Índice

[O Problema 3](#_Toc355873604)

[Como Captar Sinais Wireless 4](#_Toc355873605)

[Como Interpretar Sinais Wireless 5](#_Toc355873606)

[Como Produzir Sinais Wireless 7](#_Toc355873607)

[Tornando tudo mais usável 9](#_Toc355873608)

[Apendice A: Código fonte para interligar o Raspberry PI com o emissor RX433 10](#_Toc355873609)

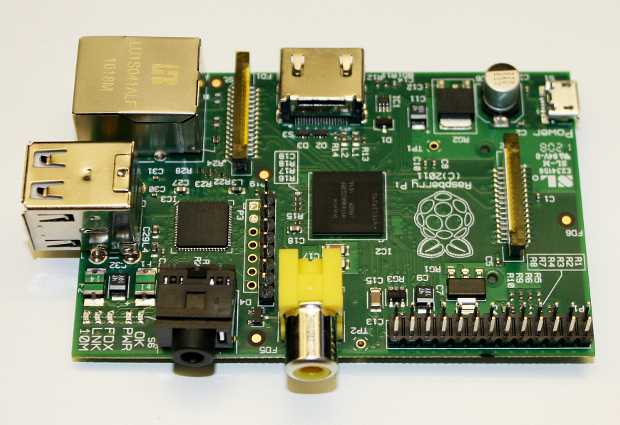
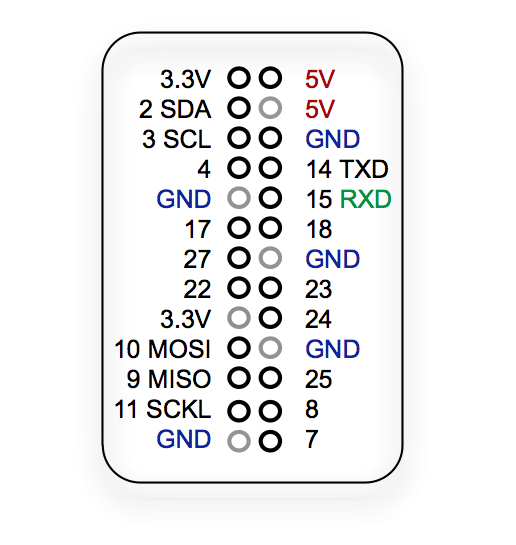
[Apendice B: Código Fonte Solução Web Raspberry Pi 14](#_Toc355873610)

# O Problema

Tudo começou com um pedido especial do meu Pai, que queria poder controlar o que se passa dentro de casa quando se ausenta nos fins de semana prolongados e nas férias.

Em casa tinha-mos instalado um sistema de automação controlado por comandos á distância que operam na frequência dos 433,92MHz. Com este sistema eramos capazes de controlar a iluminação, os estores e as tomadas elétricas.

O equipamento é semelhante ao que se encontra na imagem ao lado.

Como tinha um RaspberryPi a servir de media player, ligado ao televisor do meu quarto, pensei em usar os GPIO’s deste para controlar o sistema de automação cá de casa. O RaspberryPi é um minicomputador, criado para ensinar crianças a programar em Python. No entanto, tal equipamento tem um potencial fora do comum. Em primeiro lugar é barato, o meu, com portes e colocado em casa ficou por 39,16€; Em segundo lugar corre Linux de um cartão de memória SD, o que possibilita programar rapidamente e testar em tempo real; Em terceiro lugar dispõe de alguns pinos que podem ser programados como INPUT ou OUTPUT, o que possibilita interligar o aparelho com outros dispositivos. Como características deste belo aparelho temos, para o Modelo B, que é o modelo representado na foto, duas portas USB, uma porta LAN, uma porta HDMI, uma porta RCA, uma porta minijack, uma porta mini-USB usada para alimentação. Usa um processador BCM2835 que opera a uma frequência de 700Mhz, tem 512Mb de memória RAM partilhada com o GPU, entre outras características que fazem dele uma máquina “dos diabos”.

Em suma, pode ser comparado a um computador de meados dos anos 90, um Pentium II a 300Mhz, mas com as conveniências de ser do tamanho de um cartão de crédito, ser alimentado a 5 Volts e consumir apenas 3,5 Watts.

Tendo este cenário em mãos, como então interligar o RaspberryPi ao sistema de automação que possuía em casa?

O RaspberryPI possui alguns pinos que podem ser programados conforme as nossas necessidades, porque não usar estes pinos para controlar o sistema de automação, pensei eu. E pensei bem, só não imaginava a tarefa que estava para vir.

O pinout do RaspberryPi fica como referência, pois vai ser abordado mais adiante.

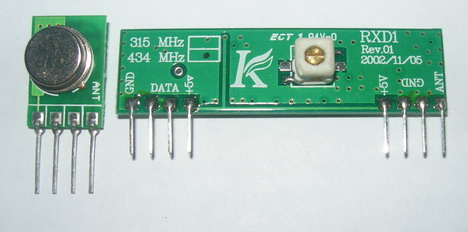
Dada esta pequena introdução, vamos descrever o trabalho completo de colocar o RaspberryPi a controlar qualquer dispositivo que opere na frequência dos 433,92Mhz.

# Como Captar Sinais Wireless

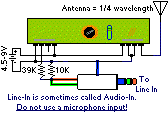
A primeira tarefa consistiu em perceber o que faz o comando á distancia. Para isso precisava entender o que dizia o comando ao receptor para este ligar e desligar a luz. Como o comando operava na frequência de 433.92Mhz, precisava, em primeiro lugar, de conseguir fazer uma leitura desta frequência.

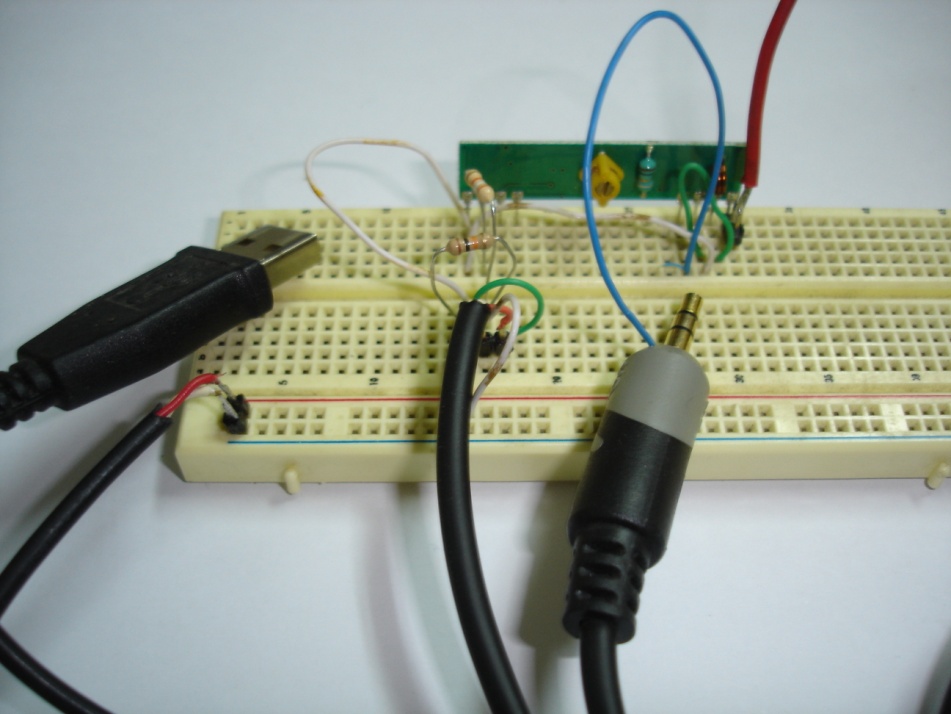
E como fazer isto, quando não se tem muito dinheiro e não se dispõe de equipamento adequado?

Depois de algumas horas de pesquisa na internet, encontrei na web alguém que tinha tido uma ideia semelhante e tinha conseguido com sucesso ler a frequência de 433.92Mhz com uma solução barata e acessível a qualquer um.

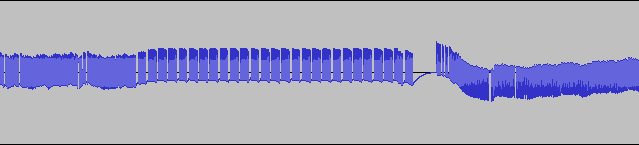
A ideia consiste em ligar um receptor da frequência 433.92Mhz ao line-in da placa de som do computador e usar um software de gravação áudio para visualizar o sinal que o receptor recebe.

Comprei um receptor RX 433, como o que se vê na figura ao lado. Segui o esquema encontrado na web e “voilá”, minutos depois estava a ver dados no meu computador.

Bem, o esquema exposto não tem muito a perceber. No meu caso, fui buscar a alimentação do circuito á porta usb do computador, a mesma tem uma tensão de 5Volts e 500mA, o que serve perfeitamente para alimentar o circuito. As resistências fazem um divisor de tensão, que não é nada mais nada menos do que baixar a voltagem do sinal, para que o mesmo não queime o line-in da placa de som. Fica uma foto da minha interpretação e montagem do do circuito exposto.



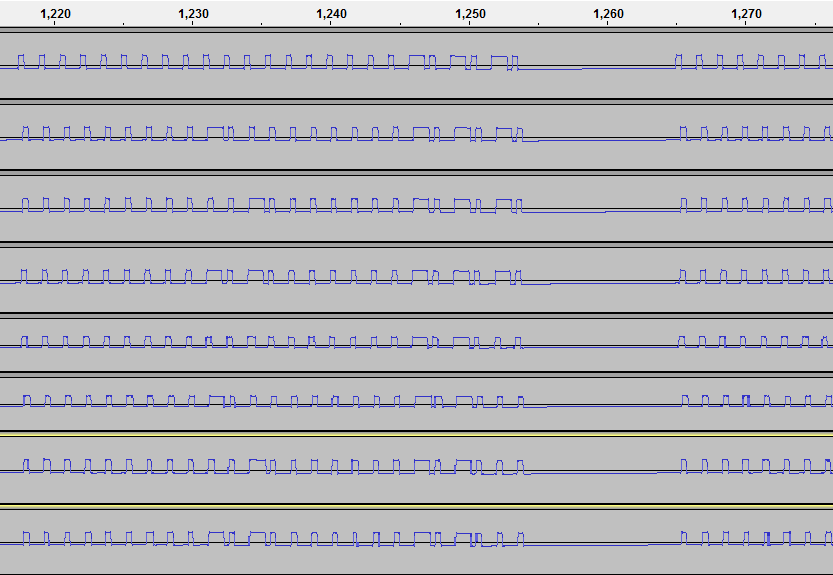
Abri o programa de gravação de áudio, coloquei-o a gravar a entrada line-in em MONO com uma taxa de amostragem de 44100Hz e fui pressionando os botões do comando. Comecei a notar que sempre que pressionava uma tecla o gráfico da gravação alterava a sua forma para uma forma constante. Brilhante, pensei eu, estou a ver qualquer coisa. Tudo estava a funcionar, estava a captar qualquer coisa vinda do comando, faltava saber o quê.



# Como Interpretar Sinais Wireless

Já me encontrava a captar sinais do comando, a seguir precisava de os saber interpretar, coisa que foi mais fácil do que pensava, em poucos minutos compreendia o comando todo.

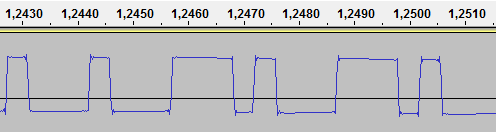
Bem, comecei por gravar as teclas 1 a 4 com a função ON e a seguir as teclas 1 a 4 com a função OFF. Fiz zoom à gravação, alinhei as coisas e comecei a desvendar o mistério.



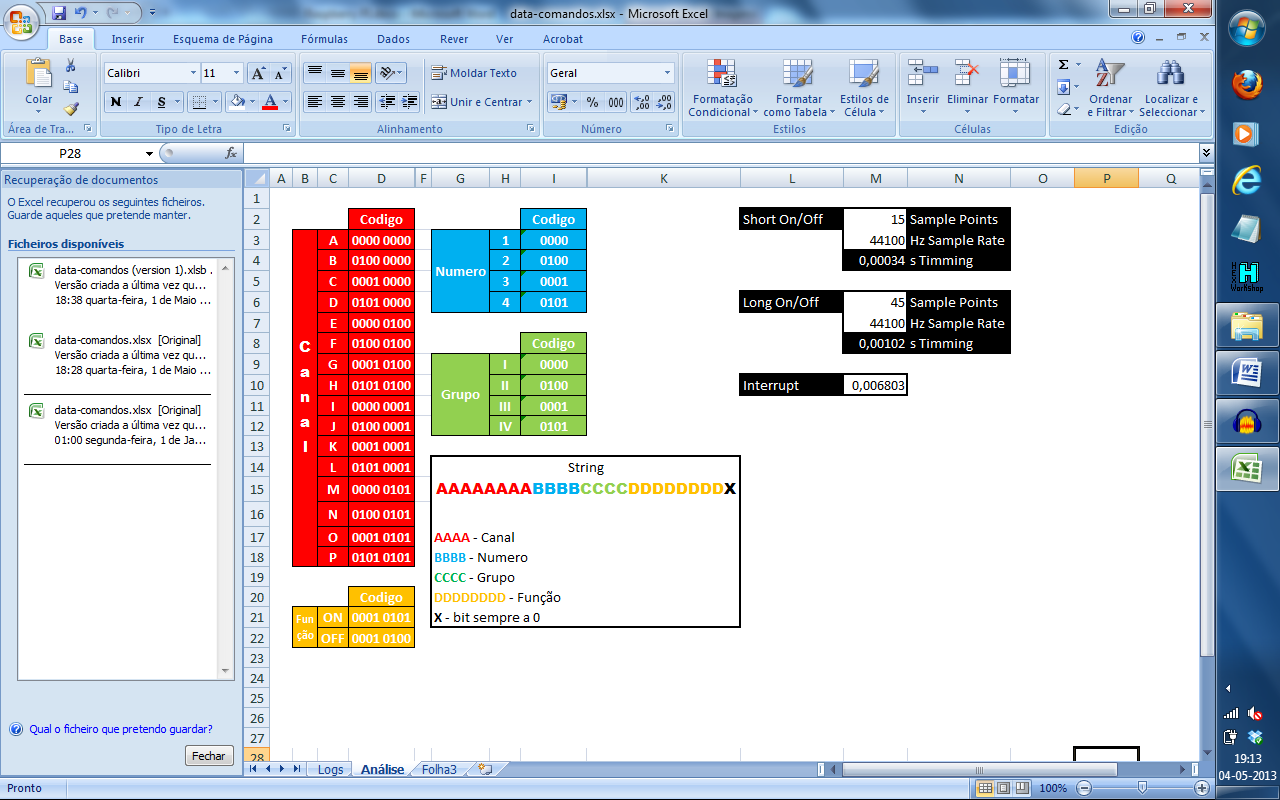
(Os sinais representados são de cima para baixo: 1 On; 2 On; 3 On; 4 On; 1 Off; 2 Off; 3 Off; 4 Off)

Percebi rapidamente que o sinal que recebia mudava de forma constante, e que cada botão enviava um sinal ligeiramente diferente. Daí para a frente, foi dividir, gravar, analisar, tomar notas, e repetir tudo novamente até ter certezas.

Reparem que o sinal é constituído por 25 impulsos, reparem também que os impulsos apresentam duas formas, uma mais longa, outra mais curta. A cada impulso curto, chamei de 0 (zero) e a cada impulso comprido chamei de 1 (um).



Fui transcrevendo os sinais captados em zeros e uns e após um par de horas tinha um alinhamento de zeros e uns que correspondiam ao mapeamento completo de cada uma das funções do comando, como podemos ver a seguir:



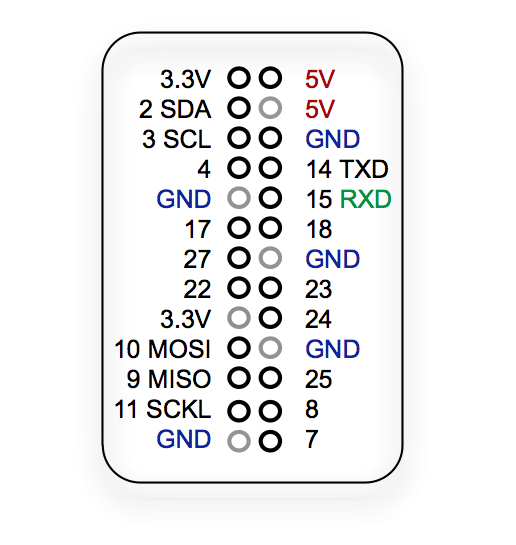
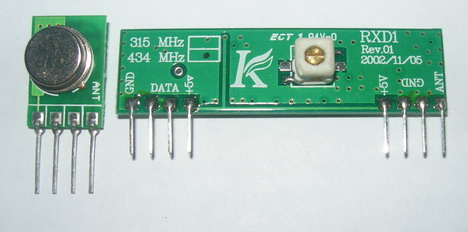
Então cada controlo remoto envia 25 bits, que o recetor vai captar, e consoante o conjunto de zeros e uns que ler vai ativar ou não a sua função.

Em baixo fica uma foto frente e verso do controlo remoto. Reparem bem na minha análise em cima e comparem com o controlo remoto.



# Como Produzir Sinais Wireless

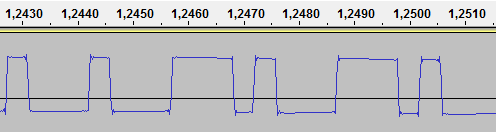
O Comando estava desvendado, percebera como funciona. O que faltava agora? Substituir o comando pelo RaspberryPi!

Aquando da compra do receptor TX433 comprei também um emissor RX433. A ligação do emissor RX433 com o raspberry é quase direta. Este emissor tem uma linha data, muito útil para ligar a um dos pinos GPIO do raspberry. Pode ser alimentado entre 3,3 volts e 12volts, sendo que a voltagem de alimentação vai influenciar diretamente o alcance do emissor.

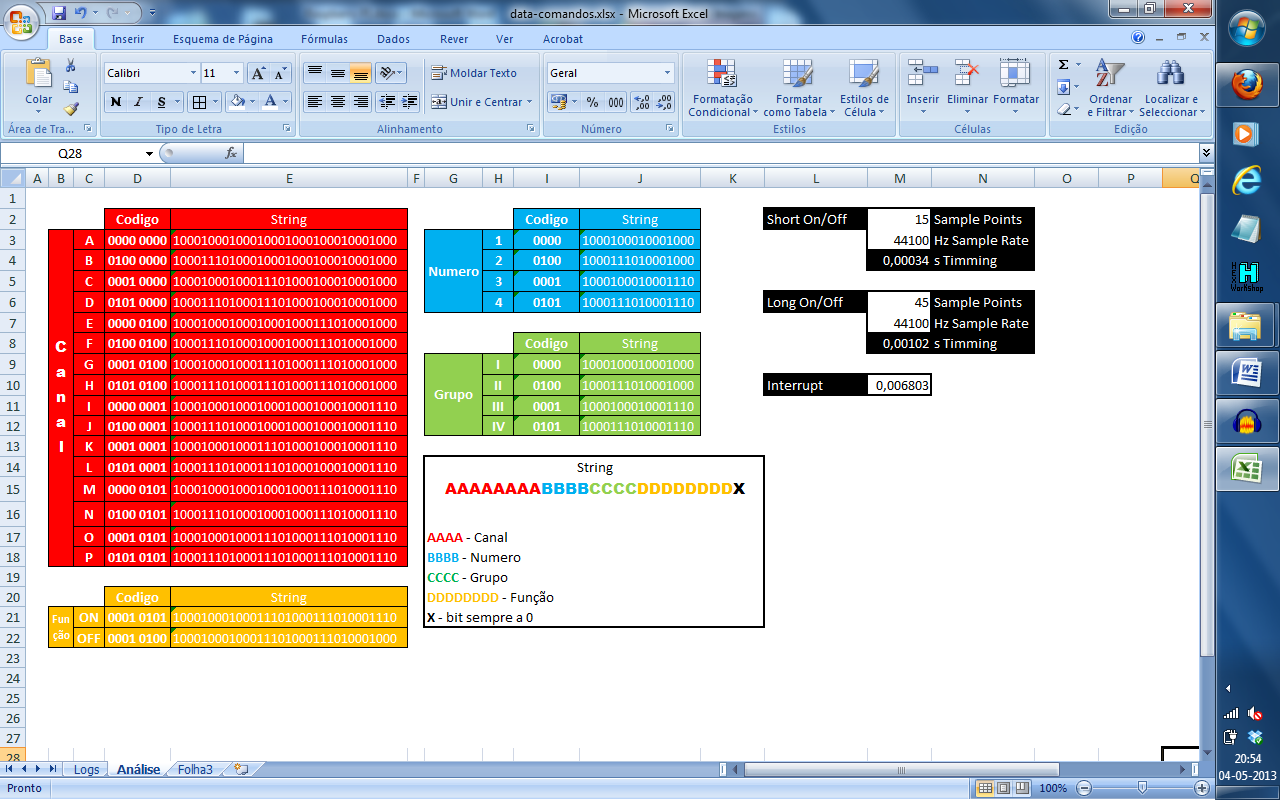


A partir de agora só faltava começar a programar. Encontrei no site da fundação RaspberryPi um exemplo muito simpático de como ligar e desligar um led através do GPIO. Com base nesse pequeno exemplo, construi um programa em C++ que funciona como um comando de sistema, recebendo 4 argumentos e permitindo-me através da consola Linux ligar e desligar os dispositivos controlados pelos comandos.

O Código fala por si, porém repararão com certeza que as strings que se vêm aquando da captura da frequência do comando são diferentes daquelas que se vêm no código. O que acontece é que para reproduzir o sinal foi necessário dizer que cada **0** corresponde á sequencia **1000** e cada **1** corresponde á sequencia **1110**. O mesmo pode ser visto nos quadros a seguir.



De notar também os tempos de duração do sinal, nos quadros a preto. Os tempos de duração do sinal são importantes, na medida que o sinal deve ser o mais parecido com aquele que foi captado, nesse sentido, a duração do sinal deve ser configurada e para isso existem duas variáveis na função SendCode.



# Tornando tudo mais usável

A consola já funcionava, faltava agora ter um ambiente gráfico, mais ou menos funcional, que fosse ao mesmo tempo usável em várias plataformas.

Com esta ideia em mente, e sem grande vontade de desenvolver um interface gráfico para cada plataforma disponível, pensei em simplificar.

Instalei e configurei o Apache2 no RaspberryPi e comecei a programar uma aplicação web, com ajuda de um CGI. A ideia de usar uma aplicação web é simplificar a implementação do sistema em várias plataformas, pois quase todos os dispositivos eletrónicos dispõe de um browser e acesso wireless lan. Desta forma é possível criar uma aplicação que seja compatível em quase todos os dispositivos electronicos de consumo.

Esta aplicação consiste num frontend em html, que permite ao utilizador ter um ambiente amigável de controlo sobre o sistema. Por detrás deste html corre um script CGI que chama a aplicação de sistema que controla a automação.