

Daniel Honda

Rafael Toma

Automação Residencial: Integrando Ambientes para Conforto e Segurança

Monografia apresentada à Escola Politécnica da Universidade de São Paulo como requisito para obtenção do Título de Engenheiro Mecatrônico.

Orientador:

Prof. Dr. Thiago de Castro
Martins

FICHA CATALOGRÁFICA

Honda, Daniel Shigueo

**Automação residencial: integrando ambientes para conforto e segurança / D.S. Honda, R.H. Toma. – São Paulo, 2013.
56 p.**

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.Edifícios residenciais (Automação) 2.Sistemas de controle 3.Automação predial I.Toma, Rafael Hideki II.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos III.t.

Sumário

Lista de ilustrações

Lista de tabelas

Resumo

Abstract

1	Introdução	7
2	Estado da Arte - Automação Residencial	8
3	Requisitos	11
3.1	Requisitos Funcionais	11
3.2	Requisitos Não Funcionais	12
4	Síntese de Soluções	13
4.1	Arquitetura de Supervisão e Controle	13
4.1.1	Supervisão e Controle Centralizados	13
4.1.2	Supervisão Descentralizada e Controle Centralizado	13
4.1.3	Supervisão e Controle Descentralizada	15
4.1.4	Supervisão e Controle Híbridos	15
4.2	Central	16
4.2.1	Central Local	16
4.2.2	Central na Nuvem	16
4.3	Interface de Controle do Usuário	16
4.3.1	Interfaces Gráficas	17

4.3.2	Controle por Voz	20
4.3.3	Controle por Gestos	21
4.4	Segurança	22
4.4.1	AES	22
4.5	Rede Doméstica	24
4.5.1	Rede de Dados Baseada no Sistema Telefônico	24
4.5.2	Rede de Dados Baseada no Sistema de Distribuição Elétrica	25
4.5.3	Rede de Dados Sem Fio	26
5	Solução Desenvolvida	34
5.1	Arquitetura de Rede	34
5.2	Arquitetura de Controle	35
5.3	Central Local	36
5.4	Central Nuvem	38
5.5	Interface	39
5.6	Rede Local	40
5.7	Módulos Desenvolvidos	40
5.7.1	Módulo de umidade e temperatura	40
5.7.2	Módulo de segurança	42
5.7.3	Módulo de banheira	43
5.7.4	Módulo de iluminação	44
5.7.5	Módulo de janela e chuva	44
5.7.6	Módulo de controle remoto	45
5.8	Sequência de Funcionamento	46
5.8.1	Usuário Local Ativa Função de Dispositivo	46
5.8.2	Usuário Remoto Ativa Função de Dispositivo	46
6	Testes Realizados	48
6.1	Modulo Iluminação	48

6.1.1	Sensor de Luminosidade	48
6.1.2	Dimerização	48
6.2	Tempos de Carregamento da Interface	49
7	Conclusões	52
	Referências	53

Lista de ilustrações

4.1	Arquitetura Centralizada(Reproduzido de ^[9])	14
4.2	Arquitetura Descentralizada(Reproduzido de ^[5])	14
4.3	Exemplo de Interface utilizando a planta-baixa (Reproduzido de ^[8])	17
4.4	Exemplo de Interface 3D (Reproduzido de ^[2])	18
4.5	Exemplo de Interface com Blocos de Programação (Reproduzido de ^[3])	19
4.6	Exemplo de com Divisão Modular	20
4.7	Fluxograma Simplificado da API de Reconhecimento de Voz (Reproduzido de ^[4])	21
4.8	S-BOX (Retirado de ^[11])	22
4.9	Estágio SubByte (Retirado de ^[11])	23
4.10	Estágio ShiftRows (Retirado de ^[11])	23
4.11	Estágio MixColumns (Retirado de ^[11])	23
4.12	Estágio AddRoundKey (Retirado de ^[11])	24
4.13	Frame de Comunicação X-10 (Reproduzido de ^[12])	25
4.14	Tipos Tranceivers em uma Rede Zigbee (Reproduzido de ^[14]) . .	28
4.15	Topologias Zigbee (Reproduzido de ^[14])	28
4.16	Gráfico Comparativo de Tecnologias Sem-fio (Adaptado de ^[14]) .	29
4.17	Auxiliary Security Header (Retirado de ^[15])	32
4.18	Data Payload (Retirado de ^[15])	33
5.1	Diagrama de Distribuição	35
5.2	Arquitetura de Controle	36
5.3	Modelo de Entidade Relacional do Banco de Dados	37
5.4	Ligação Raspberry Pi - Módulo Zigbee	38

5.5	Tela de Login	39
5.6	Tela Inicial	40
5.7	Tela Sala	41
5.8	Tela Sala	41
6.1	Circuito LDR (Reproduzido de ^[29])	48
6.2	Circuito Dimmer (Reproduzido de ^[29])	49
6.3	Ligação Raspberry Pi - Módulo Zigbee	49
6.4	Teste Login	50
6.5	Teste Tela Inicial	50
6.6	Teste Carregamento Sala	50
6.7	Teste Função	51

Lista de tabelas

4.1	Tabela Comparativa de Tecnologias Sem-fio (Adaptado de ^[14]) . .	30
5.1	Especificações Raspberry Pi Modelo B	37
5.2	Especificações Operacionais	42
5.3	Especificações Elétricas	42
5.4	Especificações MQ6 ^[24]	43
5.5	Especificações MQ2 ^[25]	43
5.6	Especificações DS18B20 ^[26]	44
5.7	Especificações LDR	45
5.8	Especificações FR-04 ^[27]	45
5.9	Especificações Sensor de IR O52116	46
5.10	Especificações IR333 ^[28]	46

Resumo

Novas tecnologias de automação residencial, que visam melhorar a qualidade de vida do usuário, são limitadas por custos de implementação, por escalabilidade ou por complexidade de uso. Propomos aqui uma arquitetura modular de automação que permita tanto automatizar atividades cotidianas quanto viabilizar atividades que são demasiadamente complexas para execução manual. O sistema poderá ser monitorado e comandado a partir de dispositivos móveis, tanto internamente quando externamente. Ademais, a arquitetura proposta será de baixo custo e escalável, e considerará questões de facilidade de uso, robustez e segurança. Após o estudo realizado, optamos por uma arquitetura de controle que será parte centralizada e parte descentralizada. Neste caso, os controles simples serão processados nos módulos e os controles mais complexos, em uma central. Isso reduz o tráfego de dados pela rede interna e aumenta a robustez do sistema. A rede interna de comunicação entre os módulos será feita através da tecnologia sem-fio utilizando o protocolo ZigBee. A não necessidade de intervenções estruturais na residência e a segurança que o protocolo ZigBee oferece nos motivou a optarmos por essa tecnologia. A interface está na internet, para que permita o acesso através de diversos dispositivos e plataformas, e é modular, dividida pelos cômodos da residência, de modo a ser intuitiva.

Palavras Chave: automação residencial, domótica, zigbee, arduino, IEEE802.15.4, controle térmico, controle luminoso, automação predial.

Abstract

New technologies for home automation, to improve life quality of the user, are limited by costs of deployment, scalability and complexity of use. The intention here is to create a modular architecture that allows both automation of daily activities as viable activities that are too complex for manual execution. The system can be monitored and controlled from mobile devices, both internally as externally. Moreover, the proposed architecture is low cost, scalable, and consider issues of ease of use, robustness and security. After the study, we chose a control architecture that will be part centralized and part decentralized, the simple control will be rendered in the modules and the complex control will be in the central. This reduces the data traffic through the internal network and increases the robustness of the system. The internal network of communication between modules is done through wireless technology using the Zigbee protocol. The not necessary structural interventions in the home and the security that the Zigbee protocol offers motivated us to choose Zigbee as protocol technologie. The web interface will be for permitting access across multiple devices and platforms, and will be modular, divided by rooms of the house, so as to be intuitive.

Key Words: home automation, home automation, zigbee, arduino, IEEE802.15.4, thermal control, light control, building automation.

1 Introdução

Pretendemos fazer com que algumas atividades em uma residência sejam possíveis, com o uso dos módulos por nós projetados, de forma automática, e transformar outras atividades penosas em situações simples de descomplexas.

Com a adição de módulos, pretendemos diminuir os diversos controles remotos que há em uma residência. Além disso, será possível que algumas atividades sejam pré-programadas para que quando o usuário chegue em sua casa elas estejam prontas. Por exemplo, ao detectar que o usuário está chegando no perímetro da casa, as janelas poderão ser abertas para arejar a casa, a banheira já começar a encher e que o alarme da residência já esteja desativado.

Todas as atividades serão possíveis com uma interface amigável, feita para um usuário sem conhecimento prévio do sistema e de forma segura. Haverá a preocupação contra acessos ao sistema por pessoas não autorizadas. Por isso haverá uma criptografia e autenticação digital entre a interface do usuário e o módulo de controle central.

Além disso, há a preocupação de projetar um sistema escalável no sentido em que o projeto seja expansível com a adição de outros módulos projetados por nós. Com isso não há a necessidade de se fazer um projeto dedicado para a residência. Dessa forma o usuário poderá adicionar um módulo conforme a necessidade e disponibilidade financeira.

2 Estado da Arte - Automação Residencial

Ao acessar seu e-mail ou utilizar uma impressora um usuário comum não lembra ou percebe a complexidade da estrutura de rede por trás dessas ações^[1]. Da mesma forma, a automação residencial deve funcionar de forma transparente para o usuário e ao mesmo tempo seu uso deve ser simples e intuitivo.

A fim de tornar natural o controle dos dispositivos inteligentes o paper ^[2] propõe uma interface 3D para o controle residencial. O usuário andaria pela residência digital através de um avatar e poderia interagir e receber feedbacks do ambiente como se estivesse no ambiente. Por exemplo, para o estado de uma lâmpada seria necessário somente caminhar com o avatar até o cômodo onde a lâmpada está e olhar na sua direção e para apagá-la o usuário enviaria um comando para o avatar interagir com o interruptor. A simplicidade da interação com o usuário conflita com a dificuldade de instalação desse tipo de solução, a instalação inicial demanda a modelagem 3D da residência. Uma simples mudança na posição de uma estante ou eletrodoméstico implicaria na necessidade de alteração do modelo 3D.

O problema com a flexibilidade e escalabilidade torna a solução ^[2] pouco atraente. Visando esses pontos, o trabalho ^[3] foi desenvolvido. O hardware construído com custo de cerca de US\$ 40,00 possui 10 funções ligadas a automação residencial pré-programadas e com espaço para mais 5 funções, saídas analógicas e digitais, e um relé embutido. As funções pré-programadas permitem controle on/off, controle por tempo e por contagem, controle de temperatura entre outros. A principal limitação do sistema é a memória do microcontrolador utilizado, dos 8Kbytes que possui de ROM quatro são utilizados nas funções pré-programadas. A interface de controle do usuário exige conhecimentos em lógica de programação por parte do usuário. Pela simplicidade dos módulos, toda lógica de controle dos aparelhos conectados é feita no programa de controle do usuário. Para criar a inteligência dos dispositivos controlados o usuário se depara com blocos de programação que lembram programação em CLP.

O controle por voz, proposto no trabalho ^[4] tornaria, segundo seus autores, o uso fácil e intuitivo, principalmente para pessoas de mais idade ou com pouca instrução. Os testes realizados mostraram uma eficiência de cerca de oitenta por cento na captação dos comandos de voz. O teste foi realizado com a ajuda de 35 pessoas que não tem o inglês como língua nativa. O dispositivo de captação dos comandos de voz é tecnicamente simples, ele capta, comprime e envia a mensagem de voz. Todo processamento de reconhecimento é feito através de um PC, o que encarece o projeto. A taxa de velocidade de transmissão dos dados é de 250 kbps e mesmo com os filtros e a compressão utilizada pode aumentar significativamente o tempo de resposta de alguns comandos mais complexos.

Para diminuir a necessidade de tráfego de dados e diminuir o uso de processamento em central de comandos, o trabalho ^[5] propõe um sistema de supervisão descentralizada dos HAD. Ao invés dos dispositivos serem conectados diretamente a central de controle, nesse modelo, eles se conectariam a módulos de supervisão e esses seriam conectados a central. Esses módulos supervisionariam o estado dos dispositivos e somente se alguma condição for verdadeira, um sinal é enviado à central para que alguma ação seja tomada. Por exemplo, em um sistema de controle térmico, somente quando a temperatura atingir certo valor uma mensagem é enviada a central.

Além da descentralização da supervisão é possível descentralizar o controle conforme estudado no paper ^[6]. A centralização do controle aumenta o risco de falha, a necessidade de processamento computacional e o tráfego de dados, acarretando a possibilidade de gargalos. O sistema proposto consiste em subcentrais autônomas de controle que supervisionam e atuam independente da central. Nessa arquitetura a central pode passar os parâmetros de controle as subcentrais, e a partir desses parâmetros todo o controle e atuação é realizado independentemente. Assim se a central falhar, as subcentrais não necessariamente param de funcionar, dando ao sistema uma maior confiabilidade.

O trabalho ^[7] introduz ao sistema de controle descentralizado a tecnologia sem fio. O uso da tecnologia wireless dá flexibilidade ao sistema e diminui a complexidade e custos de instalação, já que dispensa a instalação de fiação. O baixo alcance do sistema, devido ao transmissor utilizado, é um fator limitante que é agravado em ambientes fechados como em uma residência.

O estudo de tecnologias que dispensam a instalação de novos fios (no-new-wires) não é recente o padrão X-10 foi criado em 1978, padrão que utiliza a rede de transmissão de energia como meio de comunicação entre os dispositivos inteligentes

e a central de controle. Quase todo dispositivo eletrônico está conectado a rede elétrica ou possui ao menos possui uma tomada próxima possibilitando assim a criação de uma rede de controle sem a necessidade de quebrar paredes ou a passagem de novos fios. O alto nível de ruído e a perda significativa de sinal ao longo dos cabos são os grandes problemas desse tipo de tecnologia. Esse padrão de rede é one-way, ou seja, os dados somente são enviados da central para os dispositivos inteligentes, o que dificulta na leitura de sensores e no feedback para o usuário.

Com o pensamento no-new-wires o fórum Home PNA foi fundado 1998, fórum que desenvolve um padrão de comunicação para dispositivos inteligentes utilizando a rede de telefonia. A última versão do padrão Home PNA 3.1, lançada em dezembro de 2006, permite velocidades de até 240 Mbps. Essa tecnologia sofre também com os ruídos gerados pela atividade de telefonia e pela rede elétrica, com a desvantagem de normalmente não estar presentes em todos os cômodos de uma residência.

3 Requisitos

3.1 Requisitos Funcionais

Trabalhar em uma frequência que seja livre e que não necessite de licenciamento. Isso significa não utilizar frequências definidas para algum propósito em específico, como por exemplo, TV por satélite, telefonia ou internet wireless.

Taxa de transmissão suficiente para envio de comandos complexos. O ideal seria ter uma transmissão que não fosse limitada, independente do propósito como por exemplo a transmissão de vídeo. Considerado como requisito funcional, temos que ter uma taxa mínima de transmissão que atendas todas as nossas necessidades de comando.

Segurança. Imprescindível ter proteção contra agentes externos indesejados no comando do sistema de automação residencial.

Retrofitting. Muitas vezes para se agregar uma nova tecnologia à residência do usuário, há a necessidade de fazer grandes modificações. A não necessidade de intervenção na edificação atual para na implementação de um projeto de automação residencial é um dos nossos principais requisitos funcionais.

Confiabilidade. Para um sistema de monitoração da residência não há possibilidades de falhas. Isso tiraria a principal finalidade do sistema. Mesmo para outras funções, não podemos tolerar falhas que inviabilizariam a utilização da residência.

Interface do usuário intuitiva. Apesar de algumas funções da automação serem complexas, não pretendemos que o usuário necessite de usar um manual para cada função. Isso tornaria o uso penoso, o que muitas vezes faria com que o usuário deixe de usar a automação na sua residência.

Interface flexível para conseguir gerar funções complexas. Independente da quantidade de funções que o sistema de automação possuir, a interface deve suprir todos os possíveis comandos programados sem que haja uma necessidade

de atualização da interface.

Além disso, o sistema deve permitir:

- Controlar os dispositivos inteligentes
- Cadastrar regras de controle
- Cadastrar novos cômodos
- Cadastrar novos dispositivos
- Cadastrar novos usuários

3.2 Requisitos Não Funcionais

Baixo consumo energético. Com a crescente demanda por produtos que são politicamente corretos em relação ao meio ambiente, o consumo de energia é considerado um tipo de apelo comercial para o produto.

Facilidade de instalação. Muitos projetos necessitam de técnicos especializados para a instalação de funcionalidades em uma automação residencial. O ideal é que o próprio usuário fosse capaz de instalar e configurar os módulos do nosso projeto.

Interface compatível com múltiplos dispositivos. O ideal para um projeto de automação é que ele seja compatível com qualquer dispositivo móvel que o usuário possua. Para isso, a interface deve ser compatível com todas as plataformas existentes (Android, iOS, Windows Phone, etc).

4 Análise de Soluções

4.1 Arquitetura de Supervisão e Controle

A arquitetura de controle e de supervisão dos HAD (Home Automation Devices) é responsável pela supervisão dos dispositivos, através de seus status e de sensores, além da aplicação das regras de controle, através dos atuadores do sistema.

4.1.1 Supervisão e Controle Centralizados

Aplicada nos trabalhos [3], [4], [8] e [9], essa arquitetura utiliza uma central para o processamento das camadas de supervisão e controle. Na figura 4.1 abaixo podemos ver um exemplo.

A utilização desse tipo de arquitetura necessita de uma central com boa capacidade de processamento, normalmente um PC ou uma plataforma equivalente, o que encarece o projeto. O aumento no número de dispositivos conectados também pode criar um gargalo no meio de comunicação com o PC, afetando a resposta do sistema. A central também se torna um ponto crítico do projeto, pois em caso de falha desse componente, todo o sistema para de funcionar.

4.1.2 Supervisão Descentralizada e Controle Centralizado

No paper [5] foi proposto um meio de prevenir os gargalos que ocorrem em sistemas totalmente centralizado. Nessa proposta, a camada de supervisão foi descentralizada. Módulos intermediários são responsáveis pela verificação dos sensores e estados dos HAD, e somente quando alguma regra de supervisão é ativada o módulo envia um sinal para a central. E apenas após isso ela aplica as regras de controle. Na figur 4.2 é mostrado o sistema de hierarquia proposto no trabalho [5].

Com essa arquitetura é possível reduzir a necessidade de processamento da central e o fluxo de dados na camada de comunicação. O principal problema

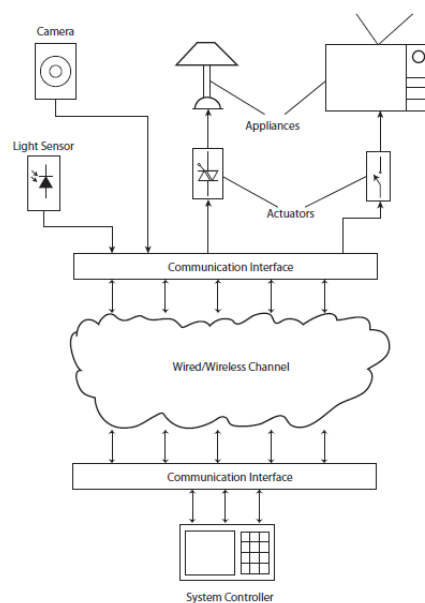


Figura 4.1: Arquitetura Centralizada(Reproduzido de ^[9])

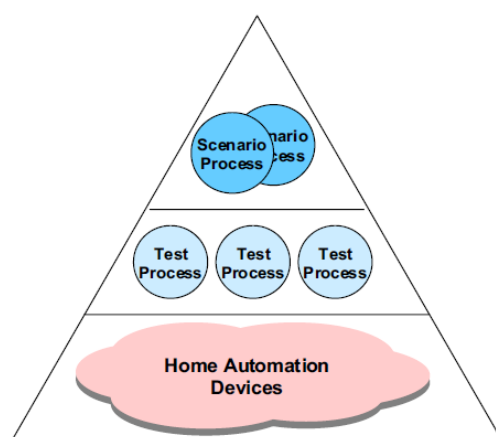


Figura 4.2: Arquitetura Descentralizada(Reproduzido de ^[5])

desse tipo de solução é que os módulos de supervisão têm apenas visão parcial do sistema, ou seja, trabalha somente com os dispositivos a ele conectado, e isso afeta diretamente na flexibilidade das regras de controle e supervisão.

4.1.3 Supervisão e Controle Descentralizada

Com o foco na diminuição de custos, propusemos um modelo em que não somente a supervisão, mas também o controle é descentralizado. Nessa arquitetura as regras de controle e supervisão são processadas em módulos que estão conectados aos HAD, e alteração das regras é feita através da mudança dos parâmetros de referência. Essa solução não elimina a necessidade de uma central, devido a necessidade de agrupamento de todos os dispositivos para a operação do usuário, mas reduz a sua necessidade de processamento assim como o fluxo de dados para a central, além da vantagem de em caso de falha da central os módulos ainda aplicarão as regras de controle.

Como na solução anterior, essa arquitetura impossibilita a utilização de regras de controle e supervisão que envolva dispositivos conectados a diferentes módulos. Além disso, essa arquitetura aumenta a necessidade de processamento dos módulos e conseqüentemente os custos dos mesmos.

4.1.4 Supervisão e Controle Híbridos

Pensando em reduzir os custos da central e também dos módulos dos HAD de uma forma que impossibilite regras de controle complexas, propomos uma arquitetura de supervisão e controle híbrida, isto é, parte do controle é descentralizada e parte é centralizada.

A fim de reduzir a necessidade de processamento e de comunicação com a central alguns controles simples seriam feitos nos módulos. Outras rotinas de supervisão e controle complexas ficariam a cargo da central.

Por exemplo, em um módulo responsável pela iluminação de um cômodo, é possível descentralizar uma função de controle em malha fechada que mantém a iluminação constante durante todo dia, pois essa funcionalidade envolve somente sensores e atuadores desse módulo. A central ficaria responsável apenas por enviar, quando necessário, a que luminosidade o cômodo deve ser mantido.

4.2 Central

Independente do tipo de arquitetura de supervisão e controle utilizados, é necessário um ponto central para a integração dos HAD, seja ele para o controle dos dispositivos ou somente como centro de comunicação. Essa central, na literatura estudada, também é utilizada para a hospedagem da interface de controle do usuário.

4.2.1 Central Local

Utilizada na maioria dos artigos pesquisados, um ponto central na casa é responsável por hospedar as camadas de supervisão e controle, além da interface de controle do usuário.

4.2.2 Central na Nuvem

Nesse tipo de central, as regras de controle e supervisão são processadas em servidor remoto. E a interação entre a casa e a central de controle fica a cargo uma central de comunicação que recebe os comandos através da internet e os retransmite para os HAD através da rede interna da residência.

Para esse tipo de aplicação é necessário o aluguel de servidores na nuvem onde seriam mantidas as camadas de supervisão e controle. A dependência contínua da conexão com a internet também pode ser um problema.

4.3 Interface de Controle do Usuário

A interface de controle do usuário é a parte importante da interação entre o morador e os dispositivos inteligentes em sua residência. Nela o usuário alterará as regras de controle e os parâmetros que garantirão o conforto e a segurança do morador. Esse controle pode ser feito por diversas formas como, por exemplo, através de interface gráfica, como é mais usual, ou através de interfaces de controle por voz ou gestos. O controle do usuário deve ser fácil e intuitivo, à medida que o nível de conhecimento e proximidade com a tecnologia é heterogêneo entre os moradores de uma residência.

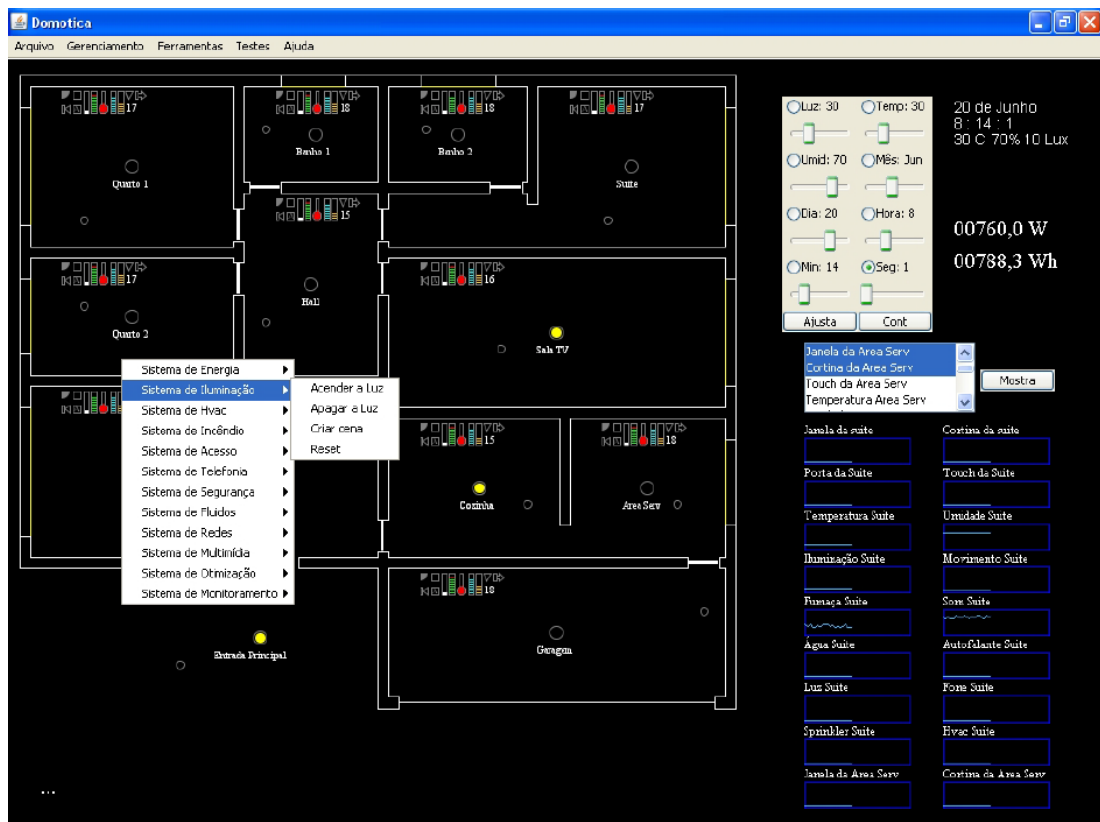


Figura 4.3: Exemplo de Interface utilizando a planta-baixa (Reproduzido de [8])

4.3.1 Interfaces Gráficas

4.3.1.1 Planta Baixa

Aplicado nos trabalhos [10] e [8], essa interface de controle tem como base a planta baixa da residência onde são representados os HAD (Home Automation Devices). A figura 4.3 mostra a interface desenvolvida no trabalho [8]. Os estados dos dispositivos são mostrados através de indicadores posicionados próximos a sua representação na planta o que torna muito fácil e rápido sua leitura, por outro lado as opções de controle dos dispositivos ficam escondidas e seu acesso é feito através de um submenu. A necessidade da transcrição da planta baixa da residência para o programa de controle, além do posicionamento de todos os HAD, dificulta e encarece a implementação desse tipo de interface. Qualquer alteração na casa ou na disposição dos HAD também é um empecilho nesse tipo de interface.

4.3.1.2 Modelo 3D da Residência

A utilização de um ambiente virtual 3D, proposto em [2], tem como principal vantagem a facilidade e naturalidade de visualização de estados dos ambientes e dispositivos controlados. A comodidade de se caminhar no ambiente, como se



Figura 4.4: Exemplo de Interface 3D (Reproduzido de [2])

fosse a residência real, esbarra no controle do avatar (representação virtual do usuário), que exige habilidade nos comandos com o mouse e teclado. A interação com os dispositivos para a alteração de seus estados é limitada a liga/desliga. E como na solução da planta baixa, a instalação e manutenção desse tipo de interface é trabalhosa. Um exemplo de um modelo 3D está na figura 4.3

4.3.1.3 Blocos de Programação

Esse tipo de interface, proposto em [3], utiliza blocos de programação para gerar os comandos de controle dos dispositivos inteligentes. Esses blocos permitem ao usuário programar funções complexas e altamente customizadas para os dispositivos, mas o obriga a ter conhecimento em lógica de programação. Na figura 4.5

4.3.1.4 Divisão Modular

Pensando em uma interface que seja multiplataforma, criamos uma interface web que divide os ambientes da residência em módulos como pode ser visto no figura 4.6. Em cada modulo estão listados os dispositivos dos respectivos ambientes, onde é possível fazer uma leitura rápida dos seus estados. Ema leitura rápida dos sensores

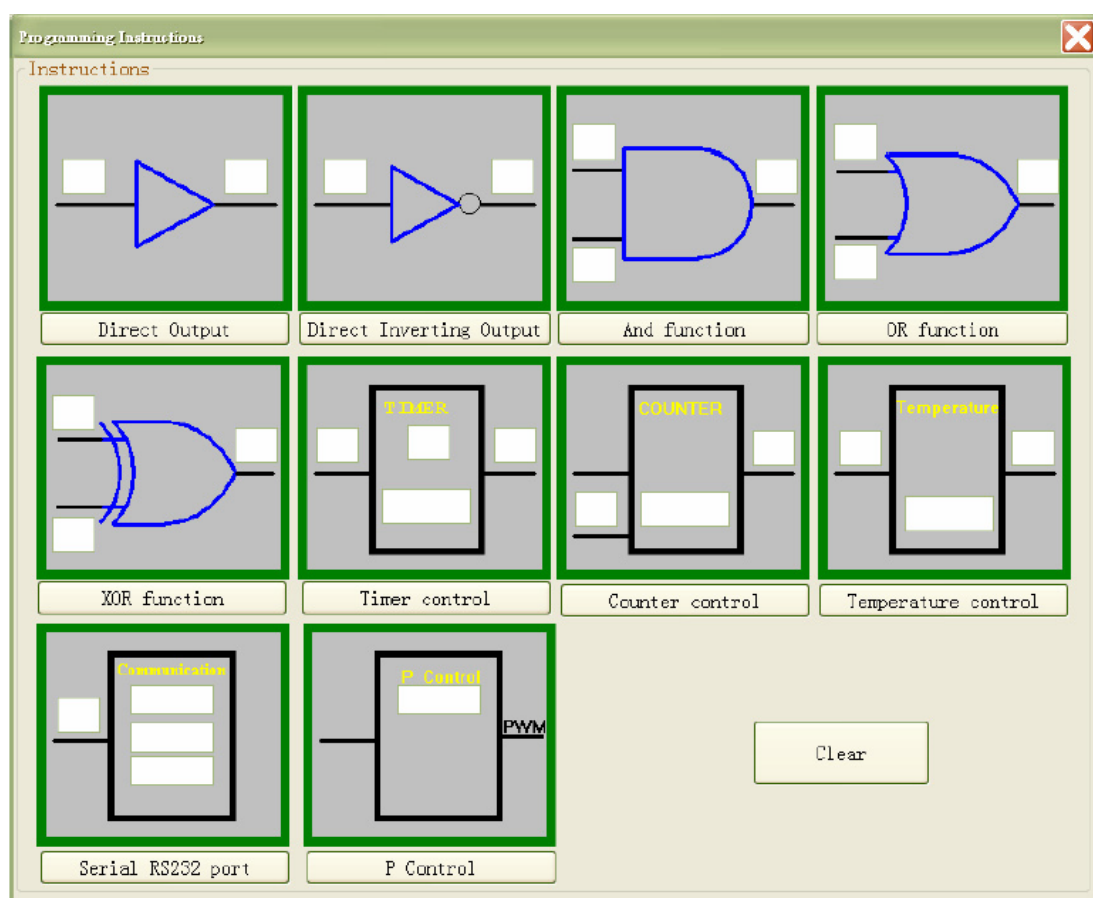


Figura 4.5: Exemplo de Interface com Blocos de Programação (Reproduzido de [3])

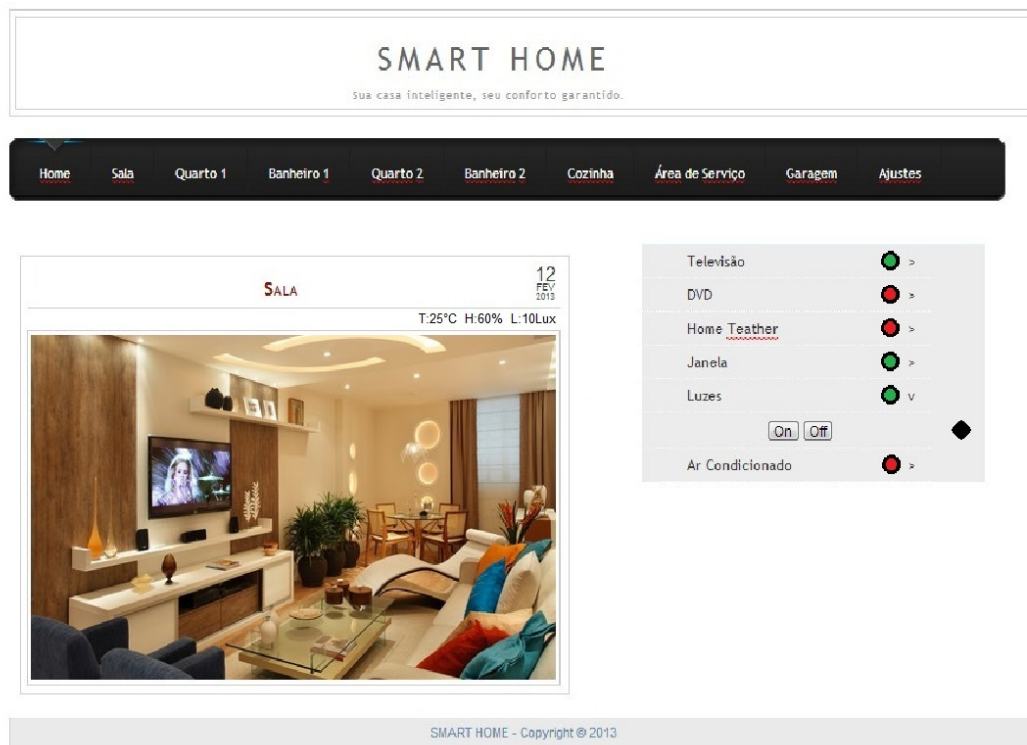


Figura 4.6: Exemplo de com Divisão Modular

do ambiente pode ser feita através dos dados mostrado logo acima da foto do ambiente. No layout proposto foi utilizado uma figura para ajudar a identificar o ambiente, mas existe a possibilidade de colocar a imagem da câmera de segurança do ambiente no lugar em questão.

Ao se selecionar um dos dispositivos é aberto, logo abaixo, um menu com as opções de controle simples e a direita desses um botão que permite a inserção de controles complexos como temporização e disparo. No layout de exemplo é mostrado o menu de controle de luzes.

4.3.2 Controle por Voz

O controle por reconhecimento de voz implementado no trabalho ^[4] utiliza um hardware simples que compacta os comandos de voz e envia a um computador central que analisa e realiza o comando enviado. O hardware simples filtra as frequências captadas pelo microfone deixando somente as necessárias para o entendimento da mensagem pelo computador. Para o reconhecimento de voz foi utilizada o aplicativo Microsoft Speech API, e segundo os testes realizados foi obtido uma taxa de sucesso de 79,8

Apesar dos filtros e da compactação realizada no hardware de captura do comando de voz as taxas de transmissão de dados ainda podem gerar atrasos entre

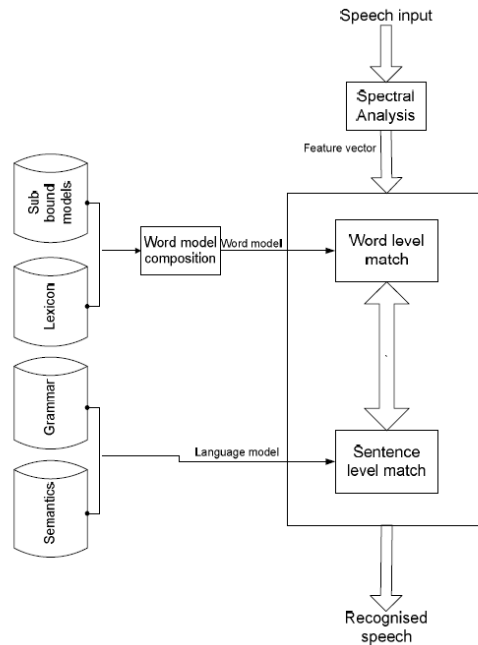


Figura 4.7: Fluxograma Simplificado da API de Reconhecimento de Voz (Reproduzido de ^[4])

o comando e a execução por parte da central. A necessidade de um computador para o processamento de voz também encarece o projeto. Na figura 4.7 é um fluxograma simplificado da API de reconhecimento de voz.

4.3.3 Controle por Gestos

O controle por gestos dispensa a necessidade de qualquer hardware “preso” ao usuário. Hoje já existem aplicações comerciais desse tipo de tecnologia como, por exemplo, o Kinect da Microsoft, utilizado para o controle do videogame Xbox 360, e tecnologia Smart Interaction da Samsung, em que o sensor é utilizado para controlar as funções da televisão.

A utilização desse tipo de tecnologia utiliza sensores que tem custo elevado se comparado ao tipo de interface de controle. E para a sua aplicação em uma residência seria necessário a instalação desses sensores em cada cômodo da casa, o que aumenta significativamente o preço do uso desse tipo de tecnologia. A necessidade do correto posicionamento do usuário para a correta captura dos gestos, na distancia correta e sempre de frente para os sensores, fez com que concluíssemos que essa solução é pouco confortável para um ambiente doméstico.

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Figura 4.8: S-BOX (Retirado de ^[11])

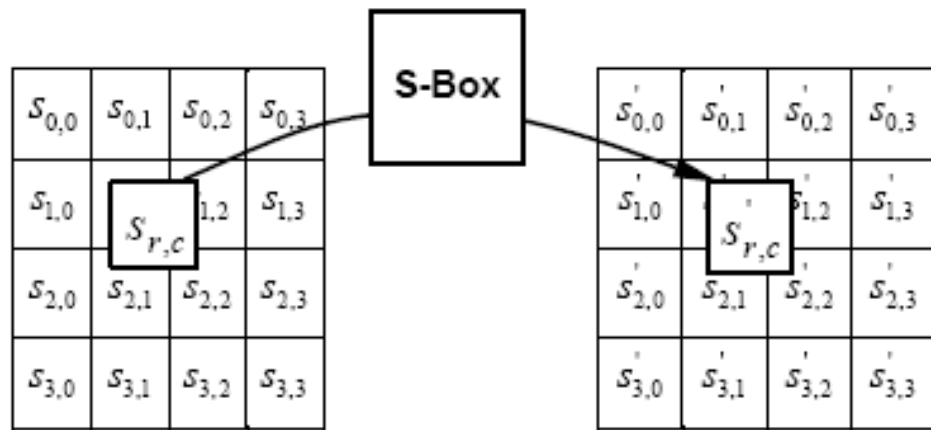
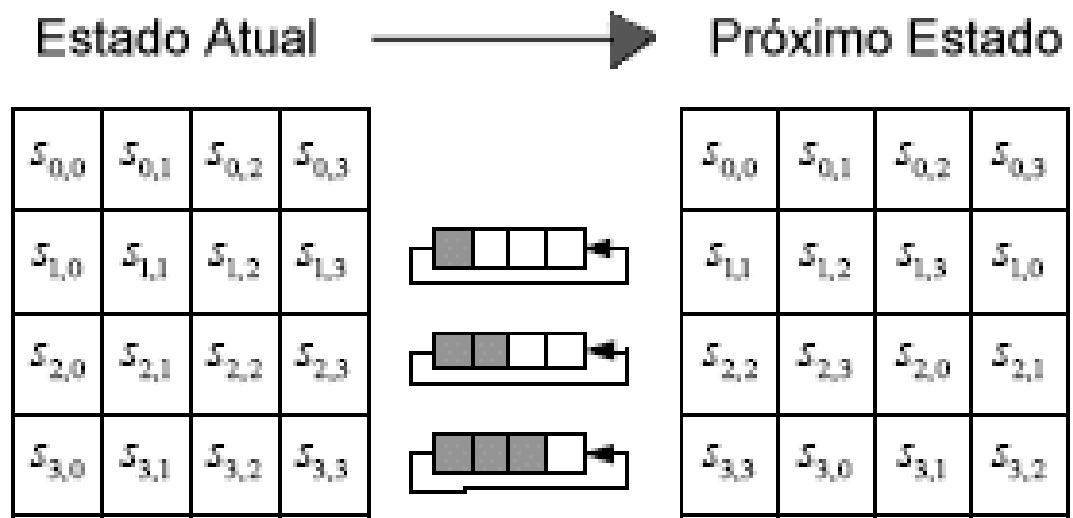
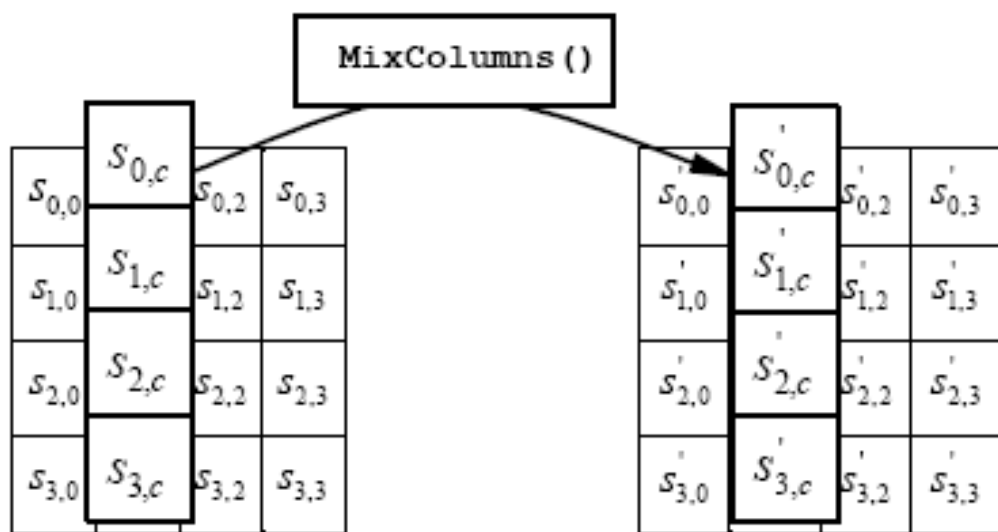
4.4 Segurança

4.4.1 AES

A criptografia de chave simétrica AES trabalha com blocos de 128bit e permite chaves de 128, 192 ou 256bits. Esse algoritmo não é considerado o mais seguro, mas é um dos mais eficientes. Ainda sim ele apresenta alta resistência a ataques como power attack e timing attack. Ele não é usado apenas para criptografar a informação, mas também para validar a informação enviada. Ele também é relativamente fácil de executar e requer pouca memória.

O AES opera sobre um estado (arranjo bidimensional de bytes com 4x4 posições). O processo de criptografia consistem em 9 turnos de 4 estágios (exceto o último, que possui 3):

- SubByte: neste estágio cada byte do estado é substituído seguindo a tabela de substituição não linear S-BOX (Figura 4.8) de 8 bits fixa. Figura 4.9.
- ShiftRows: as linhas da matriz estado são ciclicamente deslocadas, onde a linha 0 permanece fixa e as linhas 1, 2 e 3 são deslocadas de 1, 2 e 3 bytes respectivamente. Figura 4.10.
- MixColumns: os 4 bytes de cada coluna do estado são combinados usando uma transformação linear invertível. Cada coluna é tratada como um polinômio e é então multiplicado em módulo por um polinômio fixo Figura 4.11.
- AddRoundKey: nesse estágio uma subchave é combinada com o estado. Em cada turno uma subchave é derivada da chave principal. Figura 4.12.

Figura 4.9: Estágio SubByte (Retirado de ^[11])Figura 4.10: Estágio ShiftRows (Retirado de ^[11])Figura 4.11: Estágio MixColumns (Retirado de ^[11])

$$\begin{array}{|c|c|c|c|c|c|} \hline a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} & a_{0,4} & a_{0,5} \\ \hline a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ \hline a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} \\ \hline a_{3,0} & a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ \hline \end{array}
\text{ XOR }
\begin{array}{|c|c|c|c|c|c|} \hline k_{0,0} & k_{0,1} & k_{0,2} & k_{0,3} & k_{0,4} & k_{0,5} \\ \hline k_{1,0} & k_{1,1} & k_{1,2} & k_{1,3} & k_{1,4} & k_{1,5} \\ \hline k_{2,0} & k_{2,1} & k_{2,2} & k_{2,3} & k_{2,4} & k_{2,5} \\ \hline k_{3,0} & k_{3,1} & k_{3,2} & k_{3,3} & k_{3,4} & k_{3,5} \\ \hline \end{array}
=
\begin{array}{|c|c|c|c|c|c|} \hline b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} & b_{0,4} & b_{0,5} \\ \hline b_{1,0} & b_{1,1} & b_{1,2} & b_{1,3} & b_{1,4} & b_{1,5} \\ \hline b_{2,0} & b_{2,1} & b_{2,2} & b_{2,3} & b_{2,4} & b_{2,5} \\ \hline b_{3,0} & b_{3,1} & b_{3,2} & b_{3,3} & b_{3,4} & b_{3,5} \\ \hline \end{array}$$

Figura 4.12: Estágio AddRoundKey (Retirado de ^[11])

4.5 Rede Doméstica

Para a definição da rede de dados a ser utilizada para a comunicação entre os HAD, priorizamos as redes que tem como característica não ser invasivas, ou seja, que não necessitem grandes intervenções, se instalada numa casa pré-existente ou modificações em uma planta de uma nova residência.

Com essa premissa estudamos inicialmente três tipos de redes: a primeira baseada na rede telefônica de uma casa, a segunda na rede de transmissão de energia e a terceira utilizando a tecnologia wireless.

4.5.1 Rede de Dados Baseada no Sistema Telefônico

Esse tipo de rede baseada na instalação telefônica da residência tem como principal frente o fórum Home PNA (Home Phoneline Networking Alliance). Esse fórum foi criado em junho de 1998 e hoje é liderado por quatro grandes empresas: Arris, AT&T, Cisco e Sigma. No final do primeiro ano de existência do fórum foi criado o primeiro padrão industrial o Home PNA 1.0, que permitia velocidades de até 1 Mbps. No ano seguinte foi lançado o segundo padrão do fórum o Home PNA 2.0, essa versão permitia velocidades de até 10 Mbps e era compatível com a primeira versão da especificação. As duas primeiras versões trabalham em frequências acima do sistema de telefonia analógica e de sistemas ISDN e DSL. Em novembro de 2001 foi lançada a versão 3.0 do padrão com velocidade de até 128 Mbps e introduziu a coexistência com sistemas ADSL. A última versão do padrão foi lançada em Dezembro de 2006 denominada Home PNA 3.1. Permite velocidades de até 320 Mbps e trabalha em frequência acima das dos telefones analógicos convencionais, assim como as primeiras versões, e permite coexistência com dispositivos ISDN, DSL, VDSL e ADSL. Essa versão trabalha com Multi-Spectrum Operation que permite a existência de mais de uma rede Home PNA na mesma fiação.

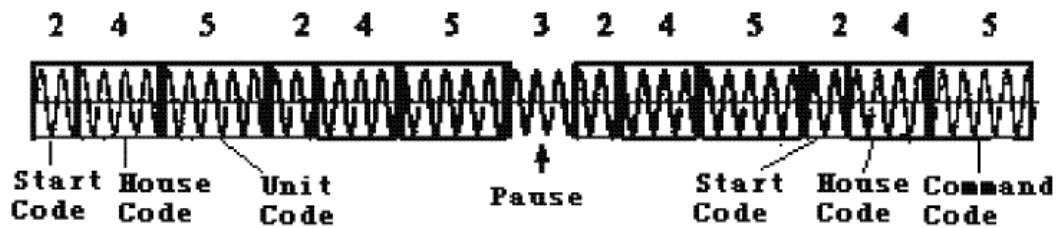


Figura 4.13: Frame de Comunicação X-10 (Reproduzido de ^[12])

4.5.2 Rede de Dados Baseada no Sistema de Distribuição Elétrica

Presente em quase todos os cômodos de uma residência a rede elétrica tem uma vantagem sobre a rede baseada em sistemas de telefonia. Conhecida comumente por Powerline Communication, existem diversas iniciativas para o desenvolvimento dessa tecnologia.

4.5.2.1 X-10

O protocolo X-10 foi proposto em 1978, e foi o primeiro protocolo de comunicação entre aparelhos utilizando a rede elétrica. Nesse padrão é possível endereçar até 256 dispositivos e transmitir dados em até 60 bps. É possível enviar comandos de turn on, turn off e dim. O sinal é enviado através de um burst (rajada de sinal) de 120 kHz. O frame de controle é por 22 dígitos binários conforme mostra a figura 4.13 e é enviado duas vezes com um intervalo de três ciclos de linha. A grande limitação desse protocolo, que também é seu ponto forte, é a sua simplicidade. Não é possível enviar comando de aumentar ou diminuir parâmetros e a comunicação é unidirecional, o que impossibilita, por exemplo, o feedback de sensores ou sistemas de segurança a central de comunicação.

4.5.2.2 CEBUS

O CEBus (Consumer Electronics Bus) é um padrão aberto baseado no modelo ISO/OSI. Permite velocidades de até 10 kbps. A transmissão é feita utilizando a tecnologia de espalhamento de espectral onde a transmissão é inicializada com pulsos de 100 kHz e aumenta linearmente até 400 kHz. Como pode-se observar o range de frequência utilizada conflita com o protocolo X-10. O CEBus utiliza o formato de rede peer-to-peer, é possível comunicação entre todos os nós, o que permite a não utilização de um sistema de controle centralizado. Para evitar colisões de dados o protocolo utiliza a tecnologia CSMA/CDRS. O padrão inclui

comandos básicos voltados a automação residencial como volume up/down, fast forward/rewind, pause e skip. O transmissão é baseada em strings ou pacotes de dados que podem ter tamanho variado, o que dá mais flexibilidade ao protocolo em relação ao X-10.

4.5.3 Rede de Dados Sem Fio

A tecnologia sem fio não é recente mas os altos preços e a baixa taxa de transferência inviabilizaram seu uso no passado. Hoje o avanço da tecnologia e iniciativas como o Zigbee Alliance permitiram a popularização dessa tecnologia.

4.5.3.1 IEEE 802.11a

O padrão 802.11a, lançado em 1999 utiliza frequência de 5 GHz, apesar da especificação 802.11 utilizar bandas próximas de 2.4 Ghz. Chega a velocidades de até 54 Mbps através da técnica de modulação OFMD (Orthogonal Frequency Division Multiplexing), técnica que divide uma única transmissão em varias em diversas faixas de frequência. O padrão 802.11a permite até 64 pontos de acesso.

4.5.3.2 IEEE 802.11b

O padrão 802.11b opera na frequência de 2.4 GHz e permite velocidades de até 11 Mbps. Foi lançado em julho de 1999, trabalha com duas técnicas de modulação a DSSS (Direct-Sequence Spread Spectrum) e a técnica CCK e permite conexão de até 32 pontos de acesso.

4.5.3.3 IEEE 802.11g

Esse padrão foi lançado em junho de 2003 e tornou-se rapidamente o padrão de mercado wireless. Trabalha na frequência de 2.4 GHz e permite compatibilidade com o padrão 802.11b. Com velocidades de até 54 Mbps. Utiliza a tecnologia OFMD e mantém a CCK para compatibilidade. Na compatibilidade com o padrão b a velocidade máxima fica limitada ao padrão mais lento (11 Mbps). O padrão 802.11g trouxe também melhorias na segurança suportando além do padrão WEP o padrão WPA que muda a senha periodicamente de forma automática.

4.5.3.4 IEEE 802.11n

Padrão visa altas velocidades e compatibilidade com as versões anteriores. Trabalhando na frequência de 2.4 e 5 GHz possui compatibilidade com os padrões 802.11a/b/g. A introdução da tecnologia MIMO (multiple-input multiple-output) garantiu um aumento da velocidade podendo chegar a até 540 Mbps. A tecnologia MIMO é utilizada em algumas soluções moveis como o LTE e envolve a utilização de múltiplas antenas de entrada e saída de dados.

4.5.3.5 Bluetooth

Tecnologia criada inicialmente pela Ericsson no início dos anos noventa, o Bluetooth é uma tecnologia que permite altas taxas de transmissão e baixo alcance. A transmissão de dados é feita através de pacotes em espalhamento espectral de 79 canais. As redes conectam-se através de piconets, pequenas redes, nela é determinado o padrão de transmissão de dados através dos canais disponíveis entre os dispositivos, somente os dois dispositivos conhecem o padrão através dos canais, o que garante a segurança da comunicação.

4.5.3.6 Zigbee

Padrão de rede sem fio focado em baixo custo, baixo consumo, segurança e confiabilidade. O padrão é definido por um fórum chamado Zigbee Alliance ^[13] liderado pelas empresas: Comcast, Freescale, Itron, Kroger, Landis+Gyr, Legrand, Philips, Reliant, Schneider Electric, Silicon Labs e Texas Instruments.

O ZigBee é baseado no protocolo IEEE 802.15.4, que opera nas frequências 868MHz, 915MHz e 2.4GHz. A taxa de transferência são 20kb/s, 40kb/s e 250kb/s respectivamente.

Em uma rede Zigbee os transceivers são divididos em três tipos de nó: Os coordenadores (coordinator), os roteadores (router) e o receptores (end device). Os três tipos podem ser vistos no exemplo da figura 4.14.

O protocolo permite dois tipos de topologia a topologia Cluster Tree Mesh e a topologia Full Mesh, exemplos do tipo de topologia podem ser visto na figura 4.14.

Os dispositivos possuem dois tipos de endereçamento, o IEEE Address de 64 bits, que é único para cada dispositivo IEEE 802.15.4 e não pode ser alterado, e o endereçamento da Rede Local que possui 16 que pode ser alterado e deve ser único em cada rede Zigbee.

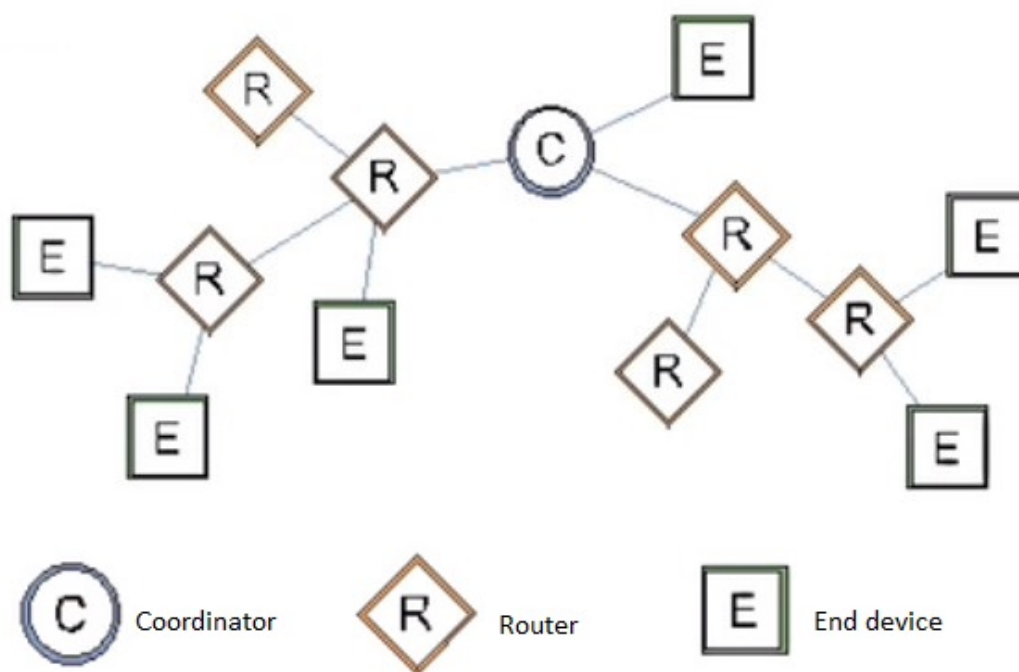


Figura 4.14: Tipos Tranceivers em uma Rede Zigbee (Reproduzido de ^[14])

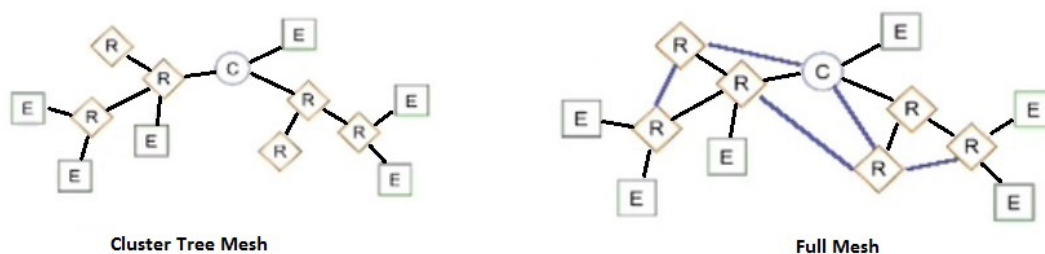


Figura 4.15: Topologias Zigbee (Reproduzido de ^[14])

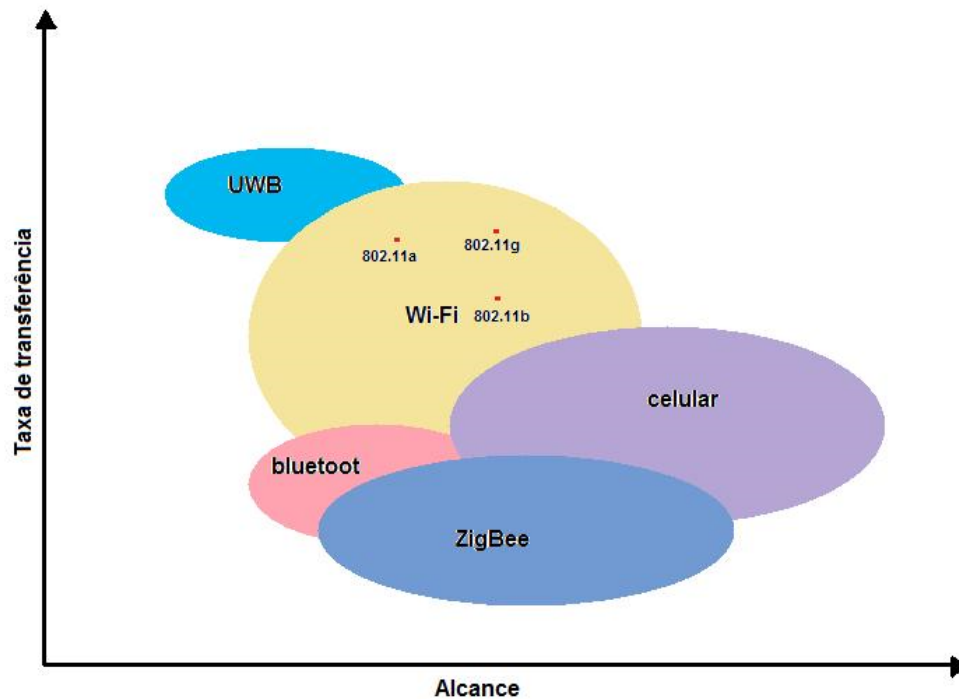


Figura 4.16: Gráfico Comparativo de Tecnologias Sem-fio (Adaptado de ^[14])

A transmissão pode ser feita em até 16 canais e o Zigbee Coordenador pode mudar o canal de comunicação se for detectada alguma interferência, deixando o sistema mais robusto contra interferências externas à rede. Para a segurança o Zigbee utiliza encriptação AES 128-bits.

Podemos ver a comparação entre os tipos de comunicação wireless estudados na figura 4.16 e na tabela.

Tipos de Zigbees

Coordenador:

- Necessário 1 por rede
- Define o canal e a PAN ID (Personal Area Network) para começar a rede
- Permite routers e end point unirem-se a rede
- Nunca pode hibernar

Router:

- Precisa unir-se a uma PAN antes de começar a transmitir, receber ou rotear dados
- Após unir-se a uma rede, permite a entrada de mais routers e end points
- Podem funcionar como roteador intermediário entre nós sem um coordenador

	IEEE 802.11b	Bluetooth	ZigBee
Bateria	Horas	Dias	Anos
Complexidade	Muito complexo	Complexo	Simples
Nós/master	32	7	64000
Latência	Contagem até 3 segundos	Contagem até 10 segundos	Contagem até 30 milisegundos
Alcance	100m-1000m	10m	70m-300m (ETSI), 1600m (FCC)
Expansão	Possível roaming	Não	Sim
Taxa de RF	11Mbps	1Mbps	250Kbps
Segurança	Serviço de autenticação por ID (SSID)	64-bit, 128-bit	128-bit AES e camada de aplicação definido pelo usuário

Tabela 4.1: Tabela Comparativa de Tecnologias Sem-fio (Adaptado de ^[14])

- Nunca pode hibernar

End point:

- Precisa unir-se a uma PAN antes de começar a transmitir ou receber dados
- Não permite a entrada de novos end point ou routers
- Não pode rotear dados
- Podem hibernar

Modos de atuação

AT (application transparent)

- Não permite mudança de parâmetros sem utilizar modo de configuração
- Não há checksum para integridade do frame
- Permite comunicação entre vários Zigbees mas necessita de comando para alterar endereço de destino

API (application programming interface)

- Permite receber dados I/O de múltiplos Zigbees
- Envia ACK (dado de recebimento de mensagem)

- Possui checksum para integridade do frame
- Permite fácil endereçamento de destino
- Permite mudança de parâmetros sem utilizar modo de configuração

Frame de comunicação

Nos módulos construídos utilizamos o Zigbee no modo comunicação API devido os requisitos do projeto, como integridade do frame de envio, múltiplos endereçamentos, segurança por criptografia.

O frame de envio tem o seguinte formato:

- Start byte (indica o início do frame)
- Length (tamanho do frame)
- Frame type (Indicador do modo de operação - AT ou API)
- Frame id (Identificação do frame)
- Destination address (endereçamento)
- Data packet
- Checksum (verifica integridade do frame)

O frame de resposta tem o seguinte formato:

- Start delimiter
- Length byte
- Identificador de modo
- Frame id
- Status byte
- Checksum

Criptografia AES no Zigbee

O Zigbee possui duas camadas de segurança, a camada da rede e a camada de aplicação. As duas camadas seguem a norma IEEE 802.15.4 e utilizaram o padrão AES. Há três tipos de chaves: master, link e network key.

- Master key: é a chave pré-instalada no Zigbee. Ela é utilizada no procedimento de definição de chaves para se manter a link key confidencial
- Link key: são únicas entre dois pares de nós. Ela é utilizada pela camada

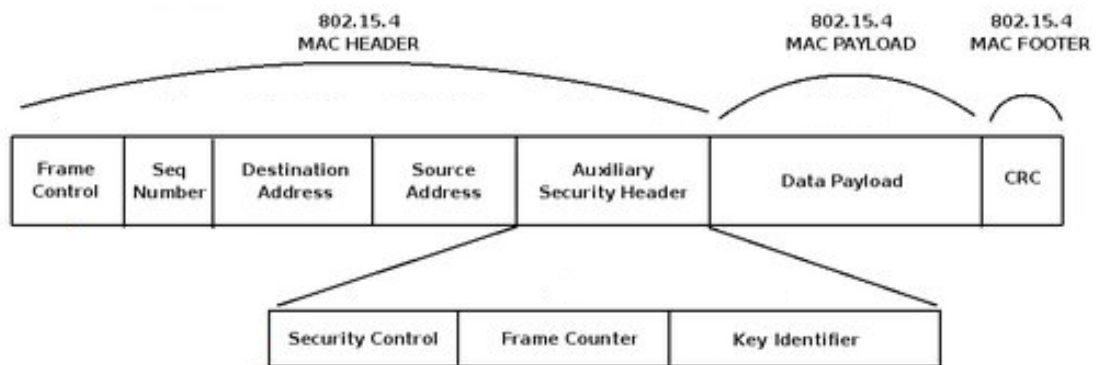


Figura 4.17: Auxiliary Security Header (Retirado de ^[15])

de aplicação e é utilizada para criptografar os dados entre dois aparelhos.

- Network key: é a chave definida pelo coordenador e comum a todos aparelhos da rede. Ela é trocada em diferentes intervalos de tempos e todo aparelho que deseja entrar na rede precisa utilizar essa chave

Todo frame de saída o Zigbee possui um subframe no cabeço para as configurações da criptografia (figura 4.17). Nele é definido os seguintes campos:

- Security control: habilita ou desabilita a criptografia AES de 128bits
- Frame counter: é um contador fornecido pela fonte do frame para proteger a mensagem de replicações
- Key identifier: define qual chave e para qual nó de destino é a mensagem

Quando a criptografia é habilitada no Zigbee todos os dados são criptografados em AES 128bit e o subframe de dados assume o seguinte formato (figura 4.18):

- Frame counter: é um novo contador fornecido pela fonte para definir um único identificador da mensagem
- Key control: quando o Frame counter assume o valor máximo, esse contador é usado para se ter todo dado encriptado
- Encrypted payload: dado criptografado
- Encrypted MAC: é o subframe de autenticação do dado

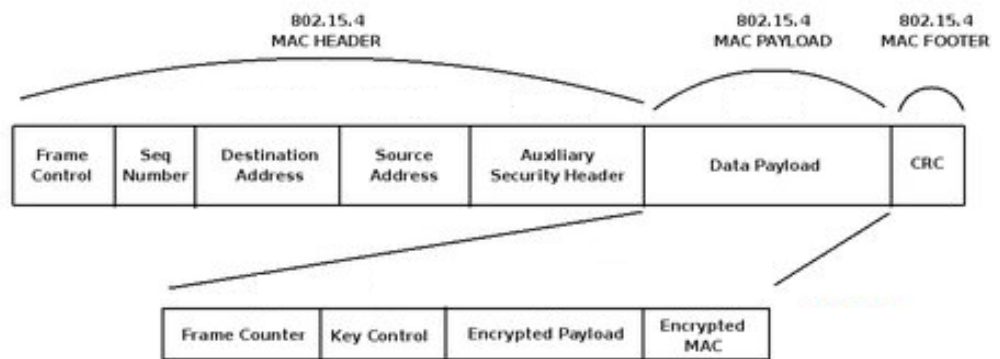


Figura 4.18: Data Payload (Retirado de ^[15])

5 Solução Desenvolvida

Após o estudo realizado, definimos a arquitetura de rede e de controle que será utilizada no sistema. Optamos por uma arquitetura de rede com duas centrais, uma local e uma na nuvem, que garante o acesso remoto as funcionalidades da residência. A rede local, formada pelo central local e os módulos de controle será sem-fio utilizando a tecnologia ZigBee. A arquitetura de controle será parte distribuída e parte centralizada, formada por módulos de controle e uma central. As seções abaixo detalham o sistema desenvolvido.

5.1 Arquitetura de Rede

Após o estudo realizado optamos por uma de rede com duas centrais, uma interna e uma externa, e rede local sem-fio, na figura 5.1 pode ser visto o Diagrama de Distribuição da arquitetura que foi implementada.

A Central Local será o HUB de acesso do usuário para os dispositivos controláveis, além disso, é onde foi implementada toda a interface do sistema. A central é responsável também pela supervisão e controle complexos, por complexo entendem-se regras que envolvam mais de um módulo ou que envolvam agendamentos de tarefas. A central pode ser ligada a rede Wi-Fi (IEEE 802.11 b/g/n) ou a rede Ethernet Cabeada (RJ45) da residência, e é através dessa rede o usuário terá acesso a Interface. Essa conexão também é utilizada para a comunicação entre a Central Local e a Central Externa.

A Central na Nuvem é um espelho da Central Local em relação à base dados e a interface, e permite o acesso remoto (externo a residência) às funcionalidades do sistema. A sincronia entre as centrais fica a cargo de um Web Service. Esse serviço é requisitado do Servidor Local para o Servidor Remoto, já que somente o Servidor Remoto possui IP fixo. A base de dados na Nuvem é inteiramente criptografada, portando durante o processo de sincronismo é necessário encriptar e decryptar os dados. Na interface da aplicação na Nuvem a encriptação e decryptação foi implementada na camada Javascript da aplicação (Client), isso garante que

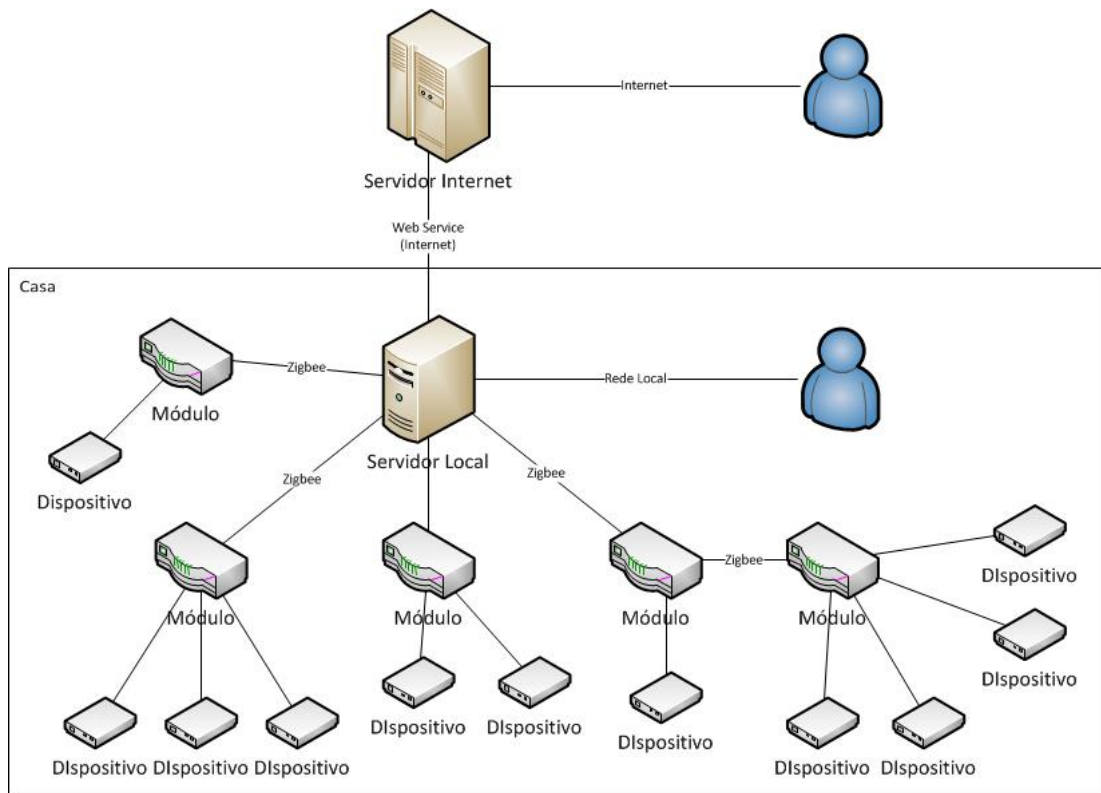


Figura 5.1: Diagrama de Distribuição

informações decriptadas não transitem pela rede ou fiquem disponíveis no servidor externo.

A comunicação interna a residência, entre a Central Local e os módulos, foi feita usando transmissores sem-fio com tecnologia Zigbee. Essa tecnologia permite uma rede local segura, já que possui criptografia e autenticação, e estável, transmissores possuem conexão dinâmica e podem usar full-mesh.

Por fim, os dispositivos que foram integrados ao sistema (Iluminação, Janela, TV, Ar Condicionado, Aparelho de Som, etc.) são ligados diretamente aos módulos, ou de forma sem-fio, no caso do módulo IR (infravermelho).

5.2 Arquitetura de Controle

A Arquitetura de Controle utilizada é parte centralizada e parte descentralizada, ou seja, parte do controle fica a cargo dos módulos e parte fica a cargo da central. A figura 5.2 mostra como a arquitetura está organizada.

Os módulos possuem funcionalidades pré-definidas e rotinas de controle parametrizáveis. Essas rotinas de controle e supervisão permitem os módulos controlar funções simples e com abrangência local, uma vez que cada módulo só tem acesso aos dispositivos a ele conectado.

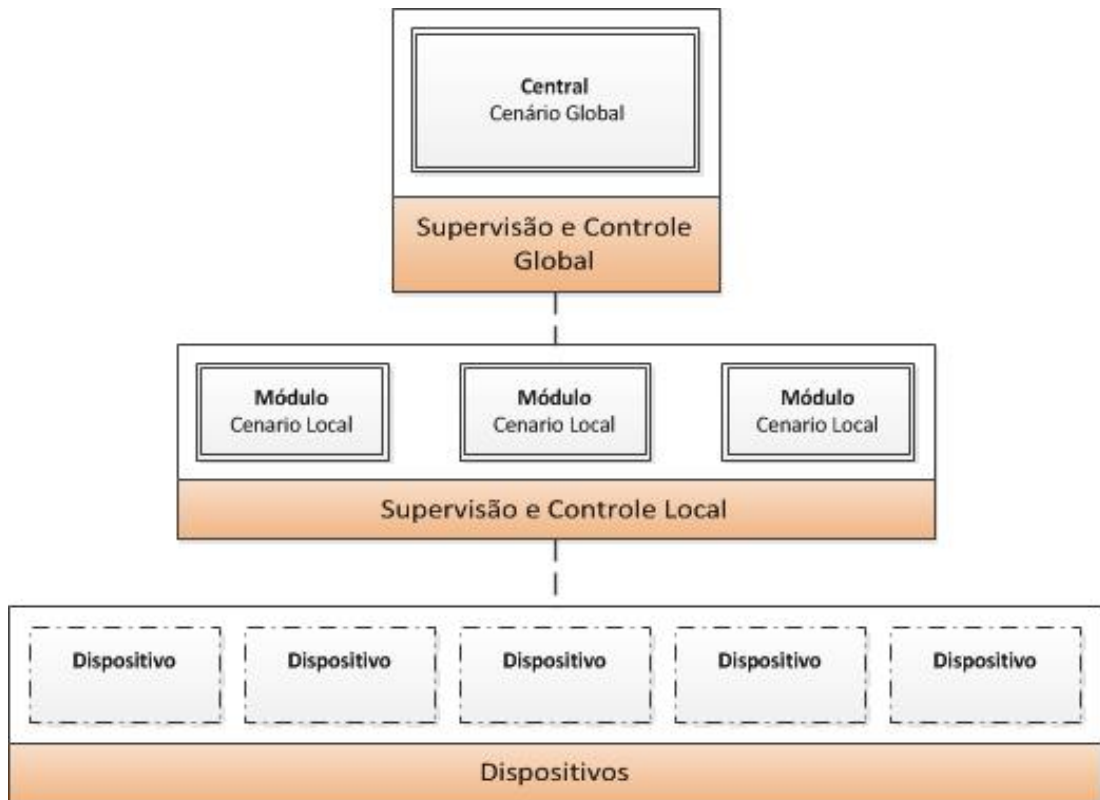


Figura 5.2: Arquitetura de Controle

As rotinas de controle e supervisão que incluem dispositivos de diferentes módulos, por exemplo, regras de disparo entre o módulo de iluminação e o módulo de sensoramento, ficam a cargo da Central. Além disso, regras que envolvam agendamento de tarefas, também ficam a cargo da central uma vez que os módulos não tem hardware para manter real time clock.

Dessa forma foi possível diminuir a quantidade de dados que trafega na rede. Além disso, caso a rede local pare de funcionar os dispositivos ainda podem operar de forma local, não deixando assim o sistema totalmente inoperante.

5.3 Central Local

A central local foi implementada utilizando um Raspberry Pi ^[16] Modelo B, um computador com dimensões próximas a de um cartão de crédito e com grande parte do hardware integrado. Na tabela 5.1 são mostradas as especificações do modelo utilizado.

Nele foi instalada a versão Debian Wheezy do Linux otimizada para o Raspberry Pi. Para manter as rotinas de controle e supervisão e a interface da aplicação foi instalado o pacote LAMP (Linux+Apache ^[17]+MySQL ^[18]+PHP ^[19]) no servidor. O Apache (ver. 2.0) é responsável por manter o serviço HTTP no servidor.

Raspberry Pi Modelo B	
Preço:	US\$35,00
SoC:	Broadcom BCM2835 (CPU, GPU, DSP, e SDRAM)
CPU:	ARM11 700 MHz
GPU:	Broadcom VideoCore IV, 12 OpenGL ES 2.0, 1080p30 decodificador h.264/MPEG-4 AVC7
Memória (SDRAM):	512 MB (compartilhada com GPU)
Portas USB 2.0:	2
Saídas de vídeo:	RCA Composto, HDMI, Painéis LCD via DSI14 15
Saídas de áudio:	Conector de 3.5 mm, HDMI
Armazenamento onboard:	SD / MMC / slot para cartão SDIO
Rede onboard:	10/100 Ethernet (RJ45)10
Outras Conexões:	8 × GPIO, UART, I2C, SPI, +3.3 V, +5 V, Terra
Power ratings:	700 mA (3.5 W)
Fonte de energia:	5V
Tamanho:	85,60 mm x 56 mm x 21mm

Tabela 5.1: Especificações Raspberry Pi Modelo B

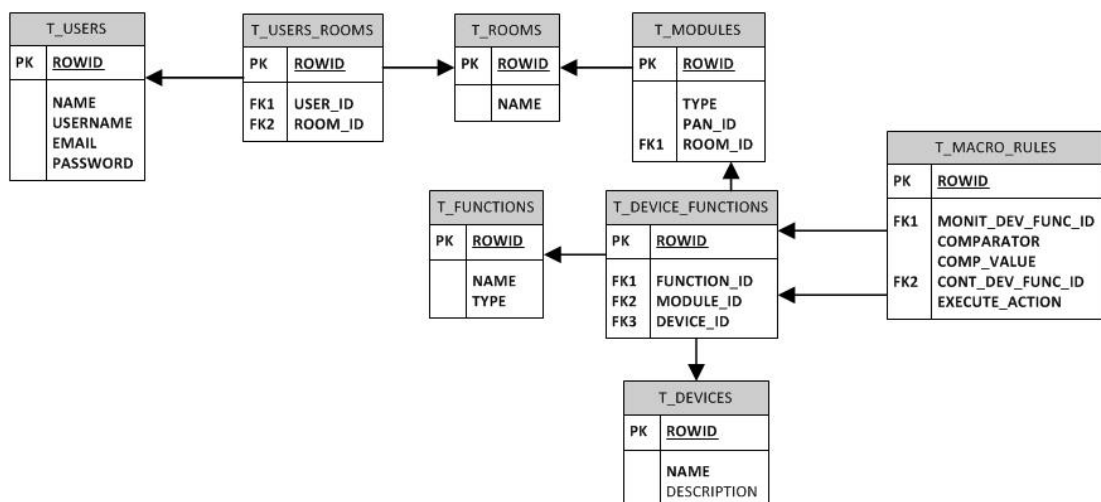


Figura 5.3: Modelo de Entidade Relacional do Banco de Dados

O PHP (ver. 5.4.4) é responsável pelos scripts do lado do servidor, mantém a interface, roda regras de controle e supervisão descentralizadas e mantém o serviço de sincronismo entre a Central Local e a Remota. O MySQL é responsável pelo armazenamento de dados relativos aos módulos e ao sistema.

Todos dispositivos do sistema foram cadastrados no banco de dados, bem como suas funções e seus estados. Foram cadastrados também os acessos de cada usuário, onde foi possível inserir as regras de acesso necessárias à residência. Na figura 5.3 pode ser visto um esboço do modelo de entidade relacional do banco de dados que será implementado.

A implementação através de um banco de dados, utilizado para o cadastro completo dos dispositivos da residência, da escalabilidade ao sistema, uma vez que

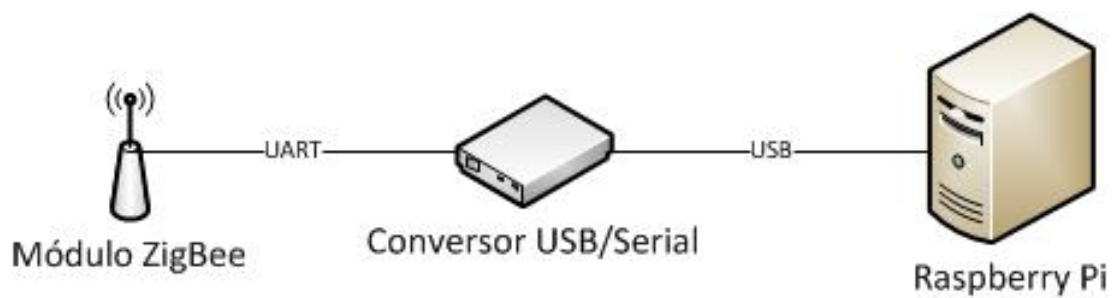


Figura 5.4: Ligação Raspberry Pi - Módulo Zigbee

para inserir novos dispositivos é necessário somente incluir os dados necessários no banco de dados, não sendo necessária a intervenção em software.

A utilização de uma plataforma web permitiu uma interface multiplataforma, conforme os requisitos do projeto, podendo assim ser acessada de computadores, tablets, celulares, etc. Durante o desenvolvimento da interface optamos sempre por utilizar soluções que funcionem em diversos dispositivos.

Nessa interface os cômodos foram divididos em módulos, onde são listados seus dispositivos controláveis com suas funções correspondentes. A criação das telas é dinâmica e montada a partir das nas informações cadastradas no banco de dados.

A Central pode ser ligada a rede Wi-Fi ou a rede Ethernet da casa e é por essa rede que a Interface fica disponível aos usuários da residência. Na comunicação entre a central e o seu módulo ZigBee foi utilizado um conversor USB/Serial, na figura 5.4 vemos um diagrama de como foi feita a ligação. Na questão de software, para realizar a integração do módulo serial ao sistema, foi utilizada a biblioteca php ^[20] que permite o acesso dos scripts (PHP) à porta serial, para que os comandos sejam enviados e os estados dos sensores sejam recebidos.

5.4 Central Nuvem

A Central na Nuvem permite ao usuário controlar as funcionalidades de sua residência remotamente, através de uma conexão a internet. A principal preocupação da Central na Nuvem foi com a segurança, já que essa interface está disponível na internet.

Nessa Central, a estrutura do banco de dados é exatamente a mesma do servidor local, a diferença são os dados que estão totalmente criptografados. Isso garante maior segurança, pois mesmo com o acesso ao servidor, e ao banco de dados, não será possível controlar a residência.



Figura 5.5: Tela de Login

A interface desse servidor é estruturalmente igual a do servidor local, trazendo assim os dados criptografados. Após a montagem das páginas uma biblioteca Javascript ^[21] irá decriptar os dados. Esse mesmo pacote será responsável por encriptar os dados novamente durante as requisições ao servidor. Assim somente dados criptografados serão transferidos durante as requisições, aumentando assim a segurança do sistema.

O servidor na nuvem foi implementada utilizando o serviço EC2 do pacote AWS da Amazon. Esse serviço permite criar um servidor cloud redimensionável e de fácil gerenciamento. Nessa central foi instalado o sistema operacional Ubuntu Server 12.04.2 LTS. A aplicação web na Nuvem é mantida, também, através do pacote LAMP.

5.5 Interface

Optamos por uma interface dividida em módulos, que representam cada um dos ambientes da residência. A figura 5.5 mostra a tela de login do sistema.

Conforme explicado anteriormente, a interface é montada a partir das informações cadastradas no banco de dados. Na figura 5.6 é mostrada a tela inicial do sistema, acessada após o login do usuário.

No menu superior são mostrados os cômodos da residência, além da opção de acesso a página inicial. Ao clicar em um dos cômodos, a parte inferior da

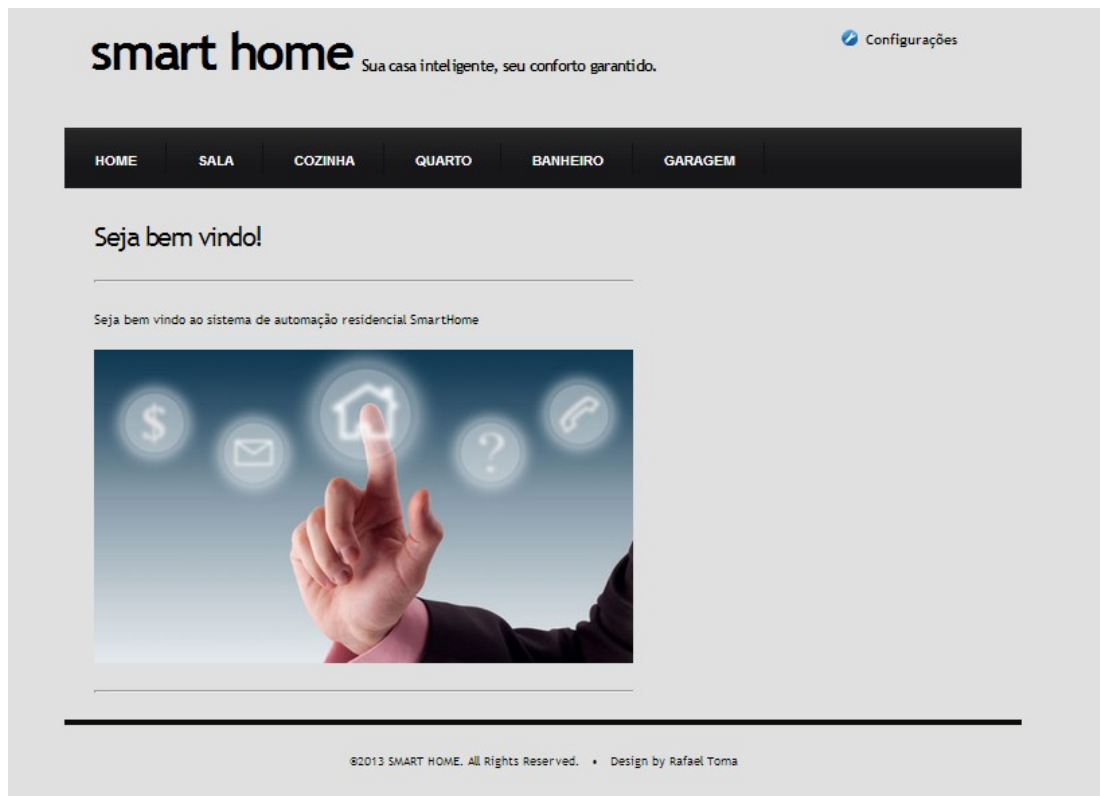


Figura 5.6: Tela Inicial

página passa a mostrar as opções referentes a esse ambiente, conforme mostra a figura 5.7. Na parte esquerda é mostrada uma foto do ambiente e algumas informações rápidas (temperatura e humidade) e na direita um menu com as dispositivos do cômodos.

5.6 Rede Local

A rede local, entre módulos e a central local, foi desenvolvida utilizando a tecnologia Zigbee. Utilizamos no sistema módulos Xbee S2 ^[22] fabricados pela Digi International. Na central o módulo Xbee respectivo foi configurado para funcionar como Coordenador da PAN, e nos módulos seus Xbees foram configurados como Routers. Todos os módulos foram configurados para trabalhar no modo API. Na figura 5.8 são mostradas as especificações do modelo utilizado.

5.7 Módulos Desenvolvidos

5.7.1 Módulo de umidade e temperatura

Esse módulo tem a finalidade de informar a central e ao usuário a temperatura ambiente e a umidade relativa do ar. Com essas informações o usuário poderá

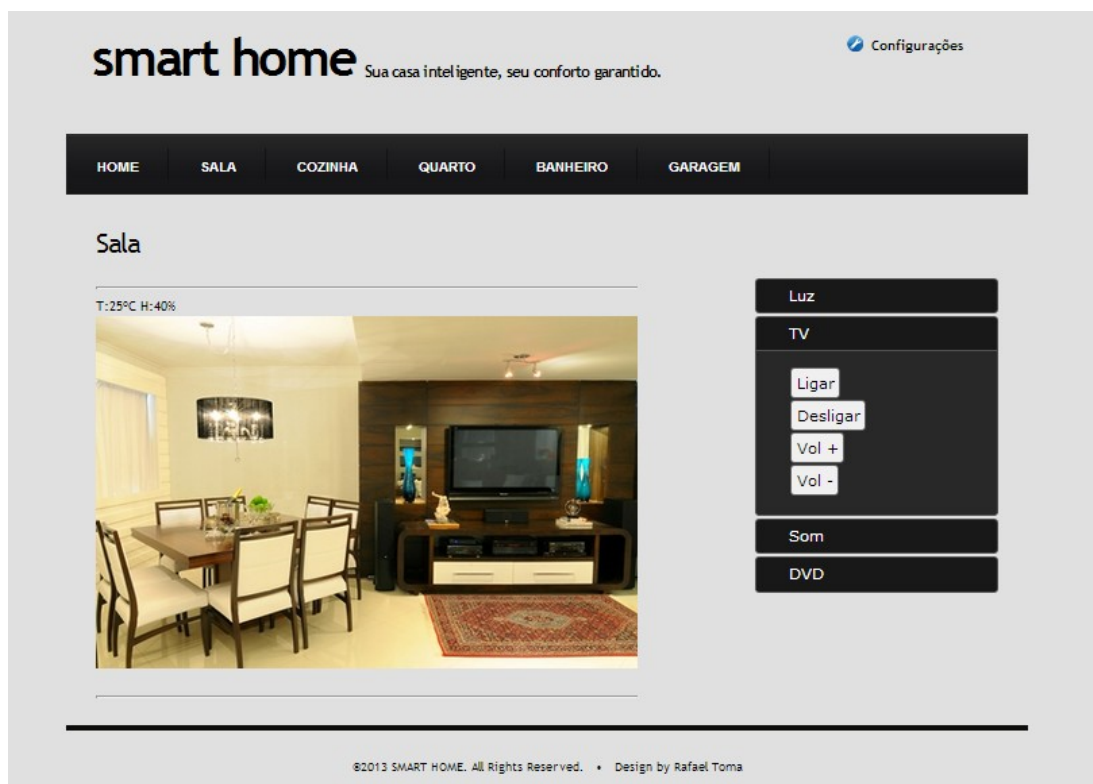


Figura 5.7: Tela Sala

Tensão nominal	2,1 a 3,6V
Corrente nominal	40mA a 3,3V
Alcance em linha reta	120m
Portas digitais	11
Portas analógicas	4
Frequência	2,4GHz
Temperatura de trabalho	-40 a 85°C

Figura 5.8: Tela Sala

Item	Faixa de trabalho	Acurácia	Resolução
Umidade	20-90%	5%	1
Temperatura	0-50 °C	2 °C	1

Tabela 5.2: Especificações Operacionais

	Condição	Mínimo	Normal	Máximo
Tensão	DC	3V (a 25°C)	5V (a 25°C)	5,5V (a 25°C)
Corrente	Em medição	0,5mA	0,5mA	2,5mA
Corrente	Em standby	100uA	100uA	150uA

Tabela 5.3: Especificações Elétricas

saber se o sistema de aquecimento/ar condicionado deverá ser acionado a distância para que quando ele retorne para casa a temperatura ambiente já esteja agradável.

Em caso de controle automático, a central, em conjunto com o módulo de controle remoto, poderá acionar os controles necessários para se chegar condições de conforto térmico que usuário deseja.

Para fornecer as informações necessárias para o módulo de umidade e temperatura foi utilizado o sensor DHT11. Ele é fabricado pela DFRobot ^[23] e é constituído por uma fita resistiva para medir a umidade, um componente ntc para detectar a temperatura e um microcontrolador de 8 bits que tem como saída um sinal digital. Essa saída digital possui um frame de 40 bits. Os 16 primeiros são referentes a umidade. Depois são 16 bits referentes a temperatura e os últimos 8 bits são o checksum.

Os valores da faixa de temperatura, umidade e o erro estão na tabela 5.2. O sensor possui 3 pinos, onde há o VCC e o GND e um terceiro que é o sinal digital fornecido pelo microcontrolador. Na tabela 5.3 são mostradas as especificações elétricas do sensor.

5.7.2 Módulo de segurança

A finalidade desse módulo é a proteção passiva do usuário. Através dos sensores de gases combustíveis e fumaça, a central poderá monitorar possíveis vazamentos de GLP na cozinha ou focos de incêndios que podem ocorrer na casa. Com esses dados a central poderá avisar o usuário através do acesso remoto e, em conjunto com outros módulos, poderá executar as ações como abrir as janelas para dissipar

Tensão	5V
Consumo	<900mW
Faixa de concentração	300-10000ppm
Gases de detecção	Butano, propano, GLP, gás natural
Condições normais (temp. e umidade)	20 °C e 65% RH

Tabela 5.4: Especificações MQ6 ^[24]

Tensão	5V
Consumo	<900mW
Faixa de concentração	300-10000ppm
Gases de detecção	Butano, metano, propano, GLP, gás natural, álcool e fumaça
Condições normais (temp. e umidade)	20°C e 65% RH

Tabela 5.5: Especificações MQ2 ^[25]

os gases e acionar os bombeiros automaticamente.

Para esse módulo é utilizado o sensor MQ6 ^[24] para os gases combustível propano, butano, GLP (gás liquefeito de petróleo) e gás natural. O material sensetivo é o SnO₂(óxido de estanho).

O sensor possui 6 pinos de conexão sendo que são 3 de Vcc, 1 de Gnd e 2 de saída. Os valores de saída precisam ser somados para que se tenha o sinal analógico com a detecção do gás combustível. Os valores de alimentação e faixa de trabalho estão na tabela 5.4.

Outro sensor utilizado nesse módulo é o MQ2 (tabela 5.5), que possui funcionamento similar ao sensor MQ6. Além dos gases propano e butano, este sensor detecta metano, gases de álcool e fumaça. Ambos possuem 6 pinos e tem saída analógica para detecção de fumaça. A diferença entre eles é o material de detecção.

5.7.3 Módulo de banheira

Através do acesso remoto, o usuário será capaz de acionar a distância o enchimento da banheira. Assim, ao chegar em sua residência, a banheira estará pronta para o uso. A temperatura será aferida por um sensor de temperatura da água e o acionamento será feito por dois servo motores capazes de abrir as torneiras de água fria e água quente.

A detecção de temperatura da água da banheira é feita utilizado o sensor DS18B20 (tabela 5.6). Ele possui 3 pinos, sendo dois de alimentação e outro para

Tensão	3,3V a 5V
Faixa de trabalho	-55°C a +125°C
Resolução	±0,5°C
Tempo de resposta	750ms

Tabela 5.6: Especificações DS18B20 ^[26]

comunicação digital. Este sensor utiliza o protocolo chamado One Wire, onde a comunicação com o microcontrolador é feita por apenas uma porta digital. Através dessa porta, o sensor envia um frame de até 12bits referente a temperatura. Para a leitura desse protocolo há a necessidade de utilizar uma biblioteca específica para comunicação com o Atmega.

5.7.4 Módulo de iluminação

Este módulo terá a finalidade de acionar a iluminação de um ambiente, assim como detectar a intensidade luminosa natural. O sensor de intensidade luminosa terá a função de fornecer o status da iluminação do ambiente. Com isso, será possível que esse ambiente tenha a iluminação artificial acionada automaticamente na ausência de luz natural.

O acionamento da iluminação será feito através de um relé, interligado ao Atmega e ao Zigbee. Através do acesso remoto, o usuário poderá ter o controle à distância da iluminação artificial, ou em caso de controle automático, a central, em conjunto com o sensor de intensidade luminosa, poderá decidir quando as luzes ambientes deverão ser ligadas.

Neste módulo utilizaremos o sensor LDR (Light Dependent Resistor) para detectar a intensidade luminosa do ambiente. Este sensor é um componente passivo e é constituído de um material semicondutor cuja resistência varia com a intensidade luminosa. O modelo adquirido é o PGM5506 da Token cujas características estão na tabela 5.7.

5.7.5 Módulo de janela e chuva

Este módulo tem a capacidade de fornecer a central e ao usuário informações sobre as condições climáticas referente a chuva. Além disso, este módulo tem controle sobre a abertura ou fechamento da janela através de um servo motor acoplado a janela.

Vmax	100 (V)
Pmax	90 mW
Temperatura de trabalho	-30°C a +70°C
Resistência em ambiente escuro	0,15 MΩ (min)
Resistência em ambiente claro	5 KΩ (max)
Tempo de resposta médio	35 ms

Tabela 5.7: Especificações LDR

Corrente permitida	100 mA
Ângulo de instalação	Entre 30° e 70°
Tensão de operação	3,3V a 5V

Tabela 5.8: Especificações FR-04 ^[27]

Com esse módulo o usuário poderá abrir ou fechar a janela à distância através do seu acesso remoto. Isso permite a ele, por exemplo, abrir a janela pouco antes de retornar a sua residência para ter um ambiente arejado ao chegar a casa.

Outra funcionalidade é o modo automático de fechamento da janela. Com essa opção habilitada, o sensor de chuva fornecerá informações para que o módulo feche, de forma automática, a janela em caso de precipitação.

Para detecção de chuva foi utilizado o sensor FR-04 (tabela 5.8). Este sensor é formado por uma placa com dois materiais diferentes. Ao entrarem em contato com a água, estes materiais passam a ser condutores.

Além da alimentação VCC e GND, o sensor possui duas saídas, uma analógica e outra saída com sinal TTL.

5.7.6 Módulo de controle remoto

Este módulo terá a finalidade de receber e enviar sinais de controle remoto para qualquer periférico com essa opção. Por exemplo, em conjunto com o módulo de temperatura e umidade, a central poderá acionar o ar condicionado automaticamente para preservar o conforto térmico. Também será possível ter controle sobre a televisão, blu-ray e home-theater através de um único controle remoto universal, disponível nos acessos remotos. Dessa forma, será possível trocar todos os controles remotos dos diversos periféricos por apenas um.

Tensão	5-20V
Corrente	65µA
Ângulo de detecção	120°
Distância máxima	7m

Tabela 5.9: Especificações Sensor de IR O52116

Alimentação	5V
Consumo	100mA
Consumo máximo	1A
Potência máxima	150mW
Temperatura de operação	-40° a +85°

Tabela 5.10: Especificações IR333 ^[28]

Como sensor de detecção, utilizamos o sensor de IR modelo O52116. As características dele estão na tabela 5.9.

Para emissão do sinal de IR, utilizamos o led IR333 da Everlight (tabela 5.10).

5.8 Sequência de Funcionamento

5.8.1 Usuário Local Ativa Função de Dispositivo

Quando um usuário local interno a residência, ativa uma função de um dos dispositivos inteligentes, na interface de controle do sistema, é feita uma requisição ao script responsável pela comunicação na central. Nessa requisição são enviados os dados dos da função, do dispositivo e do módulo relacionado a aquele dispositivo. O script trata as informações para gerar o frame de comando e envia-lo pela rede Zigbee. O módulo correspondente então recebe o frame de comando, interpreta e executa a função desejada.

5.8.2 Usuário Remoto Ativa Função de Dispositivo

Quando um usuário externo ativa uma função de um dos dispositivos também feita uma requisição ao script responsável pela comunicação na central. Nessa requisição também são enviados os dados dos da função, do dispositivo e do módulo relacionado a aquele dispositivo, mas essas informações são criptografadas por meio do Javascript antes do envio. Essas informações ficam então armazenadas esperando uma requisição de sincronismo. A cada cinco segundos a Central Local roda a rotina de sincronismo, onde são capturados os dados dos dispositivos que foram ativados remotamente. Os dados então são decriptados e tratados para

gerar o frame de comando e envia-lo pela rede Zigbee. O módulo correspondente recebem o frame de comando, interpreta e executa a função desejada.

6 Testes Realizados

6.1 Modulo Iluminação

Para o módulo de iluminação foram feitos duas provas de conceito, a primeira para testar os sensores de luminosidade e o segundo para testar um controle lógico para a dimerização de uma lanterna de LED.

6.1.1 Sensor de Luminosidade

O sensor LDR varia a sua resistência dependendo da luminosidade a que está exposto. Para os testes utilizamos o circuito mostrado na figura 6.1, em que o sensor foi colocado em série com um resistor de 10 kohms e a tensão foi medida no ponto central que funciona como um divisor de tensão proporcional à resistência do sensor.

6.1.2 Dimerização

Para a dimerização de um led utilizamos a função PWM do arduino. O circuito serve apenas para provar o conceito lógico do circuito, já que em lâmpadas residenciais a potência e tensões utilizadas são bem maiores que a do teste realizado. O circuito se comportou conforme o esperado variando bem a luminosidade a

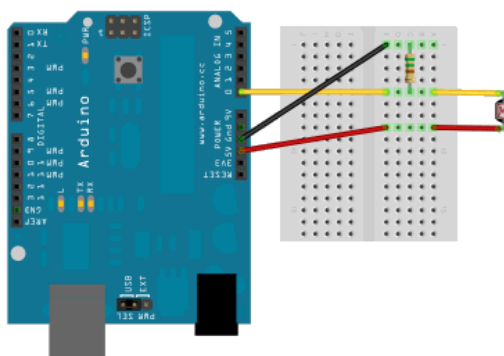


Figura 6.1: Circuito LDR (Reproduzido de [29])

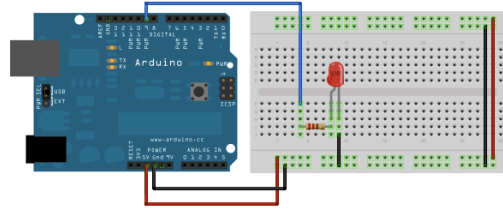


Figura 6.2: Circuito Dimmer (Reproduzido de ^[29])

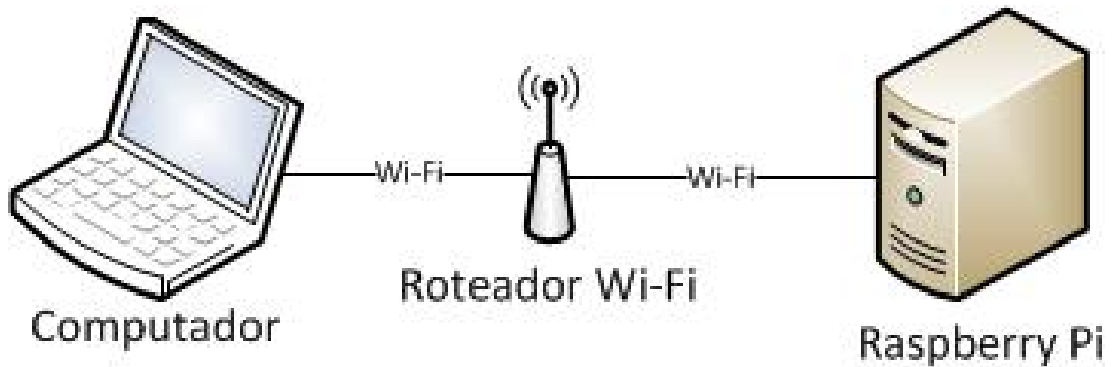


Figura 6.3: Ligação Raspberry Pi - Módulo Zigbee

medida que o sinal do PWM era modificado. A frequência do sinal de PWM gerado pelo arduino é de 31250 Hz. a figura 6.2 mostra o circuito utilizado.

6.2 Tempos de Carregamento da Interface

A fim de avaliar o desempenho do Servidor Local, que roda em um Raspberry Pi, utilizamos o Developer Tools do navegador Google Chrome ^[22] para medir os tempos de carregamento e requisições da interface.

O teste foi realizado usando uma rede Wi-Fi conforme a estrutura mostrada na figura 6.3. O roteador utilizado foi um TP-Link modelo TL-WR941ND. Para os testes o cache do navegador foi apagado para que todos os arquivos necessários sejam baixados no momento do teste.

O primeiro teste de carregamento foi feito na tela de login do sistema. A figura 6.4 é mostra os resultados do teste. É possível verificar os tempos de carregamento dos arquivos necessários para o carregamento da tela, que leva 247 milissegundos para ser carregada, com um total de 16,4 KB de dados.

O segundo teste foi realizado no carregamento da página inicial do sistema, que é carregada logo após a tela de login, a medição leva em conta o tempo entre o clique no botão de entrar até o carregamento completo da página inicial. A figura 6.5 é mostra os resultados do teste. É possível verificar o carregamento de

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
192.168.1.105	GET	200 OK	text/html	Other	826 B 963 B	140 ms 140 ms	
index.css	GET	200 OK	text/css	192.168.1.105/6 Parser	521 B 355 B	13 ms 13 ms	
logoPolii.jpg	GET	200 OK	image/jpeg	192.168.1.105/10 Parser	15.0 KB 14.8 KB	51 ms 30 ms	
3 requests 16.4 KB transferred 245 ms (load: 247 ms, DOMContentLoaded: 194 ms)							

Figura 6.4: Teste Login

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
app.css	GET	200 OK	text/css	application.php:5 Parser	1.8 KB 5.5 KB	39 ms 38 ms	
application.js	GET	200 OK	application/javas...	application.php:11 Parser	837 B 991 B	148 ms 147 ms	
application.php	GET	200 OK	text/html	http://192.168.1.105... Redirect	1.3 KB 2.0 KB	20 ms 20 ms	
home.jpg	GET	200 OK	image/jpeg	application.php:38 Parser	16.8 KB 16.5 KB	453 ms 440 ms	
img04.gif	GET	200 OK	image/gif	application.php:25 Parser	2.2 KB 2.0 KB	108 ms 106 ms	
img11.gif	GET	200 OK	image/gif	application.php:51 Parser	413 B 124 B	129 ms 128 ms	
img12.jpg	GET	200 OK	image/jpeg	application.php:13 Parser	1.0 KB 750 B	102 ms 101 ms	
img14.jpg	GET	200 OK	image/jpeg	application.php:15 Parser	133 KB 133 KB	324 ms 106 ms	
img15.gif	GET	200 OK	image/gif	application.php:27 Parser	399 B 110 B	117 ms 115 ms	
jquery-1.9.1.js	GET	200 OK	application/javas...	application.php:7 Parser	78.4 KB 262 KB	1.05 s 733 ms	
jquery-ui.css	GET	200 OK	text/css	application.php:6 Parser	6.3 KB 31.1 KB	148 ms 146 ms	
jquery-ui.js	GET	200 OK	application/javas...	application.php:8 Parser	105 KB 426 KB	1.49 s 784 ms	
jquery.ui.touch...	GET	200 OK	application/javas...	application.php:9 Parser	937 B 1.2 KB	148 ms 148 ms	
login.php	POST	302 Found	text/html	Other	471 B 0 B	86 ms 85 ms	
settings.png	GET	200 OK	image/png	application.php:21 Parser	1.2 KB 914 B	107 ms 106 ms	
15 requests 350 KB transferred 1.99 s (load: 1.99 s, DOMContentLoaded: 1.71 s)							

Figura 6.5: Teste Tela Inicial

todos os pacotes necessários para rodar a aplicação, as imagens, etc. O tempo total de carregamento da página foi de 1,99 segundos com um total de 350 KB de dados.

O terceiro teste foi realizado na chamada na abertura da sala, onde são carregados os dispositivos e suas funcionalidades, e a figura relacionada ao cômodo. O resultado do teste pode ser visto na figura 6.6, a requisição levou 504 milissegundo.

O quarto teste foi realizado na chamada de uma das funções dos dispositivos,

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
roomsSidebar....	POST	200 OK	text/html	jquery-1.9.1.js:8526 Script	490 B 1.3 KB	428 ms 427 ms	
roomsContent....	POST	200 OK	text/html	jquery-1.9.1.js:8526 Script	407 B 151 B	401 ms 400 ms	
sala.jpg	GET	200 OK	image/jpeg	jquery-1.9.1.js:6063 Script	78.7 KB 78.4 KB	114 ms 10 ms	
3 requests 79.6 KB transferred							

Figura 6.6: Teste Carregamento Sala



Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline	400 ms	600 ms
 arduino.php?PH...	POST	200 OK	text/html	<u>jquery-1.9.1.js:8526</u> Script	296 B 5 B	719 ms 719 ms			
1 requests 296 B transferred									

Figura 6.7: Teste Função

onde é chamado o script responsável pela comunicação com o módulo ZigBee. O resultado do teste pode ser visto na figura 6.7, a requisição levou 719 milissegundo.

7 Conclusões

O desenvolvimento do projeto pode ser descrito em 3 etapas. Na primeira parte foi feito um estudo abrangente sobre a domótica. A partir disso foram definidos os requisitos e as diretrizes do projeto. Em seguida, escolhemos as soluções que melhor atendiam os requisitos definidos anteriormente.

Durante a pesquisa pudemos entender melhor a relação entre homem-máquina, homem-residência e as diversas formas de melhorar a interatividade entre esses sujeitos. O estudo nos permitiu identificar as barreiras tecnológicas envolvidas no desenvolvimento de um sistema de automação residencial. Entre elas a necessidade de uma rede estável e segura para comunicação entre dispositivos, de uma interface simples e intuitiva, de um sistema de fácil implementação e retro compatível.

No segundo momento definimos as diretrizes do projeto. Optamos então por desenvolver um sistema de baixo custo, escalável, flexível e que permita acesso remoto as funcionalidades da residência. Assim decidimos utilizar uma arquitetura de controle parte distribuída, parte centralizada, que permitiu diminuir o custo computacional da central, diminuir o tráfego de dados na rede e melhorar a confiabilidade do sistema.

Na etapa final, desenvolvemos os protótipos das centrais e dos módulos que permitiram realizar testes, que comprovaram a viabilidade da arquitetura proposta.

Referências

- 1 BOLZANI, C. A. M. *Residências Inteligentes*. [S.l.]: Editora Livraria da Física, 2004.
- 2 YUN, J. H. nd J.; JANG, J.; ; PARK, K. R. User-friendly home automation based on 3d virtual world. *IEEE Transactions on Consumer Electronics*, v. 56, n. 3, August 2010.
- 3 SU, J.; LEE, C.; WU, W. The design and implementation of a low-cost and programmable home automation module. *IEEE Transactions on Consumer Electronics*, v. 52, n. 4, November 2006.
- 4 GUPTA, G. S.; ALSHU'EILI, H.; MUKHOPADHYAY, S. Voice recognition based wireless home automation system. In: *Proceedings of the IEEE International Conference on Mechatronics (ICM 2011)*. [S.l.: s.n.], 2011.
- 5 NUNES, R. J. C. Decentralized supervision for home automation. In: *Proceedings of the IEEE Mediterranean Electrotechnical Conference (MELECON 2006)*. [S.l.: s.n.], 2006.
- 6 WU, C. L.; WANG, W. C.; FU, L. C. Mobile agent based integrated control architecture for home automation system. In: *Proceedings of the International Conference on Intelligent Robots System*. [S.l.: s.n.], 2004.
- 7 ALMEIDA, A. V. de. *Implementação de um sistema de automação residencial modular sem fio: Modulo Periférico*. Dissertacao (Graduação) — Escola de Engenharia de São Carlos da Universidade de São Paulo, 2009.
- 8 BOLZANI, C. A. M. *Análise de arquiteturas e desenvolvimento de uma plataforma para residências inteligentes*. Tese (Doutorado) — Escola Politécnica da Universidade de São Paulo, 2010.
- 9 DANGE, H. V.; GONDI, V. K. Powerline communication based home automation and electricity distribution system. In: *Proceedings of the Process Automation, Control and Computing (PACC 2011)*. [S.l.: s.n.], 2011.
- 10 GOMEZ, C.; PARADELLS, J. Wireless home automation networks: A survey of archetectures and technologies. *IEEE Communications Magazine*, v. 48, 2010.
- 11 AES. *Site da Diciplica de Redes I da Universidade Federal do Rio de Janeiro*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <http://www.gta.ufrj.br/grad/05_2/aes/>.
- 12 YUEJUN, Z.; MINGGUANG, W. Design of wireless remote module in x-10 intelligent home. In: *Proceedings of the IEEE International Conference on Industrial Technology (ICIT 2005)*. [S.l.: s.n.], 2005.

- 13 ALLIANCE, Z. *Site do Zigbee Alliance*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.zigbee.org/>>.
- 14 HUSNEY, J. *802.15.4 and ZigBee Mesh Networking*. [S.l.]. Vídeo online, último acesso em novembro de 2013. Disponível em: <<http://www.digi.com-/learningcenter/video/player?id=802154-and-zigbee-mesh-networking>>.
- 15 NETWORKS, W. S. *Site Wireless Sensor Networks*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.sensor-networks.org-/index.php?page=0903503549>>.
- 16 PI, R. *Site do Raspberry Pi*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.raspberrypi.org/>>.
- 17 APACHE. *Site do Apache*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.apache.org/>>.
- 18 MYSQL. *Site do MySQL*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.mysql.com/>>.
- 19 PHP. *Site do PHP*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://php.net/>>.
- 20 CLASS, P. S. *Site do PHP Serial Class*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.phpclasses.org/browse/file-/17926.html>>.
- 21 LIBRARY, S. J. C. *Site do Stanford Javascript Crypto Library*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://bitwiseshiftleft.github.io/sjcl/>>.
- 22 CHROME, G. *Site do Google Chrome*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.google.com/intl/pt-BR/chrome/>>.
- 23 D-ROBOTICS. *D-Robotics*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.droboticsonline.com/>>.
- 24 MQ6, D. *Datasheet MQ6*. [S.l.]. Datasheet MQ6, último acesso em novembro de 2013. Disponível em: <<https://www.sparkfun.com/datasheets/Sensors-/Biometric/MQ-6.pdf>>.
- 25 MQ2, D. *Datasheet MQ2*. [S.l.]. Datasheet MQ2, último acesso em novembro de 2013. Disponível em: <<http://www.pololu.com/file/0J309/MQ2.pdf>>.
- 26 DS18B20, D. *Datasheet DS18B20*. [S.l.]. Datasheet DS18B20, último acesso em novembro de 2013. Disponível em: <<http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>.
- 27 CHUVA, M. S. de. *Manual Sensor de Chuva*. [S.l.]. Manual do Sensor de Chuva do Lab de Garagem, último acesso em novembro de 2013. Disponível em: <http://www.labdegaragem.com.br/loja/sensor_de_chuva.pdf>.
- 28 IR333, D. *Datasheet IR333*. [S.l.]. Datasheet IR333, último acesso em novembro de 2013. Disponível em: <http://www.everlight.com/datasheets-/IR333_H0_L10_datasheet.pdf>.

- 29 ARDUINO. *Site do Arduino*. [S.l.]. Website, último acesso em novembro de 2013. Disponível em: <<http://www.arduino.cc>>.