

Projeto de Sistemas RF

Aula 08 - Transmiss o de dados atrav s do protocolo RS232 – Parte II

Apresentação

Nesta aula, colocaremos juntos os circuitos do transmissor e do receptor projetados na aula passada. Dessa forma, poderemos efetivamente realizar a comunicação entre dois microcontroladores. Além da simulação realizada no software Proteus, iremos programar microcontroladores PIC18F45K20 disponíveis no laboratório. Dividiremos a aula em duas partes (como fizemos na aula anterior) sendo a primeira a parte de software e a segunda de hardware.

Objetivos

Ao final desta aula, você deverá ser capaz de:

- Simular um pequeno sistema de comunicação entre dois microcontroladores.
- Verificar o funcionamento do sistema observando o comportamento do mesmo na simulação.
- Programar dois microcontroladores para realizar a transmissão de dados equivalente ao que foi realizado na simulação.

Simulação da comunicação RS232

Parte I - *Software*

A parte de software consiste exatamente nos mesmos programas definidos na aula passada. Como cada um deles roda separadamente em cada microcontrolador, a ligação externa não faz muita diferença na programação. É importante lembrar apenas que definimos um pequeno protocolo de alto nível para que ambos os micros possam se comunicar.

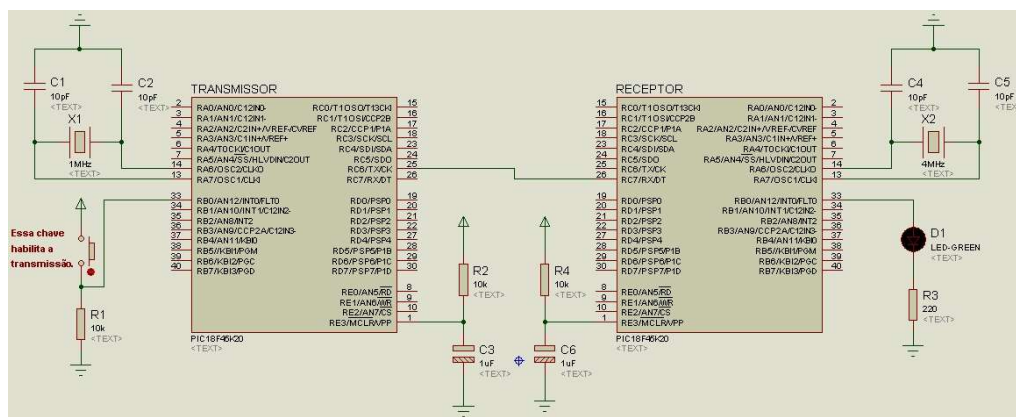
Uma questão interessante a ser feita é: O que acontece se tivermos vários sistemas tentando se comunicar sem fio? Quando um transmissor enviar algo, todos os receptores irão receber... O que fazer nesse caso?

Bem, essa é exatamente a função do protocolo que definimos na aula passada. Basta lembrar que para que um receptor acenda o LED, o mesmo teria que receber o número 85 (bem específico), caso contrário, o LED não acenderia. Se tivermos, por exemplo, 3 sistemas se comunicando ao mesmo tempo, basta que cada um “combine” um número diferente entre transmissor e receptor de maneira que se o transmissor 1 enviar um número (digamos 42), mesmo que todos os receptores recebam esse valor, cada um vai comparar com o seu número “combinado” e só irá responder o receptor que teve combinado o número 42.

Parte II - Hardware

Nessa parte da aula, analisaremos o circuito completo do sistema de comunicação. Os circuitos individuais já foram analisados em separado na aula passada. Agora, o que precisamos fazer é conectar o fio do transmissor (TX, pino 25 do PIC) ao pino RX do receptor (pino 26 do PIC). Esse fio irá simular o meio de transmissão. Dessa maneira, o circuito funcionará da seguinte forma: o usuário pressiona a chave no transmissor e o mesmo envia (via o fio que conecta ambos os circuitos) o valor correspondente ao nosso protocolo (definido na aula passada). O receptor detecta que chegou uma transmissão e testa se o valor recebido corresponde ao protocolo definido. Caso seja, o mesmo acende o LED por alguns segundos e torna a esperar outra transmissão. O circuito completo que implementa esse sistema é mostrado na Figura 1 a seguir.

Figura 01 - Esquemático completo do sistema com o transmissor e o receptor



Atividade 1

- 1. Como atividade para esta aula, faça a seguinte alteração no sistema montado: Mude o programa do transmissor e receptor para que o usuário tenha 3 chaves (no transmissor) e 3 LEDs (no receptor). Cada chave deve acender um LED diferente.**

[Clique aqui](#) para verificar suas respostas.

Respostas

- 1. Não há resposta. Atividade prática para o aluno.**

Programação de microcontroladores para comunicação RS232 - Parte 1

Nesta parte iremos utilizar os códigos já testados na simulação para programar microcontroladores e testar a transmissão dos dados, a partir da montagem do circuito de transmissão testado também na simulação.

Para otimizar o nosso tempo iremos utilizar a placa de demonstração desenvolvida pela Microchip que temos em laboratório, conhecida por "44-Pin Demo Board". Essa placa foi utilizada em disciplinas anteriores do curso, por isso não iremos reservar espaço para analisar a mesma.

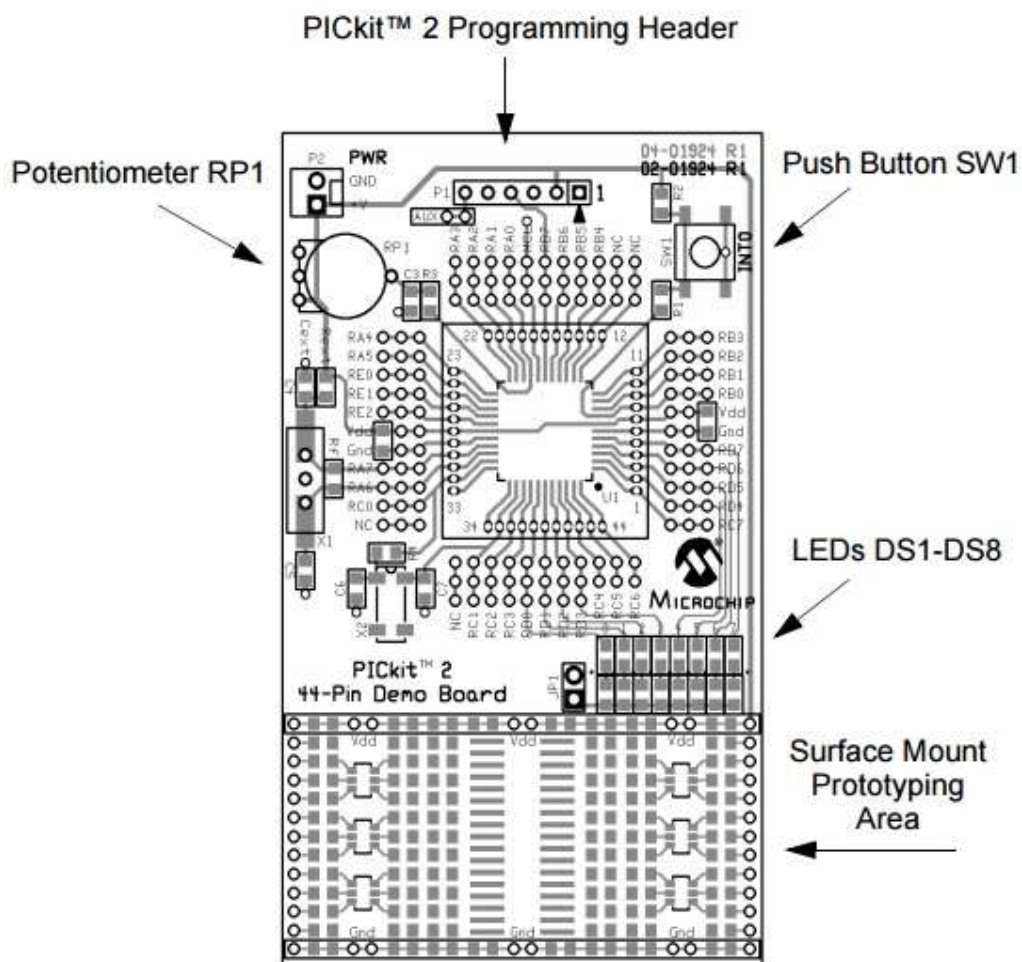
Se Liga!

Caso não lembre do esquemático da placa de demonstração, pesquise na internet o manual da placa "44-Pin Demo Board" desenvolvida pela Microchip para só então darmos continuidade a aula.

Agora vamos analisar inicialmente a parte de hardware do nosso sistema de comunicação.

Nosso objetivo é conseguir montar um circuito equivalente ao da simulação. Para isso devemos lembrar quais são os componentes da placa de demonstração que iremos utilizar, verificando o que devemos alterar no nosso projeto original para funcionar com a placa.

Figura 02 - Layout da placa de demonstração ("44-Pin Demo Board".)



No layout acima observa-se que a placa tem um microcontrolador PIC18F45K20, um botão (push button), alguns LEDs e um potenciômetro. Para nosso projeto iremos precisar de 2 placas, uma placa para transmitir o dado e outra para receber. Conforme nosso projeto original, no circuito de transmissão seria necessário de um botão, que ao ser pressionado habilitaria o envio do dado. Já para o circuito de recepção seria necessário um LED para indicar a chegada do dado. Tanto o botão quanto o LED estão disponíveis na placa de demonstração.

Programação de microcontroladores para comunicação RS232 - Parte 2

Bom, até agora vimos que a placa de demonstração servirá para nosso projeto de comunicação sem a necessidade de qualquer outro componente eletrônico externo. Porém, precisamos agora saber se nosso programa está adequado para a placa. Você deve estar se perguntando agora se o código que fizemos é para um microcontrolador PIC18F45K20, o qual é o mesmo microcontrolador da placa, qual o motivo do programa não ser adequado para nossa aplicação?

O motivo é que não sabemos se o pino do microcontrolador da placa de transmissão que está ligado ao botão é realmente o pino RB0, como definimos no projeto em simulação, e se o pino do microcontrolador da placa de recepção que está ligado ao LED é o pino RB0. Você percebeu alguma contradição? Se vamos utilizar a mesma placa para o circuito de transmissão e de recepção, provavelmente teremos que alterar o pino do botão e do LED.

Vamos agora analisar o esquemático para verificar qual o pino do botão da placa e definir qual dos oito LEDs vamos utilizar, bem como qual o pino do LED escolhido.

Figura 03 - Esquemático da placa de demonstração ("44-Pin Demo Board").

Ao pino RB0.

b. O LED1 (DS1) está ligado a qual pino do microcontrolador?

Ao pino RD0.

c. O potenciômetro (RP1) está ligado a qual pino do microcontrolador?

Ao pino RA0.

Programação de microcontroladores para comunicação RS232 - Parte 3

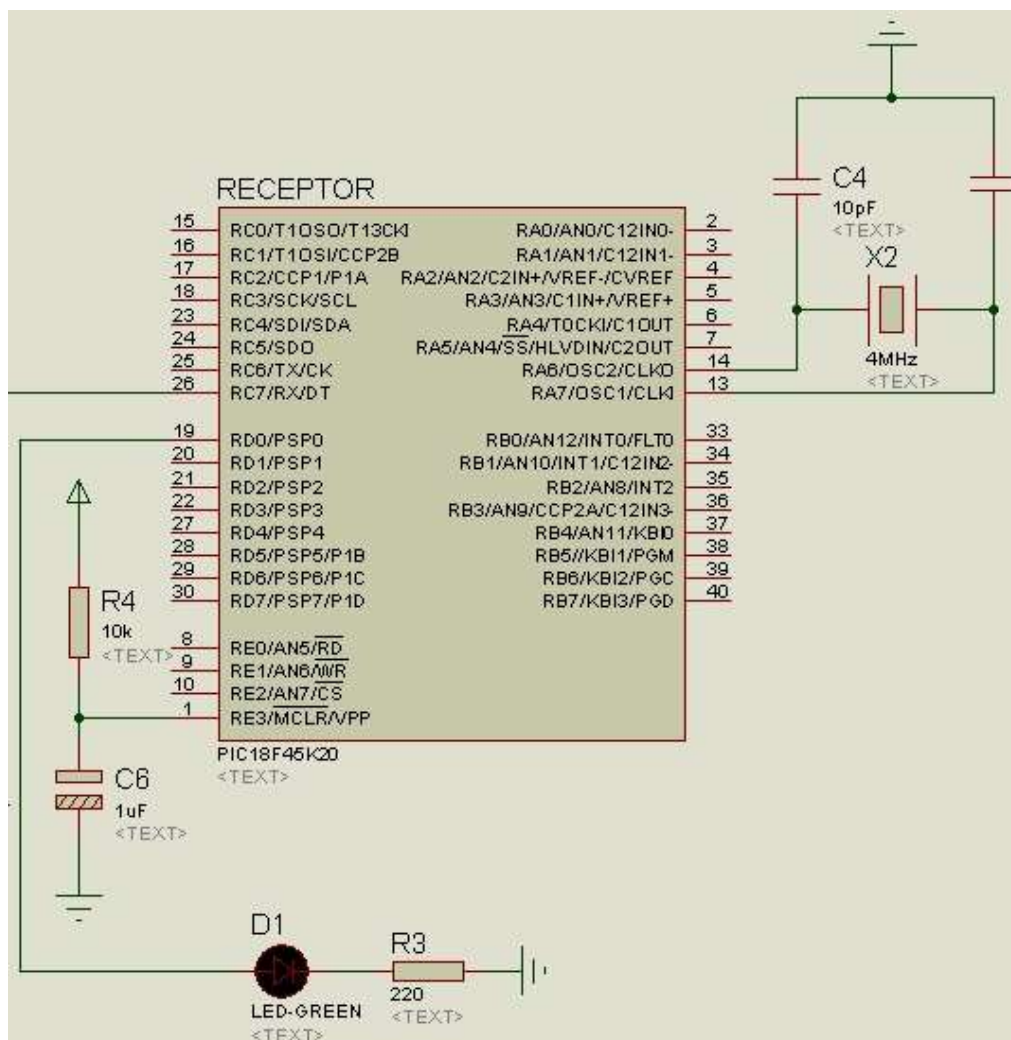
Bem, se você analisou corretamente o esquemático, respondendo as perguntas da atividade anterior, você encontrou que o pino ligado ao botão (SW1) é o RB0 e o pino ligado ao LED1 (DS1) é o RD0. Assim, como nosso circuito de transmissão tem o botão conectado ao pino RB0, não precisamos de alteração no programa de transmissão. Que bom! Mas precisaremos alterar sua lógica. Se você analisou corretamente, verificou que o pino RB0 funciona com lógica invertida, ou seja, sem pressionar o botão, a tensão enviada para o pino RB0 é do VCC e quando se pressiona o botão, a tensão cai a zero. Sendo assim, no código, a verificação do pressionamento do botão necessita ser invertida. Em vez de verificar se o valor > 0, vamos verificar quando ele valor == 0.

Uma outra alteração é que não há cristal nessa placa, logo, não há como fazer funcionar da mesma forma que na simulação (que foi inserido um oscilador externo com cristal). Para que o programa funcione, devemos utilizar o oscilador interno do microcontrolador que é da mesma frequência da que já inserimos no código (ufa!). Para configurar isso devemos adicionar ao topo do programa de ambos, transmissor e receptor, a seguinte linha:

1	// configura o oscilador interno
2	#pragma config FOSC = INTIO67

Já no programa de recepção, o LED não pode está conectado ao RB0. Teremos que alterar o programa para conectar algum LED da placa, nesse caso vamos escolher o LED1 (SW1). Sendo assim, o programa deve definir que o pino RD0 seja acionado quando o dado chegar, como mostrado na figura abaixo.

Figura 04 - Novo esquemático do circuito de recepção.



Atividade 03

1. Altere o programa do receptor para acionar o LED do pino 0 da porta D.
2. Teste o programa resultante da questão anterior no circuito do receptor simulando no Proteus.

[Clique aqui](#) para verificar suas respostas.

Respostas

1. Alterar a porta que seria saída:

// configura a porta D como saída

TRISD = 0;

E alterar quando e quem vai acender

PORTD = 0xff;

e depois quem vai apagar

PORTD=0;

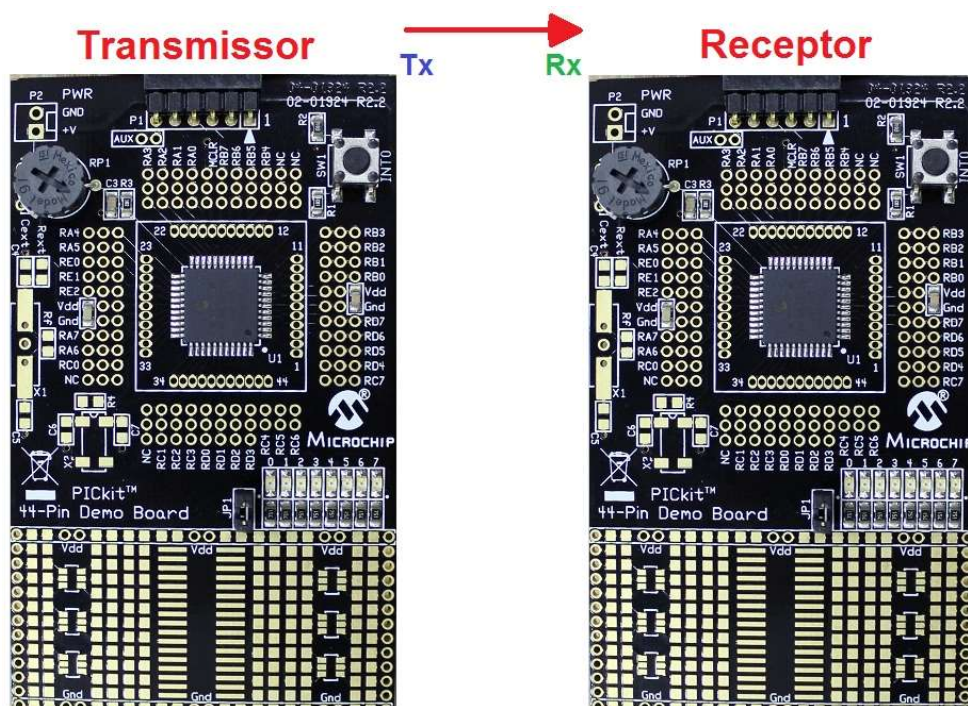
2. Teste o programa resultante da questão anterior no circuito do receptor simulando no Proteus.

Não há resposta. Atividade prática para o aluno.

Programação de microcontroladores para comunicação RS232 - Parte 4

Por fim, vamos montar o circuito de acordo com o modelo mostrado abaixo.

Figura 05 - Esquema de montagem de comunicação RS232.



Atividade 04

1. Programe o microcontrolador da placa de transmissão utilizando o programa original.
2. Programe o microcontrolador da placa de recepção utilizando o programa alterado na atividade anterior.
3. Conecte o pino Tx da placa transmissora com o pino Rx da placa receptora e teste a comunicação RS232

[Clique aqui](#) para verificar suas respostas.

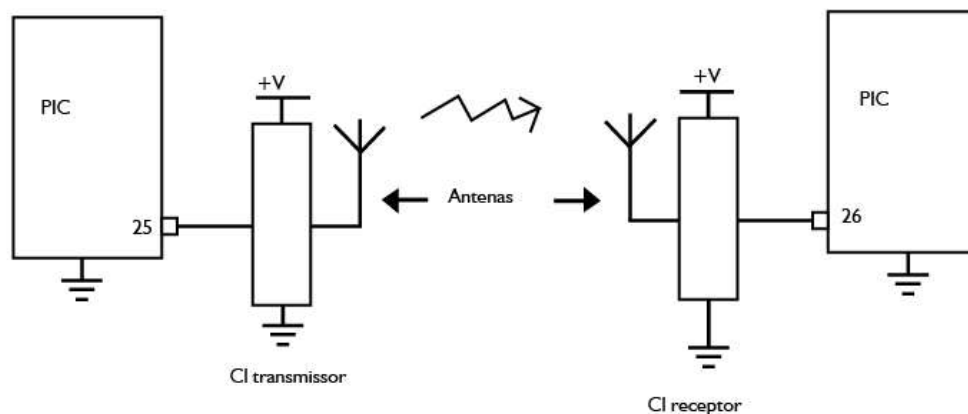
Respostas

1. Não há resposta. Atividade prática para o aluno.

Uso do transmissor sem fio

O próximo passo que deve ser feito em um sistema real é conectar os microcontroladores por transmissores sem fio (como falamos em aulas passadas). A ligação para esse tipo de sistema simples é bem fácil de ser feita. Basicamente, basta conectar (no lugar do fio) a saída do transmissor (pino 25 do PIC) ao componente transmissor de RF (que já vem pronto e montado como se fosse um circuito integrado) e o pino 26 do receptor ao CI receptor de RF. A Figura 2 ilustra como deve ser essa ligação.

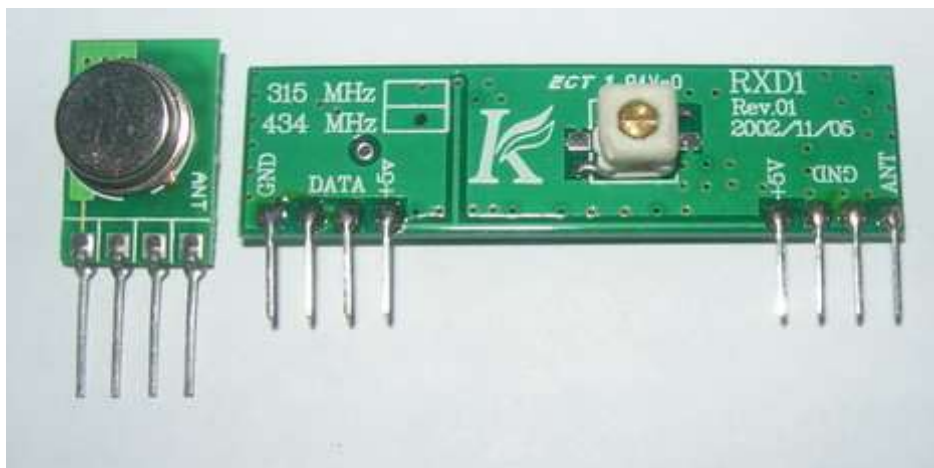
Figura 06 - Esquema da ligação dos CIs transmissor e receptor.



É importante mencionar que, embora na figura 2 existam 4 terras (2 no transmissor e 2 no receptor), esses não estão todos conectados! As conexões elétricas existem apenas entre os terras do transmissor SEPARADOS da conexão existente no receptor. O sinal elétrico será convertido em ondas de rádio e irá chegar ao circuito do receptor via RF.

Um exemplo de módulos RF simples é exibido na Figura 07 abaixo.

Figura 07 - Módulo transmissor (à esquerda) e receptor (à direita) na faixa de 434 MHz com modulação ASK.



Fonte: .

Esse módulo são usados basicamente como mostrado na Figura 06. Basta conectar o fio pino de saída (o pino TX) do PIC à entrada de dados do transmissor, que possui apenas 4 pinos, um para a antena, um pro VCC, um pro terra e um para a entrada de dados.

Já o receptor possui dois pinos de dados, um analógico e outro digital. O analógico tem como tensão média 2,5 V quando não está recebendo nada. Quando recebe um bit 1 vai a 5 V e quando recebe um bit 0 vai a 0 V. Já o pino digital fica em zero quando não está recebendo, vai a 5 V quando recebe um bit 1 e volta a zero se não houver transmissão ou se receber um bit 0.

Lembrando que, como já foi mencionado, a taxa de transmissão utilizada pelos microcontroladores deve ser limitada a taxa aceitável pelo transmissor. No caso desse da foto, ele suporta no máximo 2000 bps. Sendo assim, a taxa utilizada na maioria dos casos é a de 1200 bps se for utilizar esse módulo.

Resumo

Nesta aula, vimos como implementar um sistema bem simples para transmitir um sinal de um microcontrolador para outro. O sinal enviado foi apenas um byte que sinalizaria para o receptor o acendimento de um LED. Vimos também que a transmissão se deu utilizando a porta serial do computador. Nesse primeiro exemplo, a transmissão se deu com fio. O próximo passo é realizar a transmissão sem fios!

Autoavaliação

- 1. É possível utilizar mais de um sistema de transmissão ao mesmo tempo?**
- 2. Para que serve o número que o transmissor usa para transmissão?**
- 3. O que seria o protocolo de alto nível utilizado nesse sistema de transmissão?**

4. Por que os terras da figura 2 são separados?

[Clique aqui](#) para verificar suas respostas.

Respostas

1. É possível utilizar mais de um sistema de transmissão ao mesmo tempo?

Sim, desde que o protocolo utilizado permita diferenciar o destino das informações trafegadas.

2. Para que serve o número que o transmissor usa para transmissão?

Serve justamente como um protocolo simples para diferenciar as transmissões. Assim, muitos transmissores podem transmitir utilizando o meio sem fio e suas mensagens chegaram ao destino adequadamente.

3. O que seria o protocolo de alto nível utilizado nesse sistema de transmissão?

Seria a implementação do envio do número. Ele que diferencia os pacotes, ou seja o conteúdo, a informação. O protocolo de baixo nível é o USART que está apenas preocupado com os níveis de sinal.

4. Por que os terras da figura 2 são separados?

Porque são dois circuitos diferentes que podem estar a grandes distâncias um do outro. O meio de comunicação entre eles é a radio frequência e apenas ela.

Referências

Luz, Carlos Eduardo Sandrini. **Programando Microcontroladores PIC em C**. São Paulo: Editora Ensino Profissional, 2011.

MICROCHIP TECHNOLOGY **44-Pin Demo Board User's Guide**. Disponível em: <<http://ww1.microchip.com/downloads/en/DeviceDoc/41296B.pdf>>[Acessado em 28 março 2015].