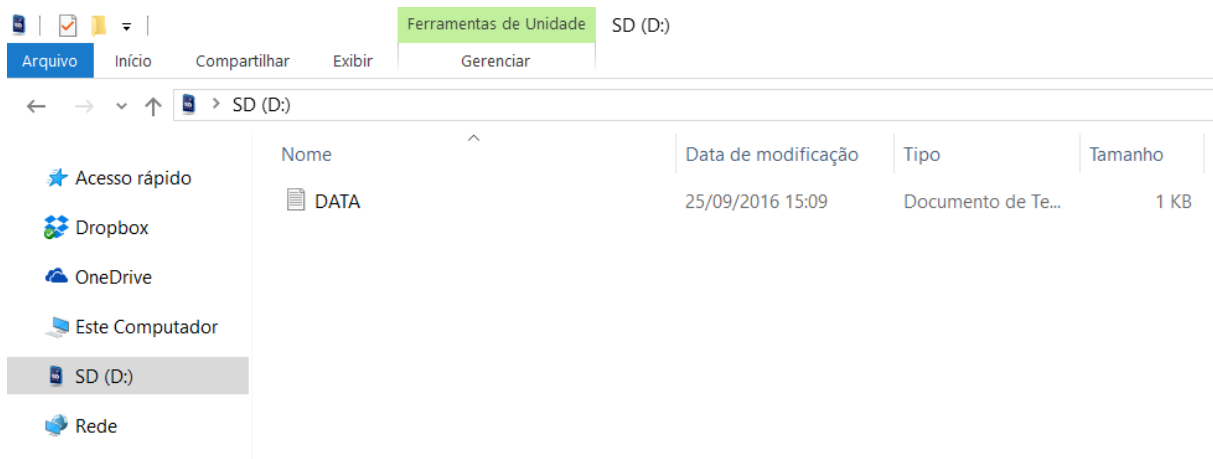


Figura 11 - Ficheiro de texto criado no Cartão SD

Sempre que o utilizador quiser verificar os dados armazenados basta introduzir o Cartão SD num computador pessoal ou num outro dispositivo que proporcione a visualização de ficheiros tipo texto presentes no Cartão. A Figura 11 demonstra ficheiro criado no Cartão SD.

DATA - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

Data: 01/01/2000	Hora: 00:00:00	Temperatura: 25.05°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 01/01/2000	Hora: 00:01:00	Temperatura: 24.11°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:08:00	Temperatura: 24.11°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:09:00	Temperatura: 24.58°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada

Figura 12 - Informação no Ficheiro de texto

Na Figura 12 encontra-se a informação deste pequeno caso, vê-se que foi feita a alteração da data e hora, a temperatura manteve-se estagnada e nenhuma tomada foi acionada durante este período de tempo, estas tomadas podem estar a comunicar com qualquer eletrodoméstico, aquecedor, micro-ondas, televisão, entre outros.

5.2 Simulação

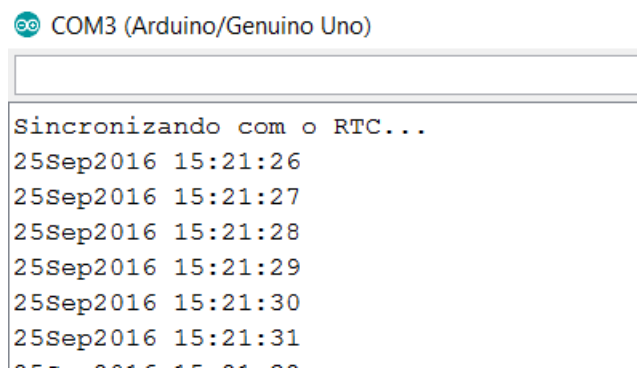
Neste capítulo serão realizadas duas simulações, uma vez que os testes estão limitados ao estado de desenvolvimento do projeto do SIGEE estando outras fases do projeto ainda em fase de desenvolvimento e não completamente terminadas. A primeira simulação realizada consistiu no acionamento de uma tomada, que comunica com um micro-ondas. A segunda simulação consistiu num acionamento de uma outra tomada que comunica com um aquecedor. Nesta poderá ver-se a temperatura a subir ligeiramente e quando esta se desliga a temperatura desce e que acaba por estagnar. Também foi ligada uma outra tomada no que poderia ser um outro dispositivo, por exemplo uma televisão. Estas simulações serão visualizadas como um programador ou utilizador. O principal objetivo destas simulações será verificar se armazenamento de informação está a ser feito corretamente.

5.2.1 Simulação Micro-ondas

Na simulação do micro-ondas será visto primeiramente no ponto do programador e em seguida no do utilizador.

No ponto de vista do programador é necessário abrir no *serial port*, liga-se o Arduino ao computador e em seguida abre-se o *serial port* para ver a informação em tempo real.

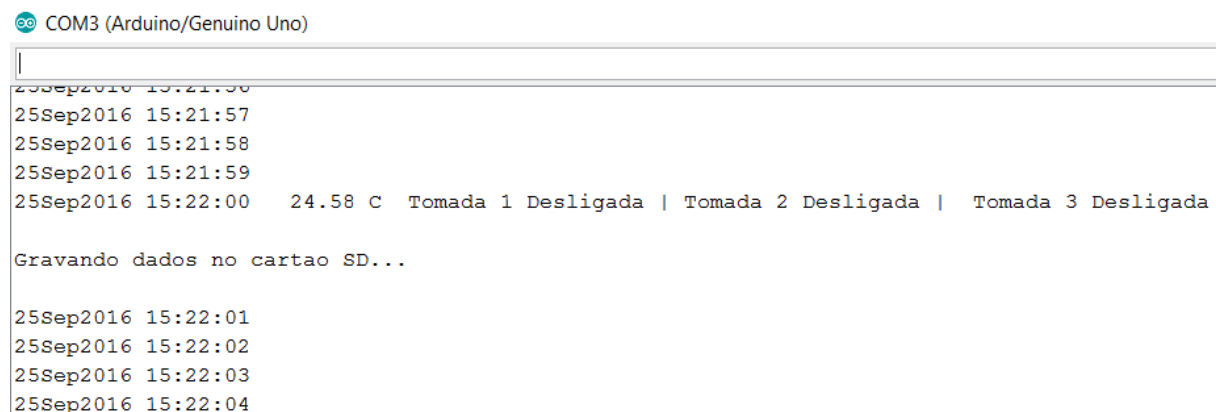
Com abertura de série vê-se logo que o Arduino sincroniza com RTC, pois como não esteve alimentado perdeu a “noção” da data e hora e por isso a sincronização com o RTC como demonstra a Figura 13.



```
COM3 (Arduino/Genuino Uno)
Sincronizando com o RTC...
25Sep2016 15:21:26
25Sep2016 15:21:27
25Sep2016 15:21:28
25Sep2016 15:21:29
25Sep2016 15:21:30
25Sep2016 15:21:31
```

Figura 13 - Sincronização com o RTC

Após sincronização, o Arduino irá começar a ser realizada uma contagem do tempo, gravando de minuto em minuto no cartão SD as informações relativamente às tomadas, como a temperatura de uma divisão ou da própria habitação. Como se vê na Figura 14, todas as tomadas encontram-se desligadas e é feita a gravação dessa informação no cartão SD.



```
COM3 (Arduino/Genuino Uno)
25Sep2016 15:21:56
25Sep2016 15:21:57
25Sep2016 15:21:58
25Sep2016 15:21:59
25Sep2016 15:22:00 24.58 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Desligada
Gravando dados no cartao SD...
25Sep2016 15:22:01
25Sep2016 15:22:02
25Sep2016 15:22:03
25Sep2016 15:22:04
```

Figura 14 - Informação de tomadas desligadas e respetiva gravação

Na Figura 15 visualiza-se que a tomada um, onde está ligado o micro-ondas, foi acionada, logo o micro-ondas entrará em funcionamento até que seja enviado um comando para que esta seja desligada, sendo esta mudança armazenada no Cartão SD.

```

COM3 (Arduino/Genuino Uno)
25Sep2016 15:22:55
25Sep2016 15:22:56
25Sep2016 15:22:57
25Sep2016 15:22:58
25Sep2016 15:22:59
25Sep2016 15:23:00  24.11 C Tomada 1 Ligada | Tomada 2 Desligada | Tomada 3 Desligada

Gravando dados no cartao SD...

25Sep2016 15:23:01
25Sep2016 15:23:02
25Sep2016 15:23:03
25Sep2016 15:23:04

```

Figura 15 - Tomada 1 (Micro-ondas) ligada

Após três minutos a tomada foi desligada de imediato e o micro-ondas também foi desligado. E mais uma vez é registada esta informação no Cartão SD, como visto na Figura 16.

```

COM3 (Arduino/Genuino Uno)
25Sep2016 15:25:57
25Sep2016 15:25:58
25Sep2016 15:25:59
25Sep2016 15:26:00  24.11 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Desligada

Gravando dados no cartao SD...

25Sep2016 15:26:01
25Sep2016 15:26:02
25Sep2016 15:26:03

```

Figura 166 - Tomada 1 Desligada (Micro-ondas desligado)

Após ser visualizado no ponto de vista do programador e ao que parece tudo bem, está altura de ser visto no ponto de vista do utilizador e verificar se os valores correspondem. No ponto de vista do utilizador, este aciona a tomada e o micro-ondas liga, quando este achar necessário desliga-a e o micro-ondas desligar-se-á. Se o utilizador de encontrar longe da cozinha no momento em que aciona a tomada, e para ter a certeza que a tomada foi ligada, pode verificar o Cartão SD e encontra toda a informação desta ação, como se pode visualizar na Figura 17. Caso a tomada não tivesse sido acionada, apareceria a tomada desligada em vez de ligada durante este período de tempo.

DATA - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

Data: 25/09/2016	Hora:15:22:00	Temperatura: 24.58°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora:15:23:00	Temperatura: 24.11°C	Tomada 1 Ligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora:15:24:00	Temperatura: 24.11°C	Tomada 1 Ligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora:15:25:00	Temperatura: 24.58°C	Tomada 1 Ligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora:15:26:00	Temperatura: 24.11°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada

Figura 177 - Informação no ficheiro de texto

5.2.2 Simulação Aquecedor

A segunda simulação irá ser realizada, tal como na primeira, com visualização como programador e de seguida como um utilizador comum. Para isso, o Arduino será conectado ao computador e será executado o *serial port*.

No início é feita na sincronização com o RTC, como na primeira simulação.

Todas as tomadas encontram-se desligadas e é efetuada a gravação no Cartão SD, como se repara na Figura 18.

```
COM3 (Arduino/Genuino Uno)
25Sep2016 15:32:55
25Sep2016 15:32:56
25Sep2016 15:32:57
25Sep2016 15:32:58
25Sep2016 15:32:59
25Sep2016 15:33:00  25.05 C  Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Desligada
Gravando dados no cartao SD...
25Sep2016 15:33:01
25Sep2016 15:33:02
25Sep2016 15:33:03
```

Figura 188 - Após sincronização com RTC, tomadas desligadas

Entretanto a tomada 2 é ligada, como se vê na Figura 19, esta tomada comunica com um aquecedor, sendo responsável pela sua alimentação.

```
COM3 (Arduino/Genuino Uno)
25Sep2016 15:33:57
25Sep2016 15:33:58
25Sep2016 15:33:59
25Sep2016 15:34:00  24.58 C  Tomada 1 Desligada | Tomada 2 Ligada | Tomada 3 Desligada |
Gravando dados no cartao SD...
25Sep2016 15:34:01
25Sep2016 15:34:02
25Sep2016 15:34:03
```

Figura 199 - Tomada 2 (Aquecedor) ligada

Após o aquecedor ser ligado, a temperatura irá aumentar por consequência desta ação. A temperatura na simulação tem um aumento muito significativo, como se vê na Figura 20, como seria de esperar.

```

25Sep2016 15:35:56
25Sep2016 15:35:57
25Sep2016 15:35:58
25Sep2016 15:35:59
25Sep2016 15:36:00 31.20 C Tomada 1 Desligada | Tomada 2 Ligada | Tomada 3 Desligada

Gravando dados no cartao SD...

25Sep2016 15:36:01
25Sep2016 15:36:02
25Sep2016 15:36:03

```

Figura 200 - Subida da temperatura, com tomada 2 ligada

Em seguida a tomada dois é desligada e por consequência a temperatura irá descer e estagnar passado um período de tempo, entretanto uma tomada 3 é ligada, como se vê na Figura 21.

```

25Sep2016 15:36:58
25Sep2016 15:36:59
25Sep2016 15:37:00 27.41 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Ligada

Gravando dados no cartao SD...

25Sep2016 15:37:01
25Sep2016 15:37:02
25Sep2016 15:37:03
25Sep2016 15:37:04

```

Figura 211 - Descida da temperatura, com a tomada 2 desligada e tomada 3 (TV) ligada

Repara-se na Figura 22, a tomada 2 já se encontra desliga há cerca de 2 minutos e a temperatura acabou por se estagnar.

```

25Sep2016 15:38:56
25Sep2016 15:38:57
25Sep2016 15:38:58
25Sep2016 15:38:59
25Sep2016 15:39:00 24.58 C Tomada 1 Desligada | Tomada 2 Desligada | Tomada 3 Ligada |

Gravando dados no cartao SD...

25Sep2016 15:39:01
25Sep2016 15:39:02

```

Figura 222 - Temperatura estagnada

Passando para a vista do utilizador, numa situação, por exemplo, em que este saiu do trabalho num dia de temperaturas baixas e ligou o aquecedor de casa para quando chegar a casa esta se encontrar mais confortável desligando-se a tomada dois caso a temperatura da casa se encontre demasiado quente. Caso o utilizador tomou conhecimento que a ação foi realizada enquanto este se encontrava fora de casa, todas as informações são armazenadas no Cartão SD, para que os utilizadores tomem conhecimento de todas as ações que foram realizadas

durante o dia. Com isto temos na Figura 23 os dados armazenados no Cartão SD desta simulação.

DATA - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

Data: 25/09/2016	Hora: 15:33:00	Temperatura: 25.05°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:34:00	Temperatura: 24.58°C	Tomada 1 Desligada	Tomada 2 Ligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:35:00	Temperatura: 28.83°C	Tomada 1 Desligada	Tomada 2 Ligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:36:00	Temperatura: 31.20°C	Tomada 1 Desligada	Tomada 2 Ligada	Tomada 3 Desligada
Data: 25/09/2016	Hora: 15:37:00	Temperatura: 27.41°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Ligada
Data: 25/09/2016	Hora: 15:38:00	Temperatura: 26.47°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Ligada
Data: 25/09/2016	Hora: 15:39:00	Temperatura: 24.58°C	Tomada 1 Desligada	Tomada 2 Desligada	Tomada 3 Ligada

Figura 233 - Informação da simulação no ficheiro de texto no Cartão SD

6 Conclusão e desenvolvimentos futuros

6.1 Conclusão do trabalho

O sistema de gestão de energia elétrica numa habitação tem um papel fundamental na gestão geral do sistema elétrico no paradigma das Casas Inteligentes. Este traz grandes benefícios quer para o consumidor doméstico quer para a operadora elétrica, através da gestão sustentável de energia.

Este projeto focou-se na conceção, desenvolvimento e validação de metodologias de gestão inteligente de energia a serem instituídas em consumidores domésticos.

Os sistemas de gestão de energia utilizados nas habitações são cada vez mais sofisticados considerando diferentes recursos de energia, levando a necessidade de desenvolvimento de metodologias capazes de juntar todos os recursos de modo a resultar uma gestão de energia mais eficaz e eficiente.

O trabalho desta dissertação incide na implementação de um DataLogger de filosofia simples, com tamanho reduzido e que permita uma elevada capacidade de armazenamento de dados provenientes do projeto SIGEE, com a possibilidade de ser ligado a vários dispositivos. O DataLogger possui a capacidade de ligação a um Cartão SD, proporcionando uma elevada capacidade de armazenamento, com muita flexibilidade na transferência de dados. Devido ao sistema de ficheiros do Cartão SD e do próprio formato dos ficheiros, estes podem ser diretamente reconhecidos em qualquer sistema operativo, sendo apenas necessário um leitor de cartões de memória que nos dias de hoje já se encontram implementado nos computadores pessoais, tornando assim a sua leitura simples. O DataLogger é de dimensões reduzidas, devido aos componentes utilizados e consome pouca energia. Visto que neste projeto, SIGEE, existir pelo menos um Arduíno para cada função neste projeto o DataLogger foi projetado para que seja alimentado com a tensão de 5V.

Portanto, o DataLogger desenvolvido preenche os principais requisitos propostos, sendo um dispositivo muito versátil para o armazenamento de dados de outros dispositivos.

6.2 Trabalhos futuros

Quanto a integração do DataLogger no projeto SIGEE, não foi possível devido a arquitetura baseada em Arduínos não se encontrar completa. Esta arquitetura consistiria numa comunicação PC e WI-FI, sem esta comunicação o DataLogger não consegue comunicar com o restante sistema. Contudo se esta comunicação estivesse previamente estabelecida, a integração com o restante sistema não iria oferecer grande dificuldade. A nível de hardware seria somente realizar umas ligações para comunicar com o restante sistema. Quanto ao software seria necessário alterar o programa para que este comunica-se com os restantes Arduínos, mas contudo esta alterações seriam muito pequenas, porque quem estaria responsável por essa parte iria fornecer os respetivos dados para esta integração fosse possível.

Mesmo assim ao longo do desenvolvimento deste trabalho foram surgindo ideia, entre as quais:

- Passar de protótipo do DataLogger para o formato de produto final fazendo uma implementação em placa de circuito impresso (PCB) sendo inclusivamente colocado dentro de uma caixa apropriada.
- Tal como exemplificado nas simulações o programador pode aceder ao monitor de série, para que visualize em tempo real o que o DataLogger está a armazenar no Cartão SD, com a implementação do LCD também seria possível ao utilizador visualizar.

7 Referencias bibliográficas

1. Eficiencia energética em edifícios, realidades empresariais e oportunidades, relatório síntese
2. Auriza Lopes de Barros, Edifícios Inteligentes e a Domotica, Universidade Jean Piaget de Cabo Verde
3. Renato Nunes, Carlos Serrô, Edifícios inteligentes: Conceitos e serviços, DEEC, IST/INESC
4. Paulo Marin, Edifício Inteligente - Conceito e Componentes
5. Curso de Arduino, Aula 2 - O hardware do arduino, Daniel O. Basconcelho Filho, robotizando.com.br
6. Ala-Paavola, Jaakko, software interrupt based real time clock source code project for PIC microcontroller
7. Sérgio Silva Pereira, O que são os cartões de memória SD?, officinadanet.com.br
8. Carlos E. Morimoto, Hardware Manual Completo, O que é um sistema de arquivos

Anexo A

A Código de Programação

```
#include <DS3232RTC.h>
#include <Streaming.h>
#include <Time.h>
#include <Wire.h>
#include <SD.h>
#include <SPI.h>

int CS_PIN = 4;
int pinTomada1 = 2;
int pinTomada2 = 7;
int pinIlu = 8;
float temp;
int tempPin = 0;
File file;
float cel;

void setup(void)
{
  Serial.begin(9600);
  //Inicializa o cartao SD
  inicia_SD();

  setSyncProvider(RTC.get);
  Serial << F("Sincronizando com o RTC...");
  if (timeStatus() != timeSet) Serial << F(" Falha!");
  Serial << endl;
}

void loop(void)
{
  static time_t tLast;
  time_t t;
  tmElements_t tm;

  //TEMPERATURA FORMULA

  cel =(( 4.4 * analogRead(tempPin) * 110) / 1024.0);
```

```

if (Serial.available() >= 12) {
  int y = Serial.parseInt();
  if (y >= 100 && y < 1000)
    Serial<<F("Erro: Ano deve ter dois ou quatro digitos!") <<endl;
  else {
    if (y >= 1000)
      tm.Year = CalendarYrToTm(y);
    else //(y < 100)
      tm.Year = y2kYearToTm(y);
    tm.Month = Serial.parseInt();
    tm.Day = Serial.parseInt();
    tm.Hour = Serial.parseInt();
    tm.Minute = Serial.parseInt();
    tm.Second = Serial.parseInt();
    t = makeTime(tm);
    RTC.set(t);
    setTime(t);
    Serial << F("Horario modificado para: ");
    printDateTime(t);
    Serial << endl;
    while (Serial.available() > 0) Serial.read();
  }
}

```

```

t = now();
if (t != tLast) {
  tLast = t;
  printDateTime(t);

  if (second(t) == 0)
  {
    Serial << F(" ") << cel << F(" C ");
    if (digitalRead(pinTomada1) == 1)
    {
      Serial.print("Tomada 1 Ligada | ");
    }else
    {
      Serial.print("Tomada 1 Desligada | ");
    }

    if (digitalRead(pinTomada2) == 1)
    {
      Serial.print("Tomada 2 Ligada | ");
    }else
    {
      Serial.print("Tomada 2 Desligada | ");
    }

    if (digitalRead(pinIlu) == 1)
    {
      Serial.println(" Tomada 3 Ligada | ");
    }else
    {
      Serial.println(" Tomada 3 Desligada | ");
    }
  }
  Serial.println("\nGravando dados no cartao SD...");
  grava_SD(t);
}

```

```

    }
    Serial << endl;
}
}

void printDateTime(time_t t)
{
    printI00(day(t), 0);
    Serial << monthShortStr(month(t)) << _DEC(year(t));
    Serial << ' ';
    //printTime(t);
    printI00(hour(t), ':');
    printI00(minute(t), ':');
    printI00(second(t), ' ');
}

void grava_SD(time_t t)
{
    abre_arquivo_gravacao("data.txt");
    file.print("Data: ");
    file.print(day(t));
    file.print("/");
    if (month(t) < 10)
    {
        file.print("0");
    }
    file.print(month(t));
    file.print("/");
    file.print(year(t));
    file.print(" | ");
    file.print("Hora: ");
    if (hour(t) < 10)
    {
        file.print("0");
    }
    file.print(hour(t));
    file.print(":");
    if (minute(t) < 10)
    {
        file.print("0");
    }
    file.print(minute(t));
    file.print(":");
    if (second(t) < 10)
    {
        file.print("0");
    }
    file.print(second(t));
    file.print(" | ");
    file.print("Temperatura: ");
    file.print(temperatura);
    file.print("°C | ");
    if(digitalRead(pinTomada1)==1)
    {
        file.print(" Tomada 1 Ligada | ");
    }else
    {
        file.print(" Tomada 1 Desligada | ");
    }
}

```

```

if(digitalRead(pinTomada2)==1)
{
  file.print(" Tomada 2 Ligada | ");
}else
{
  file.print(" Tomada 2 Desligada | ");
}
if(digitalRead(pinllu)==1)
{
  file.println(" Tomada 3 Ligada | ");
}else
{
  file.println(" Tomada 3 Desligada | ");
}
fecha_arquivo();
}

void printI00(int val, char delim)
{
  if (val < 10) Serial << '0';
  Serial << _DEC(val);
  if (delim > 0) Serial << delim;
  return;
}

void inicia_SD()
{
  pinMode(CS_PIN, OUTPUT);

  if (SD.begin())
  {
  } else
  {
    return;
  }
}

int abre_arquivo_gravacao(char filename[])
{
  file = SD.open(filename, FILE_WRITE);

  if (file)
  {
    return 1;
  } else
  {
    return 0;
  }
}

void fecha_arquivo()
{
  if (file)
  {
    file.close();
  }
}

```