

Conversor de protocolo USB 2.0 para UART com GPIO

Características

Barramento serial universal (USB)

- Suporta USB de velocidade total (12 Mb / s)
- Implementa o dispositivo composto do protocolo USB:
 - Classe de dispositivo de comunicação (CDC) para comunicações e configuração
 - Dispositivo de interface humana (HID) para controle de E / S
- Buffer de 128 bytes para lidar com a taxa de transferência de dados em qualquer taxa de transmissão UART:
 - Transmissão de 64 bytes
 - Recebimento de 64 bytes
- Atribuições e descritores VID e PID totalmente configuráveis
- Alimentado por barramento ou alimentado automaticamente
- Compatível com USB 2.0: TID 40001150

Driver USB e suporte de software

- Usa janelas padrão ® Drivers para Porta Virtual Com (VCP): Windows XP (SP2 ou posterior), Windows Vista, Windows 7, Windows 8, Windows 8.1 e Windows 10
- Utilitário de configuração para configuração inicial

Receptor / Transmissor Assíncrono Universal (UART)

- Responde a SET_LINE_CODING Comandos para alterar dinamicamente as taxas de transmissão
- Suporta taxas de transmissão: 300-1000k
- Controle de fluxo de hardware
- Opção de polaridade de sinal UART

Pinos de entrada / saída de uso geral (GPIO)

- Oito pinos de E / S de uso geral

EEPROM

- 256 bytes de EEPROM do usuário

De outros

- Saídas do LED de atividade USB (TxLED e RxLED)
- Pino de saída SSPND
- Pino de saída USBCFG (indica quando a enumeração é concluída)
- Tensão de funcionamento: 3.0V-5.5V
- Entrada do oscilador: 12 MHz
- Proteção contra descarga eletrostática (ESD): > 4 kV Modelo de corpo humano (HBM)
- Industrial (I) Temperatura de operação: -40 ° C a + 85 ° C

Tipos de embalagem

O dispositivo é oferecido nos seguintes pacotes:

- VQFN de 20 derivações (5x5 mm)
- SOIC de 20 derivações
- SSOP de 20 derivações

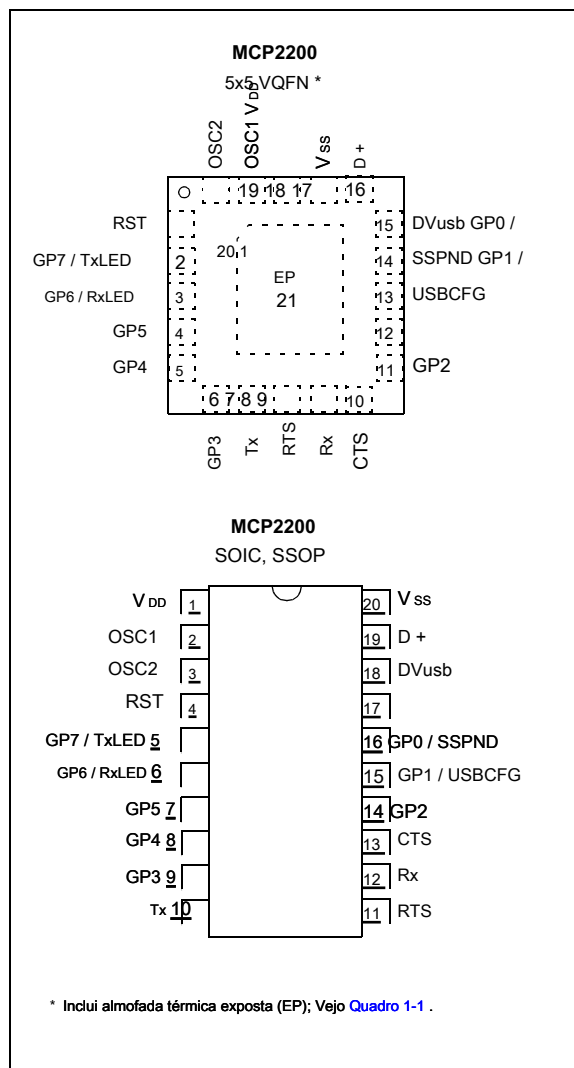
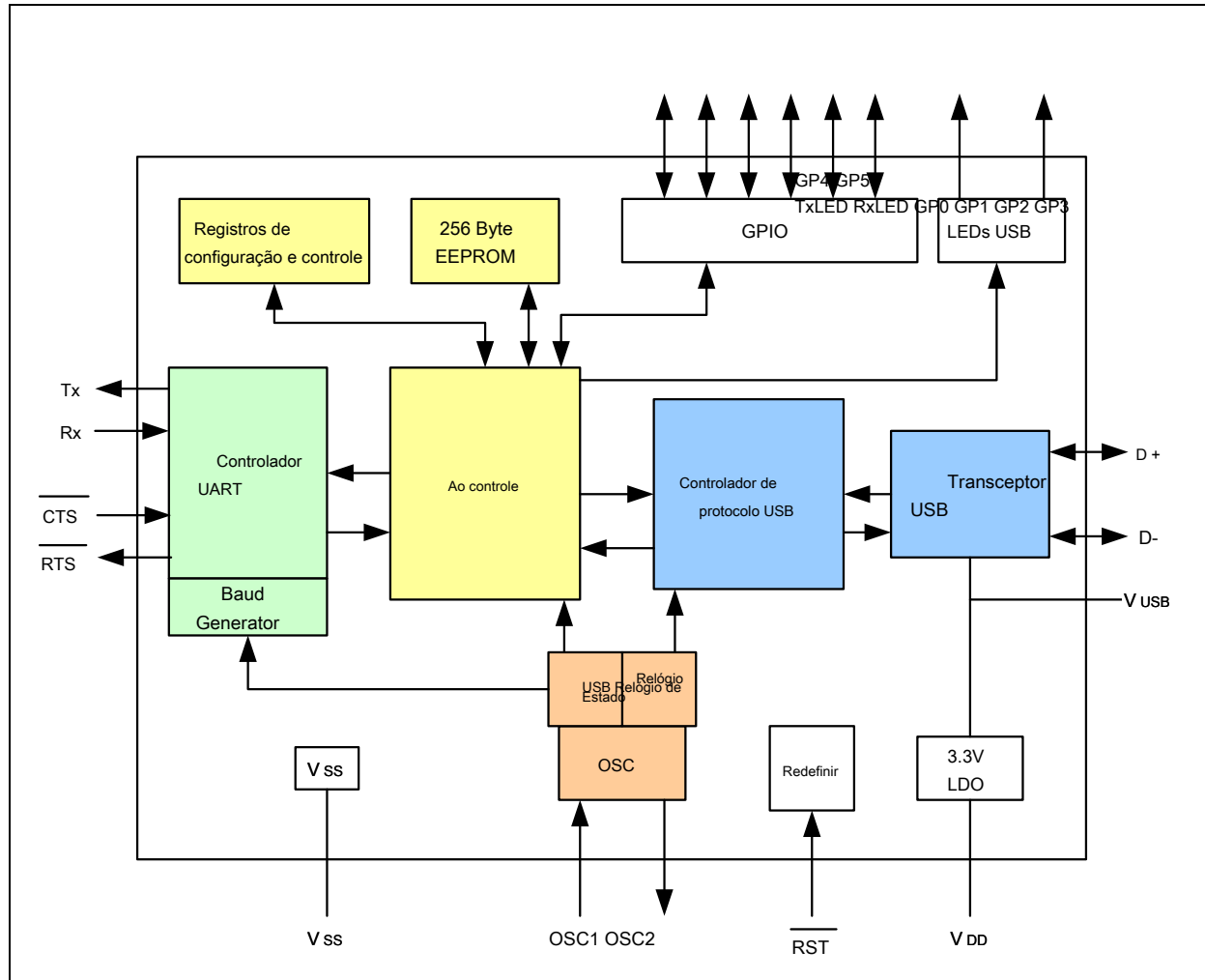


Diagrama de bloco



1.0 DESCRIÇÃO FUNCIONAL

O MCP2200 é um conversor serial USB-UART que permite a conectividade USB em aplicativos que possuem uma interface UART. O dispositivo reduz componentes externos ao integrar os resistores de terminação USB. O MCP2200 também possui 256 bytes de EEPROM de usuário integrado.

O MCP2200 possui oito pinos de entrada / saída de uso geral. Quatro pinos têm funções alternativas para indicar USB e status da comunicação. Veja [Quadro 1-1](#) e [Seção 1.6 "Módulo GPIO"](#) para detalhes sobre as funções dos pinos.

TABELA 1-1: DESCRIÇÃO DO PINOUT

| Nome do PIN | VQFN SSOP | Pin SOIC | Tipo | Função padrão | Função alternativa |
|----------------|-----------|----------|-------------|--|--|
| GP0 / SSPND 13 | | 16 | E / S E / S | E / S de uso geral | Pino de status de suspensão do USB (consulte Seção 1.6.1.1 "Função de pino SSPND") |
| GP1 / USB-CFG | 12 | 15 | E / S E / S | E / S de uso geral | Pino de status da configuração USB (consulte Seção 1.6.1.2 "Função de pino USBCFG") |
| GP2 | 11 | 14 | E / S E / S | E / S de uso geral | |
| GP3 | 6 | 9 | E / S E / S | E / S de uso geral | |
| GP4 | 5 | 8 | E / S E / S | E / S de uso geral | |
| GP5 | 4 | 7 | E / S E / S | E / S de uso geral | |
| GP6 / RxLED | 3 | 6 | E / S E / S | E / S de uso geral | Saída do LED de atividade de recebimento USB (consulte Seção 1.6.1.3 "Função de pinos RxLED (mensagem IN)") |
| GP7 / TxLED | 2 | 5 | E / S E / S | E / S de uso geral | Saída do LED de atividade de transmissão USB (consulte Seção 1.6.1.4 "Função de pino TxLED (mensagem OUT)") |
| CTS | 10 | 13 | Eu | Sinal de entrada "Clear to Send" do controle de fluxo de hardware | |
| RTS | 8 | 11 | O | Controle de fluxo de hardware "Request to Send" sinal de saída | |
| Rx | 9 | 12 | Eu | Entrada USART RX | |
| Tx | 7 | 10 | O | Saída USART TX | |
| RST | 1 | 4 | Eu | A entrada de redefinição deve ser polarizada externamente | |
| VDD | 18 | 1 | P | Potência | |
| VSS | 17 | 20 | P | Terra | |
| OSC1 | 19 | 2 | Eu | Entrada do oscilador | |
| OSC2 | 20 | 3 | O | Saída do oscilador | |
| D+ | 16 | 19 | E / S | USB D+ | |
| D- | 15 | 18 | E / S | USB D- | |
| Vusb | 14 | 17 | P | Pino de alimentação USB (conectado internamente ao 3.3V). Deve ser contornado localmente com um capacitor de cerâmica de alta qualidade. | |
| EP | 21 | - | - | Almofada térmica exposta (EP). Não conecte eletricamente. | |

1.1 Sistemas operacionais suportados

Windows XP (SP2 e posterior), Windows Vista, Windows
Os sistemas operacionais 7, Windows 8, Windows 8.1 e Windows 10 são suportados.

1.1.1 ENUMERAÇÃO

O MCP2200 será enumerado como um dispositivo USB após a reinicialização na inicialização (POR). O dispositivo enumera como um dispositivo de interface humana (HID) para controle de E / S e uma porta de comunicação virtual (VCP).

1.1.1.1 Dispositivo de interface humana (HID)

O MCP2200 enumera como um HID, para que o dispositivo possa ser configurado e a E / S possa ser controlada. Uma DLL que facilita o controle de E / S através de uma interface personalizada é fornecida pela Microchip.

1.1.1.2 Porta Virtual Com (VCP)

A enumeração VCP implementa a conversão de dados de USB para UART.

1.2 Módulo de Controle

O módulo de controle é o coração do MCP2200. Todos os outros módulos são ligados e controlados através do módulo de controle. O módulo de controle gerencia as transferências de dados entre o USB e o UART, bem como as solicitações de comando geradas pelo controlador host USB e os comandos para controlar a função do UART e da E / S.

1.2.1 INTERFACE SERIAL

O módulo de controle faz interface com os módulos UART e USB.

1.2.2 INTERFACE AO DISPOSITIVO O MCP2200 pode ser acessado para leitura e gravação via comandos do host USB. O dispositivo não pode ser acessado e controlado pela interface UART.

1.3 Interface UART

O UCP MCP2200 em terfa c ec consiste nos sinais de dados Tx e Rx e nos pinos de controle de fluxo RTS / CTS. O UART é configurável para várias taxas de transmissão. As taxas de transmissão disponíveis estão listadas em [Quadro 1-3](#).

1.3.1 CONFIGURAÇÃO INICIAL A configuração padrão do UART é 19200, 8, N, 1. A taxa de transmissão padrão de inicialização pode ser alterada usando a ferramenta de PC de configuração fornecida pelo Microchip. Como alternativa, uma ferramenta de configuração personalizada pode ser criada usando a DLL fornecida pelo Microchip para definir a taxa de transmissão e outros parâmetros. Veja [Seção 2.0 "Configuração"](#) para detalhes.

TABELA 1-2: Parâmetro CONFIGURAÇÕES

| UART | Configuração |
|-------------------------------|---------------------------------|
| Taxas de transmissão primária | Vejo Quadro 1-3 |
| Bits de dados | 8 |
| Paridade | N |
| Stop Bits | 1 |

1.3.2 GET / SET CODING DE LINHA GET_LINE_CODING e SET_LINE_CODING

Os comandos são usados para ler e definir os parâmetros UART enquanto em operação. Por exemplo, O HyperTerminal envia o SET_LINE_COMMAND ao conectar à porta. O MCP2200 responde definindo apenas a taxa de transmissão. Os outros parâmetros (bits de dados, paridade, bits de parada) permanecem inalterados.

1.3.2.1 Erros de arredondamento

A configuração da taxa de transmissão primária (com os erros de arredondamento) é mostrada em [Quadro 1-3](#). Se forem usadas taxas de transmissão diferentes das mostradas na tabela, a porcentagem de erro poderá ser calculada usando [Equação 1-1](#) para encontrar a taxa de transmissão real.

TABELA 1-3: TAXAS DEuções primárias da UART

| Taxa desejada | Taxa real | % De erro |
|---------------|-----------|-----------|
| 300 | 300 | 0,00% |
| 1200 | 1200 | 0,00% |
| 2400 | 2400 | 0,00% |
| 4800 | 4800 | 0,00% |
| 9600 | 9600 | 0,00% |
| 19200 | 19200 | 0,00% |
| 38400 | 38339 | 0,16% |
| 57600 | 57692 | 0,16% |
| 115200 | 115385 | 0,16% |
| 230400 | 230769 | 0,16% |
| 460800 | 461538 | 0,16% |
| 921600 | 923077 | 0,16% |

EQUAÇÃO 1-1: RESOLVENDO A TAXA DE baud real

$$\text{ActualRate} = \frac{12 \text{ MHz}}{\text{int } x \cdot}$$

Onde:

$$x = \frac{\text{Baud de 12 MHz}}{\text{Desejado}}$$

1.3.3 TAXAS DE TRANSMISSÃO PERSONALIZADAS As taxas de

transmissão personalizadas são configuradas enviando o

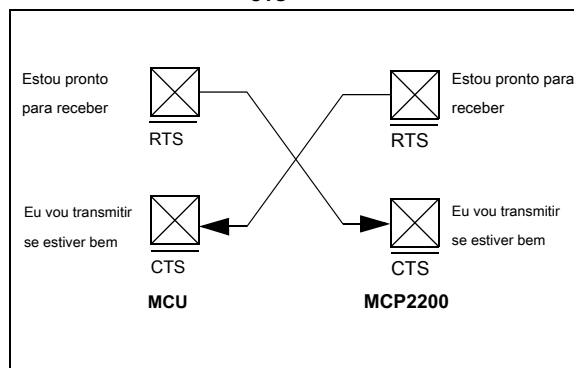
SET_LINE_CODING Comando USB ou usando a DLL. [Vejo Seção 2.0 "Configuração"](#) para mais em formação.

1.3.4 CONTROLE DE FLUXO DE HARDWARE O controle de fluxo de hardware usa o RTS e CTS pinos como um aperto de mão entre dois dispositivos. o RTS o pino de um dispositivo normalmente está conectado ao CTS do outro dispositivo.

RTS é uma saída ativa-baixa que notifica o outro dispositivo quando está pronto para receber dados por driving o pino baixo. O ponto de disparo do MCP2200 para desativar o RTS (alto) é de 63 caracteres. Este é um caractere com falta de "buffer cheio".

O CTS é uma entrada ativa baixa que notifica o MCP2200 when n está pronto para enviar dados. O MCP2200 verifica o CTS antes de carregar e enviar dados UART. Se o pino for afirmado durante uma transferência, a transferência continuará. Referir-se [Figura 1-1](#).

FIGURA 1-1: EXEMPLO DE CONEXÕES RTS / CTS



1.3.4.1 Controle de fluxo desativado

O ponteiro do buffer não aumenta (ou redefine para zero) se o buffer estiver cheio. Portanto, se o controle de fluxo de hardware não estiver ativado e ocorrer um estouro (ou seja, 65 caracteres não processados recebidos), os novos dados substituirão a última posição no buffer.

1.4 Controlador de Protocolo USB

O controlador USB no MCP2200 é USB de velocidade total 2.0 compatível.

- Dispositivo composto (CDC + HID):
 - CDC: comunicações USB-UART
 - HID: controle de E / S, acesso EEPROM e configuração inicial
- Buffer de 128 bytes para lidar com a taxa de transferência de dados em qualquer taxa de transmissão UART:
 - transmissão de 64 bytes
 - recebimento de 64 bytes
- Atribuições e descritores VID e PID totalmente configuráveis (armazenados no chip)
- Alimentação por barramento ou alimentação própria

1.4.1 DESCRITORES Durante a configuração, a interface do PC fornecida armazena os descritores no MCP2200.

1.4.2 SUSPENDER E CONTINUAR Os sinais USB Suspend e Continuar são suportados para gerenciamento de energia do MCP2200. O dispositivo entra no modo de suspensão quando "suspender sinalização" é detectado no barramento.

O MCP2200 sai do modo Suspend quando ocorre um dos seguintes eventos:

- "Reiniciar sinalização" é detectado ou gerado.
- Um sinal USB "Reset" é detectado.
- Uma redefinição do dispositivo ocorre.

1.5 Transceptor USB

O MCP2200 possui um transceptor USB 2.0 de alta velocidade interno conectado internamente ao módulo USB. O transceptor USB obtém **energia do V_{USB} pino, conectado internamente ao regulador de 3,3V. A melhor qualidade de sinal elétrico é obtida quando V_{USB} é contornado localmente com um capacitor de cerâmica de alta qualidade.**

1.5.1 RESISTORES DE PRESSÃO INTERNA Os dispositivos MCP2200 possuem resistores de pull-up projetados para atender aos requisitos de USB de velocidade total.

1.5.2 OPÇÕES DE ENERGIA DO MCP2200 A seguir, estão as principais opções de energia do MCP2200:

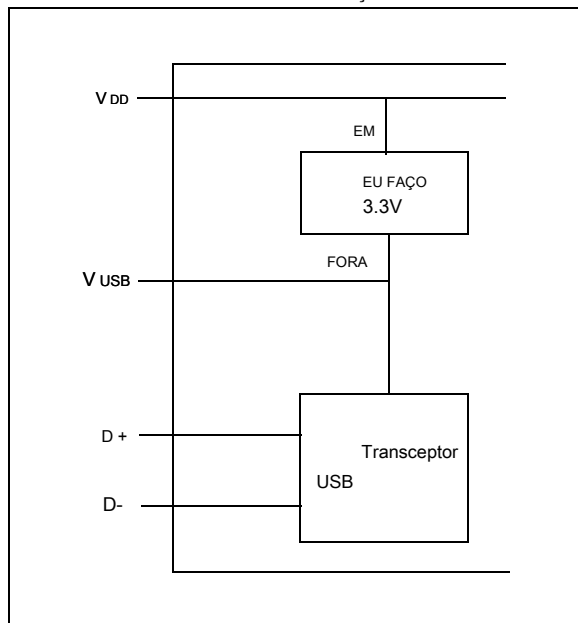
- Alimentado por barramento USB (5V)
- 3.3V Auto-Alimentado

1.5.2.1 Detalhes da fonte de alimentação interna

O MCP2200 oferece várias opções para fonte de alimentação. Para atender aos níveis de sinalização USB necessários, o dispositivo MCP2200 incorpora um LDO interno usado exclusivamente pelo transceptor USB para apresentar os níveis corretos de tensão D+ / D-.

Figura 1-2 mostra as conexões internas do transceptor USB LDO em relação ao V_{DD} trilho de fonte de alimentação. A saída do transceptor USB LDO está ligada ao V_{USB} linha. Um capacitor conectado ao V_{USB} é necessário um pino se o transceptor USB LDO fornecer a alimentação de 3,3V ao transceptor.

FIGURA 1-2: MCP2200 DETALHES INTERNOS DA FONTE DE ALIMENTAÇÃO



O V fornecido DD tensão influencia diretamente a **volt ag el presentes nos pinos GPIO (Rx / Tx e RTS / CTS). Quando V_{DD} é de 5V, todos esses pinos terão uma lógica '1' em torno de 5V com as variações especificadas em**

Seção 3.1 "Características da CC".

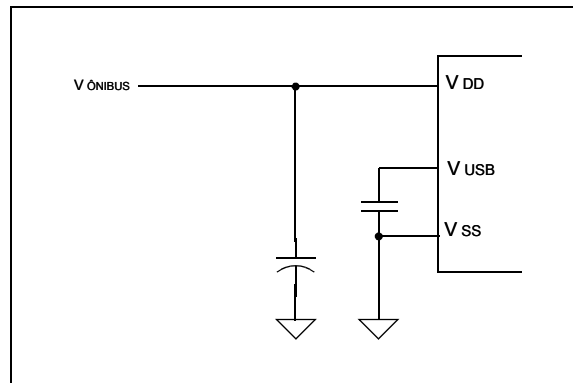
Para aplicativos que exigem uma lógica lógica de 3,3V '1' nível, V_{DD} deve estar conectado a uma fonte de alimentação

Tensão de 3.3V. Nesse caso, o transceptor USB interno LDO não pode fornecer os 3.3V necessários de energia. É necessário conectar também o V_{USB} pino no trilho da fonte de alimentação de 3,3V. Dessa forma, o transceptor USB é ligado diretamente a partir da fonte de alimentação de 3.3V.

1.5.2.2 Alimentado por barramento USB (5V)

No modo Bus Power Only, toda a energia do aplicativo é extraída do **USB (Figura 1-3)** Este é efetivamente o método de energia mais simples para o dispositivo.

FIGURA 1-3: SOMENTE POTÊNCIA



Para atender aos requisitos atuais de irrupção das especificações USB 2.0, a capacitância total efetiva exibida em V_{0NBUS} e o solo não deve ter mais que 10 µF. Se não for superior a 10 µF, é necessário algum tipo de limitação de corrente de irrupção. Para obter mais detalhes sobre a limitação de corrente de irrupção, considere a versão mais recente do

"Especificação de barramento serial universal".

De acordo com a especificação USB 2.0, todos os dispositivos USB também devem suportar o modo de suspensão de baixa energia. No modo de suspensão USB, os dispositivos não devem consumir mais que 500 µA (ou 2,5 mA para dispositivos de alta potência com capacidade de ativação remota) a partir de 5V V_{0NBUS} linha do cabo USB.

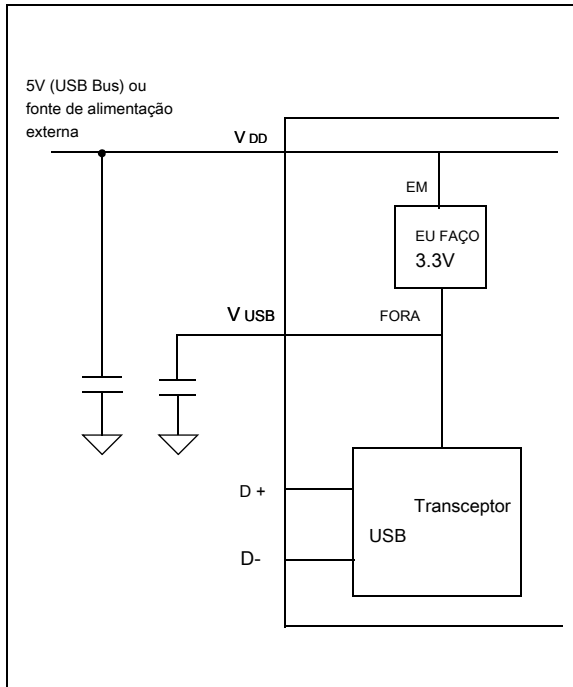
O host sinaliza o dispositivo USB para entrar no modo Suspend, interrompendo todo o tráfego USB nesse dispositivo por mais de 3 ms.

O barramento USB fornece uma tensão de 5V. No entanto, o transceptor USB requer 3,3V para a sinalização (nas linhas D+ e D-).

Durante o modo USB Suspend, o resistor pull-up D+ ou D- deve permanecer ativo, o que consumirá parte do orçamento atual suspenso permitido (500 µA / 2,5 mA). O V_{USB} é necessário um pino para ter um capacitor de derivação externo. Recomenda-se que o capacitor seja um capacitor de cerâmica entre 0,22 µF. e 0,47 µF.

Figura 1-4 mostra um circuito em que o LDO interno do MCP2200 é usado para fornecer 3,3V ao transceptor USB. A voltagem no V_{DD} **aff ects t ele v** níveis de olagem nos pinos GPIO (Rx / Tx e RTS / CTS). Com V_{DD} a 5V, esses pinos terão uma lógica '1' de 5V com as variações especificadas em **Seção 3.1 "Características da CC"**.

FIGURA 1-4: OPÇÃO DE ALIMENTAÇÃO TÍPICA USANDO OS 5V FORNECIDOS PELO USB



1.5.2.3 3.3V Auto-Alimentado

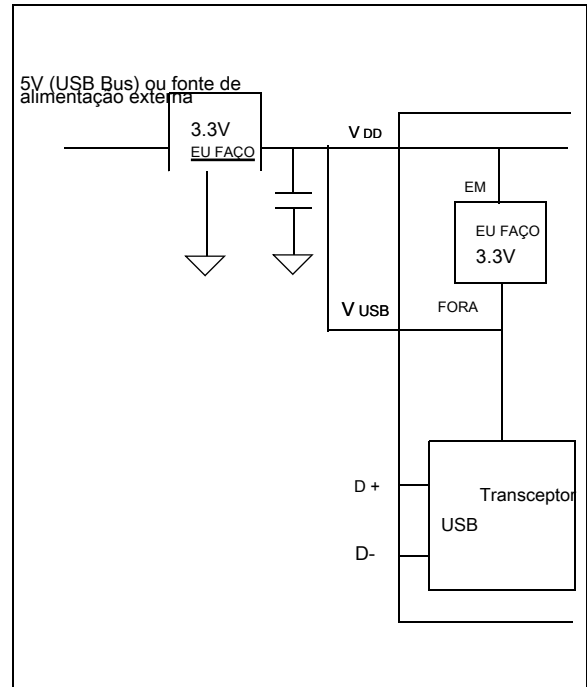
Normalmente, muitos aplicativos incorporados estão usando fontes de alimentação de 3.3V. Quando essa opção está disponível no sistema de destino, o MCP2200 pode ser ligado a partir do trilho de fonte de alimentação existente de 3.3V. As conexões típicas do MCP2200 são mostradas em [Figura 1-5](#). Neste exemplo, o MCP2200 possui ambos VDD e VUSB

linhas amarradas ao trilho de 3.3V. Essas conexões vinculadas desativam o transceptor USB interno LDO do MCP2200 para regular a fonte de alimentação em VUSB PIN. Outra consequência é que o nível lógico dos pinos GPIO estará no nível de 3.3V, de acordo com as variações especificadas

em [Seção 3.1](#) "DC

Características".

FIGURA 1-5: USANDO UMA FONTE DE ALIMENTAÇÃO DE 3.3V FORNECIDA EXTERNAMENTE



1.6 Módulo GPIO

O módulo GPIO é uma porta de E / S padrão de 8 bits.

1.6.1 FUNÇÕES DE PIN CONFIGURÁVEIS Os pinos podem ser configurados como:

- GPIO - entrada ou saída de uso geral configurável individualmente
- SSPND - estado de suspensão USB
- USBCFG - indica o status da configuração USB
- RxLED - indica tráfego de recepção USB
- TxLED - indica tráfego de transmissão USB

1.6.1.1 Função de pino SSPND

O pino SSPND (se ativado) reflete o estado USB (Suspend / Continuar). O pino está ativo-baixo quando o estado de suspensão foi emitido pelo host USB. Da mesma forma, o pino é acionado 'alto' após o estado Continuar.

Este pino permite que o aplicativo entre no modo de baixa energia quando a comunicação USB é suspensa e muda para um estado ativo completo quando a atividade USB é retomada.

1.6.1.2 Função Pin USBCFG

O pino USBCFG (se ativado) inicia 'baixo' durante a inicialização ou após a redefinição e fica 'alto' depois que o dispositivo é configurado com êxito no USB. O pino ficará 'baixo' quando estiver no modo de suspensão e 'alto' quando o USB for reiniciado.

1.6.1.3 Função de pinos RxLED (mensagem IN)

O 'Rx' no nome do pino refere-se ao host USB. O pino RxLED é um indicador para mensagens USB 'IN'. Esse pino irá pulsar baixo por um período de tempo (configurável para ~ 100 ms ou ~ 200 ms) ou alternar para o estado oposto

para toda mensagem recebida (Mensagem IN) pelo host USB. Isso permite que o aplicativo conte mensagens ou forneça uma indicação visual do tráfego USB.

1.6.1.4 Função de pinos TxLED (mensagem OUT)

O 'Tx' no nome do pino refere-se ao host USB. O pino TxLED é um indicador de mensagens USB 'OUT'. Esse pino pulsará baixo por um período de tempo (configurável para ~ 100 ms ou ~ 200 ms) ou alternará para o estado oposto para cada mensagem transmitida (mensagem OUT) pelo host USB. Isso permite que o aplicativo conte mensagens ou forneça uma indicação visual do tráfego USB.

1.7 Módulo EEPROM

O módulo EEPROM é uma matriz de 256 bytes de memória não volátil. Os locais de memória são acessados para operações de leitura / gravação via comandos do host USB. Referir-se Seção 2.0 "Configuração" para obter detalhes sobre como acessar a EEPROM. As células de memória para EEPROM de dados são classificadas para suportar milhares de ciclos de apagamento / gravação, até 100K para EEPROM. Retenção de dados sem

atualização é conservadora estimado em mais de 40 anos.

O host deve aguardar a conclusão do ciclo de gravação e, em seguida, verifique a gravação lendo o (s) byte (s).

1.8 RESET / POR

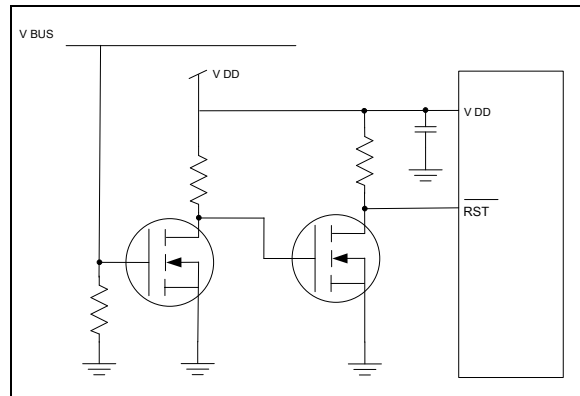
1.8.1 REINICIAR PIN

O pino RST fornece um método para acionar uma redefinição externa do dispositivo. Uma redefinição é gerada mantendo o pino baixo. Esses dispositivos possuem um filtro de ruído no caminho Reset, que detecta e ignora pequenos pulsos.

1.8.2 RESET POWER-ON (POR) Um pulso POR é gerado no chip sempre que V_{DD} sobe acima de um certo limiar. Isso permite que o dispositivo inicie no estado inicializado quando V_{DD} é adequado para operação.

Para tirar proveito do circuito POR, amarre o pino RST através de um resistor (1 kΩ a 10 kΩ) a V_{DD}. Isso eliminará os componentes externos de RC geralmente necessários para criar um atraso no POR. No sduende- configuração de alimentação, recomenda-se amarrar o pino RST ao V_{DD} linha do conector USB, como em Figura 1-6 .

FIGURA 1-6: CONECTANDO O PIN RST EM UMA CONFIGURAÇÃO AUTOMÁTICA (RECOMENDADO)



Quando o dispositivo inicia a operação normal (ou seja, sai da redefinição condição), os parâmetros operacionais do dispositivo (tensão, frequência, temperatura etc.) devem ser atendidos para garantir a operação. Se essas condições não forem alcançadas, o dispositivo deverá ser mantido em Reset até que as condições operacionais sejam atendidas.

1.9 Oscilador

O relógio de entrada deve ser de 12 MHz para fornecer a frequência adequada para o módulo USB.

A velocidade máxima do USB é definida como 12 Mb / s. A precisão da entrada do relógio é de $\pm 0,25\%$ (máximo de 2.500 ppm).

FIGURA 1-7: OPERAÇÃO DE CRISTAL DE QUARTZO

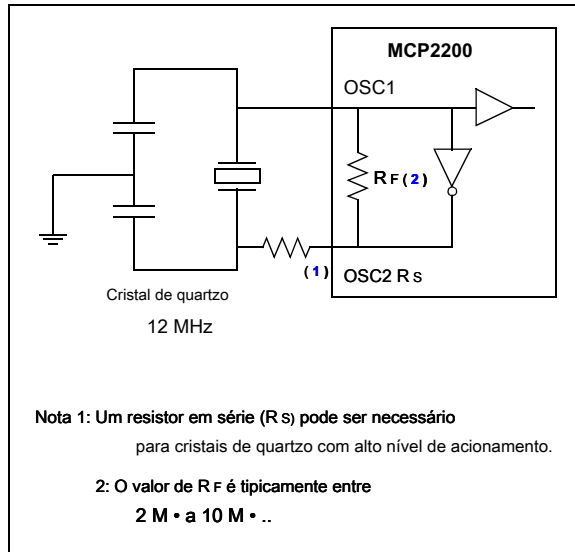
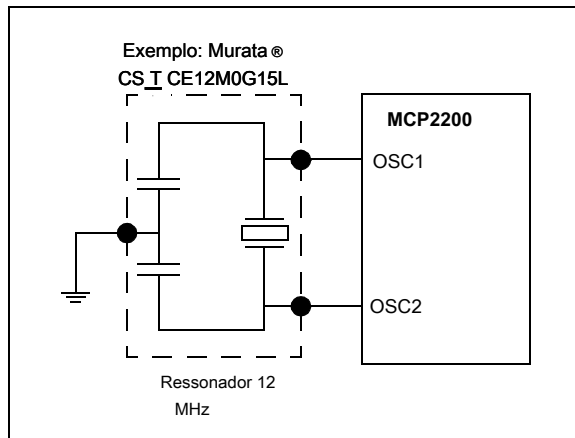


FIGURA 1-8: OPERAÇÃO DE RESSONADOR CERÂMICO



2.0 CONFIGURAÇÃO

O MCP2200 é configurado escrevendo comandos especiais usando a interface HID. A configuração pode ser alcançada usando o utilitário de configuração fornecido pelo Microchip. Como alternativa, um utilitário personalizado pode ser desenvolvido usando a DLL disponível na página do produto MCP2200.

2.1 Utilitário de configuração

O utilitário de configuração fornecido pelo Microchip permite ao usuário configurar o MCP2200 para padrões personalizados. O utilitário de configuração (mostrado em [Figura 2-1](#)) se conecta à interface HID do dispositivo, onde todos os recursos configuráveis podem ser definidos.

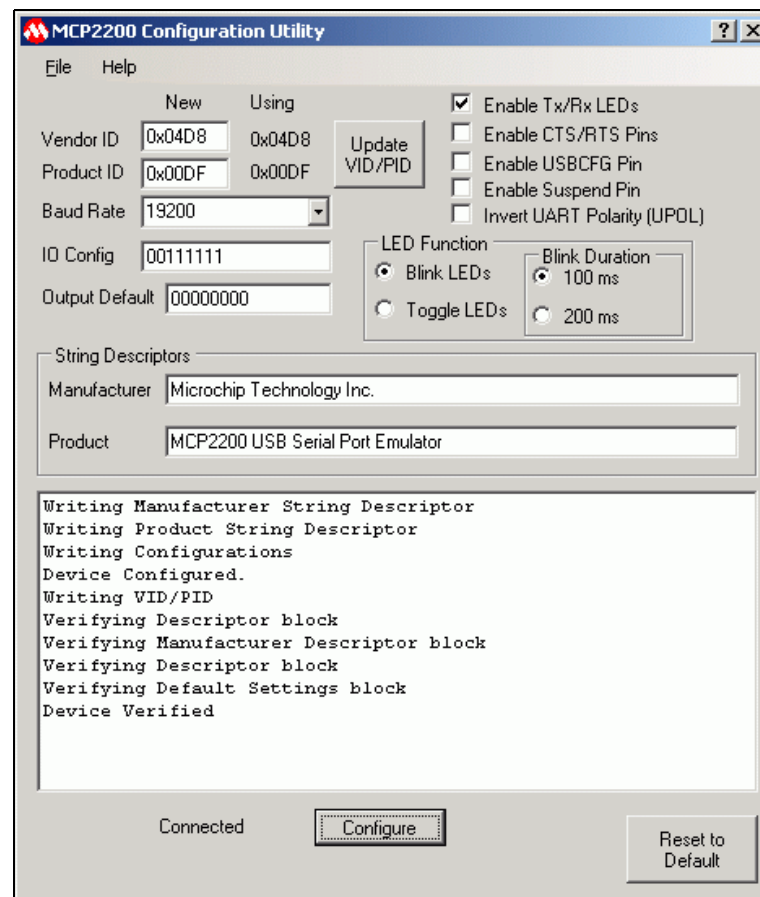
2.2 Sequência serial

O MCP2200 é fornecido de fábrica com uma sequência serial USB serializada.

TABELA 2-1: DESCRIÇÕES DA CONFIGURAÇÃO Nome da

| configuração | Descrição |
|-------------------------------|--|
| ID do fornecedor (0x04D8) | A identificação do fornecedor de USB atribuída ao Microchip pelo consórcio USB. |
| ID do produto (0x00DF) | ID do dispositivo atribuído pelo Microchip. O dispositivo pode ser usado como está ou o Microchip pode atribuir um PID personalizado mediante solicitação. |
| Taxa de transmissão | Define a taxa de transmissão UART usando uma lista de taxas de transmissão primárias. Consulte a seção UART para obter detalhes sobre a configuração de taxas de transmissão não primárias. |
| Configuração de E / S | Configura individualmente a E / S para entradas ou saídas. |
| Padrão de E / S | Configura individualmente o estado padrão de saída para os pinos configurados como saídas. |
| LEDs Tx / Rx | Ativa / desativa os pinos GP6 e GP7 para funcionarem como indicadores de tráfego USB. Os pinos estão ativos-baixos <u>W</u> galinha co <u>nfig</u> como indicadores de tráfego. |
| Controle de fluxo de hardware | Ativa / desativa o controle de fluxo CTS e RTS. |
| Pin USBCFG | Ativa / desativa o pino GP1 como um indicador de status de configuração USB. |
| Suspend Pin | Ativa / desativa o pino GP0 como um pino de status de suspensão USB. |
| Inverter Sentido | Ativa / desativa o UART lin es st uma tes: <ul style="list-style-type: none">- Normal - Tx / Rx inativo; <u>CTS</u> / <u>RTS</u> ativo-baixo- Invertido - Tx / Rx ocioso-baixo; CTS / RTS ativo-alto |
| Cadeia de fabricação | String do fabricante do USB. |
| Cadeia de Produto | Sequência de produtos USB. |

FIGURA 2-1: UTILIDADE DE CONFIGURAÇÃO



2.3 Configuração simples e DLL de E / S

Para ajudar o usuário a desenvolver um configurador personalizado, o Microchip fornece uma DLL que usa o Microsoft®. NET Framework 3.5. Há documentação sobre drivers e utilitários na página do produto MCP2200 em www.microchip.com (na seção Software) com informações sobre como associar a DLL a um projeto Visual C ++.

2.3.1 CHAMADAS SIMPLES DE DLL DE E / S

Quadro 2-2 lista as funções fornecidas pela DLL para permitir a configuração do dispositivo e o controle da E / S.

TABELA 2-2: FUNÇÕES DE CONFIGURAÇÃO

| | |
|---|-------------------------------|
| Inicialização de nome de categoria e | |
| função (Nota 1) | |
| void InitMCP2200 (VID, PID) | |
| Configuração (Nota 2) | |
| bool ConfigureIO (máscara) | |
| bool ConfigureDefaultOutput (máscara, defaultGpioOutputValue) bool fnRxLED (DESLIGADO / ALTERNADO / | |
| BLINKSLOW / BLINKFAST) bool fnTxLED (DESLIGADO / TOGGLE / BLINKSLOW / BLINKFAST) bool | |
| fnHardwareFlowControl (ON / OFF) bool fnUL (ON / OFF) | |
| | |
| | |
| | |
| bool ConfigureMCP2200 (máscara, taxa de transmissão, RxLedMode, TxLedMode, flowCtrl, ULoad, suspender) bool ConfigureIO (máscara) | |
| | |
| Diversos | |
| String ^ GetDeviceInfo (deviceIndex) não assinado int | |
| GetNoOfDevices () int GetSelectedDevice () String ^ | |
| GetSelectedDeviceInfo () bool IsConnected () | |
| | |
| | |
| | |
| int SelectDevice (uiDeviceNo) int ReadEEPROM (uiEEPAddress) int | |
| WriteEEPROM (uiEEPAddress, ucValue) | |
| | |
| Controle de E / S | |
| bool ClearPin (pinnumber) bool SetPin | |
| (pinnumber) | |
| bool ReadPin (pinnumber, * pinvalue) int ReadPinValue | |
| (pinnumber) bool ReadPort (* portValue) int ReadPortValue () | |
| bool WritePort (portValue) | |
| | |
| | |
| Sumário | |
| bool SimpleIOClass :: ClearPin (pino int não assinado) | Seção 2.3.1.1 |
| bool SimpleIOClass :: ConfigureIO (IOMap de caracteres não assinado) | Seção 2.3.1.2 |
| bool SimpleIOClass :: ConfigureDefaultOutput (char não assinado ucIoMap, char não assinado ucDefValue) | Seção 2.3.1.3 |
| bool SimpleIOClass :: ConfigureMCP2200 (IOMap de caractere não assinado, BaudRateParam longo não assinado, int RxLEDMode não assinado, int TxLEDMode não assinado, int TxLEDMode não assinado, FLOW booleano, ULOAD booleano, SSPND booleano) | Seção 2.3.1.4 |
| bool SimpleIOClass :: fnHardwareFlowControl (unsigned int onOff) | Seção 2.3.1.5 |

Nota 1: Antes de qualquer uso da API DLL, uma chamada para o InitMCP2200 () função é necessária. Esta função é a única função de inicialização na DLL apresentada.

2: A configuração precisa ser definida apenas uma vez - é armazenada no NVM.

TABELA 2-2: FUNÇÕES DE CONFIGURAÇÃO (CONTINUAÇÃO)

| Nome da Categoria e Função | |
|---|--------------------------------|
| (Continuação) | |
| bool SimpleIOClass :: fnRxLED (modo int não assinado) | Seção 2.3.1.6 |
| bool SimpleIOClass :: fnSetBaudRate (BaudRateParam longo não assinado) | Seção 2.3.1.7 |
| bool SimpleIOClass :: fnSuspend (unsigned int onOff) | Seção 2.3.1.8 |
| bool SimpleIOClass :: fnTxLED (modo int não assinado) | Seção 2.3.1.9 |
| bool SimpleIOClass :: fnULoad (unsigned int onOff) | Seção 2.3.1.10 |
| String ^ SimpleIOClass :: GetDeviceInfo (não assinado int uiDeviceNo) | Seção 2.3.1.11 |
| unsigned int SimpleIOClass :: GetNoOfDevices (void) | Seção 2.3.1.12 |
| int SimpleIOClass :: GetSelectedDevice (void) | Seção 2.3.1.13 |
| String ^ SimpleIOClass :: GetSelectedDeviceInfo (void) | Seção 2.3.1.14 |
| void SimpleIOClass :: InitMCP2200 (unsigned int VendorID, unsigned int ProductID) | Seção 2.3.1.15 |
| bool SimpleIOClass :: IsConnected () | Seção 2.3.1.16 |
| int SimpleIOClass :: ReadEEPROM (não assinado int uiEEPAddress) | Seção 2.3.1.17 |
| bool SimpleIOClass :: ReadPin (pino int não assinado, int não assinado * valor de retorno) | Seção 2.3.1.18 |
| int SimpleIOClass :: ReadPinValue (pino int não assinado) | Seção 2.3.1.19 |
| bool SimpleIOClass :: ReadPort (unsigned int * returnvalue) | Seção 2.3.1.20 |
| int SimpleIOClass :: ReadPortValue () | Seção 2.3.1.21 |
| int SimpleIOClass :: SelectDevice (não assinado int uiDeviceNo) | Seção 2.3.1.22 |
| bool SimpleIOClass :: SetPin (pino int não assinado) | Seção 2.3.1.23 |
| int SimpleIOClass :: WriteEEPROM (não assinado int uiEEPAddress, char não assinado ucValue) | Seção 2.3.1.24 |
| bool SimpleIOClass :: WritePort (não assinado int portValue) | Seção 2.3.1.25 |
| Constantes | |
| const não assinado int OFF = 0; const sem sinal int ON = 1; | |
| const não assinado int TOGGLE = 3; const não assinado int | |
| BLINKSLOW = 4; const não assinado int BLINKFAST = 5; | |
| | |
| | |

Nota 1: Antes de qualquer uso da API DLL, uma chamada para o `InitMCP2200 ()` função é necessária. Esta função é a única função de inicialização na DLL apresentada.

2: A configuração precisa ser definida apenas uma vez - é armazenada no NVM.

2.3.1.1 ClearPin

Função:

bool SimpleIOClass :: ClearPin (pino int não assinado)

Resumo: Limpa o pino especificado.

Descrição: Limpa o pino especificado para a lógica '0'.

Pré-condição: Este pino deve ser configurado anteriormente como uma saída via `ConfigureIO` ou `ConfigureDefaultOutput` ligar.

Parâmetros: PIN - O número do pino a ser definido (0-7).

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Nenhum

EXEMPLO 2-1: Resumo

```
if (SimpleIOClass :: ClearPin (2))
{
    lblStatusBar-> Text = "Sucesso";

}
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: GetLastError;
```

2.3.1.2 ConfigureIO

Função:

bool SimpleIOClass :: ConfigureIO (IOMap de caracteres não assinado)

Resumo: Configura os pinos GPIO para entrada digital ou saída digital.

Descrição: Os pinos GPIO podem ser configurados como entrada digital ou saída digital.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros:

IOMap - um byte que representa um bitmap da configuração do GPIO:

- um pouco definido como '1' será uma entrada digital
- um pouco definido como '0' será uma saída digital
- MSB - - - - - LSB

GP7 GP6 GP5 GP4 GP3 GP2 GP1 GP0 Retorna:

Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-2:

```
if (SimpleIOClass :: ConfigureIO (0xA5) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.3 ConfigureIODefaultOutput

Função:

bool SimpleIOClass :: ConfigureIODefaultOutput (char não assinado ucIoMap, char não assinado ucDefValue)

Resumo: Configura os pinos de entrada / saída para entrada digital, saída digital e também o valor padrão da trava de saída.

Descrição: Os pinos IO podem ser configurados como entrada digital ou saída digital. O valor padrão da trava de saída é recebido como um parâmetro.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros:

1 ucIoMap - um byte que representa um bitmap usado para definir os GPIOs como entrada ou saída.

- '1' configura o GPIO como entrada
- '0' configura o GPIO como saída
- MSB - - - - - LSB

GP7 GP6 GP5 GP4 GP3 GP2 GP1 GP0

2) ucDefValue - o valor padrão que será carregado na trava de saída (afeta apenas os pinos configurados como saídas).

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-3:

```
if (SimpleIOClass :: ConfigureIODefaultOutput (IoMap, DefValue) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.4 ConfigureMCP2200

Função:

bool SimpleIOClass :: ConfigureDefaultOutput (BaudRateParam longo não assinado, não assinado int RxLEDMODE, não assinado int TxLEDMODE, bool FLOW, bool ULOAD, bool SSPND)

Resumo: Configura o dispositivo.

Descrição: Define a designação padrão do GPIO, taxa de transmissão, modos de LED TX / RX, controle de fluxo.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros:

1 iOMap - Um byte que representa o estado de entrada / saída dos pinos (cada bit pode ser um '1' para entrada ou '0' para resultado).

2) BaudRateParam - a taxa de transmissão de comunicação padrão.

3) RxLEDMODE - pode assumir um dos valores constantes (OFF, ON, TOGGLE, BLINKSLOW, BLINKFAST) para definir o comportamento do LED Rx.

• OFF = 0 0

• ON = 1

• TOGGLE = 3

• BLINKSLOW = 4

• BLINKFAST = 5

4) TxLEDMODE - pode assumir um dos valores definidos (OFF, ON, TOGGLE, BLINKSLOW, BLINKFAST) para definir o comportamento do LED Tx.

5) FLOW - este parâmetro estabelece o método de controle de fluxo padrão (False - no HW flow control, True - RTS / CTS controle de fluxo).

6 ULOAD - este parâmetro estabelece quando o USB carregou a configuração.

7) SSPND - este parâmetro estabelece quando o USB envia o sinal do modo de suspensão. Devoluções:

Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Nenhum.

EXEMPLO 2-4:

```
if (SimpleIOClass :: ConfigureMCP2200 (0x43, 9600, BLINKSLOW, BLINKFAST, falso, falso, falso) == SUCESSO)

    lblStatusBar-> Text = "Sucesso"; else lblStatusBar-> Text = "Comando

    inválido"
```

2.3.1.5 fnHardwareFlowControl

Função:

bool SimpleIOClass :: fnHardwareFlowControl (unsigned int onOff)

Resumo: Configura o controle de fluxo do MCP2200. A configuração de controle baixo será armazenada na NVRAM.

Descrição: Define o controle de fluxo como controle de fluxo HW (RTS / CTS) ou sem controle de fluxo.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Ligado desligado:

• '1' se for necessário controle de fluxo HW

• '0' se nenhum controle de fluxo for necessário

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-5:

```
if (SimpleIOClass :: fnHardwareFlowControl (1) == SUCESSO)

    lblStatusBar-> Text = "Sucesso";

    else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.6 fnRxLED

Função:

bool SimpleIOClass :: fnRxLED (modo int não assinado)

Resumo: Configura o modo LED Rx. A configuração do LED Rx será armazenada na NVRAM.

Descrição: Define o modo LED Rx para um dos valores possíveis.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: modo (constante): OFF, TOGGLE, BLINKSLOW, BLINKFAST

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-6:

```
if (SimpleIOClass :: fnRxLED (BLINKFAST) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.7 fnSetBaudRate

Função:

bool SimpleIOClass :: fnSetBaudRate (BaudRateParam longo não assinado)

Resumo: Configura a taxa de transmissão padrão do dispositivo. O valor da taxa de transmissão será armazenado na NVRAM.

Descrição: Define a taxa de transmissão desejada e a armazena na NVRAM do dispositivo.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: BaudRateParam - o valor da taxa de transmissão desejada

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError. Esta função é usada apenas para definir o valor padrão da taxa de transmissão de inicialização. Quando usado com um programa de terminal, não é necessário chamar esta função para alterar a taxa de transmissão. Alterar a taxa de transmissão do programa do terminal enviará o pacote CDC apropriado que alterará a taxa de transmissão da comunicação sem a necessidade de chamar esta função.

EXEMPLO 2-7:

```
if (SimpleIOClass :: fnSetBaudRate (9600) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.8 fnSuspend

Função:

bool SimpleIOClass :: fnSuspend (unsigned int onOff)

Resumo: Configura o pino GP0 do MCP2200 para mostrar o status dos estados USB Suspend / Resume.

Descrição: Quando o GP0 é designado para mostrar os estados USB Suspend / Resume, o pino fica 'baixo' quando o estado de suspensão é emitido ou fica 'alto' quando o estado Continuar está ativado.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Ligado desligado:

- "1" GP0 refletirá os estados USB Suspend / Resume
- "0" GP0 não refletirá os estados USB Suspend / Resume (pode ser usado como GPIO)

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-8:

```
if (SimpleIOClass :: fnSuspend (1) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.9 fnTxLED**Função:**

bool SimpleIOClass :: fnTxLED (modo int não assinado)

Resumo: Configura o modo LED Tx. A configuração do LED Tx será armazenada na NVRAM.

Descrição: Define o modo LED Tx para um dos valores possíveis.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: modo (constante): OFF, TOGGLE, BLINKSLOW, BLINKFAST

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-9:

```
if (SimpleIOClass :: fnTxLED (BLINKSLOW) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.10 fnULoad**Função:**

bool SimpleIOClass :: fnULoad (unsigned int onOff)

Resumo: Configura o pino GP1 do MCP2200 para mostrar o status da configuração do USB.

Descrição: Quando o GP1 é designado para mostrar o status da configuração do USB, o pino começará 'baixo' (durante a inicialização ou após a redefinição) e ficará 'alto' depois que o MCP2200 for configurado com êxito pelo host.

Condição prévia: O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Ligado desligado:

- " 1 'GP1 refletirá o status da configuração USB
- " 0 0 'GP1 não refletirá o status da configuração USB (pode ser usado como GPIO)

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: O código de erro é retornado em LastError.

EXEMPLO 2-10:

```
if (SimpleIOClass :: fnULoad (1) == SUCESSO)
    lblStatusBar-> Text = "Sucesso";
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.11 Função GetDeviceInfo:

String ^ SimpleIOClass :: GetDeviceInfo (não assinado int uiDeviceNo)

Resumo: Retorna o nome do caminho para um dos dispositivos conectados.

Descrição: A função retornará o nome do caminho para o ID do dispositivo especificado.

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível dispositivos.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: uiDeviceNo: O ID do dispositivo para o qual as informações do caminho são necessárias. Pode ter um valor entre 0 e o número de dispositivos menos 1.

Devoluções: Esta função retorna uma string contendo o nome do caminho da ID do dispositivo fornecida.

- Caso o ID fornecido esteja fora do intervalo, a função retornará o " Erro no índice do dispositivo " corda.
- Caso o dispositivo para o qual o nome do caminho seja necessário não esteja mais conectado, a sequência de retorno será " Dispositivo não conectado ".

Observações: Nenhum.

EXEMPLO 2-11:

```
lblStatusBar-> Text = SimpleIOClass :: GetDeviceInfo (0);
```

2.3.1.12 Função GetNoOfDevices:

unsigned int SimpleIOClass :: GetNoOfDevices (void)

Resumo: A função retorna o número de dispositivos disponíveis presentes no sistema.

Descrição: A função retorna o número de dispositivos HID (com o VID / PID fornecido) conectado ao sistema.

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível dispositivos. Além disso, para saber o número real de dispositivos conectados ao sistema, ligue para o SimpleIOClass :: IsConnected () função. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Nenhum.

Devoluções: Esta função retorna o número de dispositivos HID com o VID / PID fornecido (como parâmetros do SimpleIOClass :: InitMCP2200 () função).

Observações: Ligar para SimpleIOClass :: IsConnected () antes da chamada desta função para ter o número mais recente de dispositivos presentes no sistema.

EXEMPLO 2-12:

```
SimpleIOClass :: IsConnected (); // chame esta função para atualizar o número de // dispositivos presentes no sistema
```

```
lblStatusBar-> Text = SimpleIOClass :: GetNoOfDevices ();
```

2.3.1.13 Função GetSelectedDevice:

int SimpleIOClass :: GetSelectedDevice (void)

Resumo: Retorna o ID do dispositivo selecionado.

Descrição: A função retorna o ID do dispositivo selecionado atual.

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível dispositivos. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Nenhum.

Devoluções: Esta função retorna o ID do dispositivo selecionado atual. Seu valor pode variar de 0 ao número de dispositivos menos 1.

Observações: Nenhum.

EXEMPLO 2-13:

```
lblStatusBar-> Text = SimpleIOClass :: GetSelectedDevice ();
```

2.3.1.14 Função GetSelectedDeviceInfo:

String ^ SimpleIOClass :: GetSelectedDeviceInfo (void)

Resumo: Retorna o nome do caminho do dispositivo selecionado.

Descrição: A função retorna uma sequência que contém o nome do caminho exclusivo do dispositivo selecionado.

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível

dispositivos. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Nenhum.

Devoluções: Essa função retorna uma sequência que contém o nome do caminho exclusivo do dispositivo selecionado.

Observações: O dispositivo selecionado padrão é o primeiro encontrado pela DLL. Se o usuário quiser recuperar os nomes de caminhos de outros dispositivos (supondo que mais de um dispositivo esteja presente no sistema), uma chamada para SimpleIOClass :: SelectDevice (deviceNo) É necessário.

EXEMPLO 2-14:

```
lblStatusBar-> Text = SimpleIOClass :: GetSelectedDeviceInfo (void)
```

2.3.1.15 InitMCP2200

Função:

void SimpleIOClass :: InitMCP2200 (unsigned int VendorID, unsigned int ProductID)

Resumo: Configura a classe IO simples para um fornecedor e ID do produto específicos.

Descrição: Define o ID do fornecedor e do produto usado para o projeto.

Pré-condição: Nenhuma.

Parâmetros: **1 ID do fornecedor - atribuído por USB IF (www.usb.org)**
2) ID do produto - atribuído pelo titular da ID do fornecedor

Devoluções: Nenhum.

Observações: Chame esta função antes de outras chamadas, para definir os IDs do fornecedor e do produto.

EXEMPLO 2-15:

```
InitMCP2200 (0x4D8, 0x00DF);
```

2.3.1.16 Está conectado

Função:

bool SimpleIOClass :: IsConnected ()

Resumo: Verifica com o sistema operacional se o dispositivo VID / PID atual está conectado.

Descrição: Verifica se um dispositivo MCP2200 está conectado ao computador. Nesse caso, ele retorna True; caso contrário, o resultado será falso.

Pré-condição: o VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Nenhum.

Devoluções: Verdadeiro - se pelo menos um dispositivo estiver conectado ao host. Falso - se não houver dispositivos conectados ao host.

Observações: Nenhuma comunicação real com o dispositivo final está ocorrendo. A função pergunta ao sistema operacional se o VID / PID especificado foi enumerado.

EXEMPLO 2-16:

```
int rv não assinado;  
if (SimpleIOClass :: IsConnected ()) {lblStatusBar-> Text = "Dispositivo conectado";  
  
} else lblStatusBar-> Text = "Dispositivo desconectado";
```

2.3.1.17 Função ReadEEPROM:

int SimpleIOClass :: ReadEEPROM (não assinado int uiEEPAddress)

Resumo: Lê um byte da EEPROM.

Descrição: Lê um byte da EEPROM no endereço fornecido.

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível dispositivos. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: uiEEPAddress - o local do endereço EEPROM no qual precisamos escrever (deve ser de 0 a 255, inclusive).

Devoluções: Esta função retorna qualquer valor positivo como sendo o valor da localização da EEPROM:

- E_WRONG_ADDRESS (-3) - caso o endereço EEPROM fornecido esteja fora da faixa
- E_CANNOT_SEND_DATA (-4) - caso a função não possa enviar o comando para o dispositivo

Observações: Nenhum.

EXEMPLO 2-17:

```
int iRetVal = SimpleIOClass :: ReadEEPROM (0x01);  
if (iRetVal >= 0)  
{lblStatusBar-> Text = "Sucesso";  
  
} else lblStatusBar-> Text = "Erro ao ler na EEPROM" + SimpleIOClass :: LastError;
```

2.3.1.18 Função ReadPin:

bool SimpleIOClass :: ReadPin (pino int não assinado, int não assinado * valor de retorno)

Resumo: Lê o pino especificado.

Descrição: Lê o pino especificado e retorna o valor em valor de retorno. Se o pino tiver sido configurado como uma entrada digital, o valor de retorno será '0' ou '1'.

Pré-condição: deve ser configurado anteriormente como uma entrada por meio de um ConfigureIO ligar.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros:

- PIN - o número do pino a ser definido (0-7)
- valor de retorno - o valor lido no pino ('0' ou '1')

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Nenhum.

EXEMPLO 2-18:

```
int rv não assinado;  
if (SimpleIOClass :: ReadGPIO (0, & rv)) {lblStatusBar-> Text = "Êxito";  
  
} else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.19 Função ReadPinValue:

int SimpleIOClass :: ReadPinValue (pino int não assinado)

Resumo: Lê o pino especificado.

Descrição: Lê o pino especificado e retorna o valor como o valor de retorno. Se o pino tiver sido configurado como uma entrada digital, o **valor de retorno será '0' ou '1'**. Se ocorrer um erro, a função retornará um valor de 0x8000.

Pré-condição: deve ser configurado anteriormente como uma entrada por meio de um ConfigureIO ligar.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: PIN - o número do pino a ser definido (0-7)

Devoluções: Esta função retorna o valor de leitura do pino ou retorna um valor de 0x8000, se ocorrer um erro.

Observações: Nenhum.

EXEMPLO 2-19:

```
int rv não assinado;  
if (SimpleIOClass :: ReadPinValue (0) != 0x8000) {lblStatusBar-> Text = "Êxito";  
  
} else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.20 Função ReadPort:

bool SimpleIOClass :: ReadPort (unsigned int * returnvalue)

Resumo: Lê a porta GPIO como entrada digital.

Descrição: Lê a porta GPIO e retorna o valor em valor de retorno. Isso fornece um meio de ler todos os pinos simultaneamente, em vez de um por um.

Pré-condição: deve ser configurado anteriormente como uma entrada por meio de um ConfigureIO ligar.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros:

- PIN - o número do pino a ser definido (0-7)
- **valor de retorno - o valor lido no pino ('0' ou '1')**

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Os pinos configurados para saída retornam o estado atual da porta. Os pinos configurados como entrada são lidos como zero.

EXEMPLO 2-20:

```
int rv não assinado;  
if (SimpleIOClass :: ReadGPIOPort (0, & rv)) {lblStatusBar-> Text = "Êxito";  
  
} else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.21 Função ReadPortValue:

int SimpleIOClass :: ReadPortValue ()

Resumo: Lê a porta GPIO como entrada digital.

Descrição: Lê a porta GPIO e retorna o valor da porta. Isso fornece um método para ler todos os pinos simultaneamente, em vez de um por um. Em caso de erro, o valor retornado será 0x8000.

Pré-condição: deve ser configurado anteriormente como uma entrada por meio de um ConfigureIO ligar.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: Nenhum.

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Os pinos configurados para saída retornam o estado atual da porta. Os pinos configurados como entrada são lidos como zero.

EXEMPLO 2-21:

```
int rv;
rv = SimpleIOClass :: ReadPortValue () if (rv! = 0x8000)

{lblStatusBar-> Text = "Sucesso";

} else{lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.22 Função SelectDevice:

int SimpleIOClass :: SelectDevice (não assinado int uiDeviceNo)

Resumo: Seleciona um dos dispositivos ativos no sistema.

Descrição: A função é usada para selecionar um dos dispositivos detectados no sistema como o "dispositivo ativo".

Pré-condição: pelo menos uma chamada para o InitMCP2200 () é necessário para iniciar uma pesquisa DLL para o compatível dispositivos. Além disso, para saber o número real de dispositivos no sistema, ligue para o

SimpleIOClass :: IsConnected () função. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para InitMCP2200 (VID, PID).

Parâmetros: uiDeviceNo - o ID do dispositivo a ser selecionado (pode ter um valor entre 0 e o número de dispositivos menos 1).

Devoluções: Esta função retorna '0' o 'em caso de sucesso na seleção, caso contrário, retornará:

- E_WRONG_DEVICE_ID (-1) para um ID de dispositivo fora do intervalo
- E_INACTIVE_DEVICE (-2) para um dispositivo inativo.

Observações: Ligar para SimpleIOClass :: IsConnected () antes da chamada desta função para ter o número mais recente de dispositivos presentes no sistema.

EXEMPLO 2-22:

```
int iResult;
iResult = SimpleIOClass :: SelectDevice (1) if (iResult == 0)

{lblStatusBar-> Text = "Sucesso";

} else{lblStatusBar-> Text = "Erro ao selecionar o dispositivo";
```

2.3.1.23 Função SetPin:

bool SimpleIOClass :: SetPin (pino int não assinado)

Resumo: Define o pino especificado.

Descrição: Define o pino especificado como lógico '1'.

Pré-condição: deve ser configurado anteriormente como uma saída por meio de um `ConfigureIO` ou `ConfigureIODefaultOutput` ligar.

O VID e o PID devem ser definidos anteriormente por meio de uma chamada para `InitMCP2200` (VID, PID).

Parâmetros: PIN - o número do pino a ser definido (0-7)

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Nenhum.

EXEMPLO 2-23:

```
if (SimpleIOClass :: SetPin (2)) {lblStatusBar-> Text = "Êxito";

} else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```

2.3.1.24 Função WriteEEPROM:

int SimpleIOClass :: WriteEEPROM (não assinado int uiEEPAddress, char não assinado ucValue)

Resumo: Grava um byte na EEPROM do dispositivo MCP2200.

Descrição: grava um byte no endereço fornecido na EEPROM interna de 256 bytes. Pré-condição: pelo menos uma chamada para o `InitMCP2200`

() é necessário para iniciar uma pesquisa DLL para o

dispositivos compatíveis. O VID e o PID devem ser definidos anteriormente por meio de uma chamada para `InitMCP2200` (VID, PID).

Parâmetros:

- `uiEEPAddress` - o local do endereço EEPROM no qual gravar (deve ser de 0 a 255, inclusive).

- `ucValue` - o valor de byte necessário para gravar no local especificado.

Devoluções: Esta função retorna '0' se o comando write foi enviado com sucesso ao dispositivo, caso contrário, ele retornará:

- `E_WRONG_ADDRESS` (-3) caso o endereço EEPROM fornecido esteja fora da faixa
- `E_CANNOT_SEND_DATA` (-4) caso a função não possa enviar o comando para o dispositivo.

Observações: A função enviará o comando write EEPROM, mas não tem confirmação se o local da EEPROM foi realmente gravado. Para verificar a correção da gravação da EEPROM, o usuário pode emitir um `SimpleIOClass :: ReadEEPROM ()` e verifique se o valor retornado corresponde ao valor escrito.

EXEMPLO 2-24:

```
int iRetVal = SimpleIOClass :: WriteEEPROM (0x01, 0xAB);

if (iRetVal == 0)
{lblStatusBar-> Text = "Sucesso";

} else lblStatusBar-> Text = "Erro ao gravar na EEPROM" + SimpleIOClass :: LastError;
```

2.3.1.25 Função WritePort:

bool SimpleIOClass :: WritePort (não assinado int portValue)

Resumo: Grava um valor na porta GPIO.

Descrição: grava a porta GPIO. Isso fornece um meio de escrever todos os pinos simultaneamente, em vez de um por um. Pré-condição: deve ser configurado anteriormente como uma saída por meio de um `ConfigureIO` ligar. VID e PID devem ser previamente

definido através de uma chamada para `InitMCP2200` (VID, PID).

Parâmetros: portValue - valor de byte a ser definido na porta.

Devoluções: Esta função retorna True se a transmissão for bem sucedida e retorna False se a transmissão falhar.

Observações: Nenhum.

EXEMPLO 2-25:

```
if (SimpleIOClass :: WritePort (0x5A))
{
    lblStatusBar-> Text = "Sucesso";

}
else lblStatusBar-> Text = "Comando inválido" + SimpleIOClass :: LastError;
```


NOTAS:

MCP2200

3.0 características elétricas

Classificações máximas absolutas (†)

| | | | | | |
|---|----------------------------------|--|---------------------|--|-----------------------------------|
| Temperatura ambiente sob viés | –40 ° C a + 85 ° C | Temperatura de armazenamento | –65 ° C a + 150 ° C | Tensão em V _{DD} em relação a V _{SS} | –0,3V a + 9,0V |
| Tensão de –0,3V a + 6,0V no MCLR em relação a V _{SS} | –0,3V a + 9,0V | Tensão em V _{USB} (1) pino em relação a V _{SS} | –0,3V a + 4,0V | Tensão nos pinos D+ e D– em relação a V _{SS} | –0,3V a (V _{USB} + 0.3V) |
| Tensão em todos os outros pinos em relação a V _{SS} | –0,3V a (V _{DD} + 0.3V) | Dissipação de energia total (2) | 800 mW | Corrente máxima fora de V _{SS} PIN | 95 mA |
| Corrente máxima em V _{DD} PIN | 95 mA | Corrente de aperto, I _K (V _{PIN} < 0 ou V _{PIN} > V _{DD}) | 20 mA | Corrente máxima de saída afundada por qualquer pino de E / S | 25 mA |
| Corrente de saída máxima fornecida por qualquer pino de E / S | 25 mA | Corrente máxima afundada por todas as portas | 90 mA | Corrente máxima fornecida por todas as portas | 90 mA |

Nota 1: V_{USB} deve ser sempre • V_{DD} + 0.3V.

2: A dissipação de energia é calculada da seguinte forma: $P_{DIS} = V_{DD} I_{DD} + E_{UOH} + ((V_{DD} - V_{OH}) I_{OH}) + (V_{OL} I_{OL})$.

† **AVISO PRÉVIO:** As tensões acima das listadas em "Classificações máximas absolutas" podem causar danos permanentes ao dispositivo. Esta é apenas uma classificação de tensão e a operação funcional do dispositivo nessas ou em quaisquer outras condições acima daquelas indicadas nas listagens de operação desta especificação não está implícita. A exposição acima das condições máximas de classificação por períodos prolongados pode afetar a confiabilidade do dispositivo.

3.1 Características da CC

| Características da CC | | Condições de operação (salvo indicação em contrário): 3.0V ≤ V _{DD} ≤ 5.5V a −40 °C ≤ T _{UMA} ≤ + 85 °C (I-Temp) | | | | | |
|---|--|---|-------------------------|-------------------|---------------------|----------|---|
| Param. Não. | Característica | Sym. | Min. | Typ. | Máx. | Unidades | Condições |
| T001 | Tensão de alimentação | V _{DD} | 3.0 | - | 5,5 V | | |
| | Tensão de liberação de redefinição de inicialização | V _{POR} | | 1.6 | | V | |
| | Redefinição na inicialização Rearm Voltage | | | 0,8 | | V | |
| D003 | Taxa de aumento para Garantir reinicialização na inicialização | SV _{DD} | 0,05 | - | - | V / ms | somente orientação de projeto, Não testado |
| Corrente de alimentação D004 | | Eu _{DD} | | | | | |
| | V _{DD} = 3.0V | | - | 10 | 12 | mA | F _{OSC} = 12 MHz, |
| | V _{DD} = 5.0V | | - | 13 | 15 | mA | (330 nF em V _{USB}) |
| Corrente em espera D005 | | Eu _{DOS} | - | 9 | - | µA | |
| Baixa tensão de entrada | | | | | | | |
| D031 | Schmitt Trigger (GPIO) | V _{IL} | - | - | 0,2 V _{DD} | V | 3.0V ≤ V _{DD} ≤ 5.5V |
| | TTL (pino CTS) | | - | - | 0.8 V | | 4.5V ≤ V _{DD} ≤ 5.5V |
| Alta tensão de entrada | | | | | | | |
| D041 | Schmitt Trigger (GPIO) | V _{IH} | 0.8 V _{DD} | - V _{DD} | | V | 3.0V ≤ V _{DD} ≤ 5.5V |
| | TTL (pino RTS) | | 2.0 | - V _{DD} | | V | 4.5V ≤ V _{DD} ≤ 5.5V |
| Vazamento de entrada Corrente | | | | | | | |
| D060 | GPIO, CTS | Eu _{IL} | - | ± 50 | ± 100 | nA | V _{SS} ≤ V _{PIN} ≤ V _{DD} , pino em Z alto |
| | RST | | - | ± 50 | ± 200 | n / D | |
| | OSC1 | | - | ± 50 | ± 100 | n / D | |
| Baixa tensão de saída | | | | | | | |
| GP0 D080, UART Tx / Rx | | V _{OL} | - | - | 0,6 V | | Eu _{OL} = 8,0 mA, V _{DD} = 5.0V |
| | | | - | - | 0,6 V | | Eu _{OL} = 6,0 mA, V _{DD} = 3.3V |
| Saída de alta tensão | | | | | | | |
| GP0 D090, UART Tx / Rx | | V _{OH} | V _{DD} - 0,7 - | | - V | | Eu _{OH} = −3,5 mA, V _{DD} = 5.0V |
| | | | V _{DD} - 0,7 - | | - V | | Eu _{OH} = −3,0 mA, V _{DD} = 3.3V |
| Especificações de carga capacitiva nos pinos de saída | | | | | | | |
| D101 | OSC2 | C _{OSC2} | - | - | 15 | pF | Nota 1 |
| D102 | GPIO | C _{IO} | - | - | 50. | pF | Nota 1 |

Nota 1: Este parâmetro é caracterizado, mas não testado.

FIGURA 3-1: POR E POR REARM COM AUMENTO LENTO V_{DD}

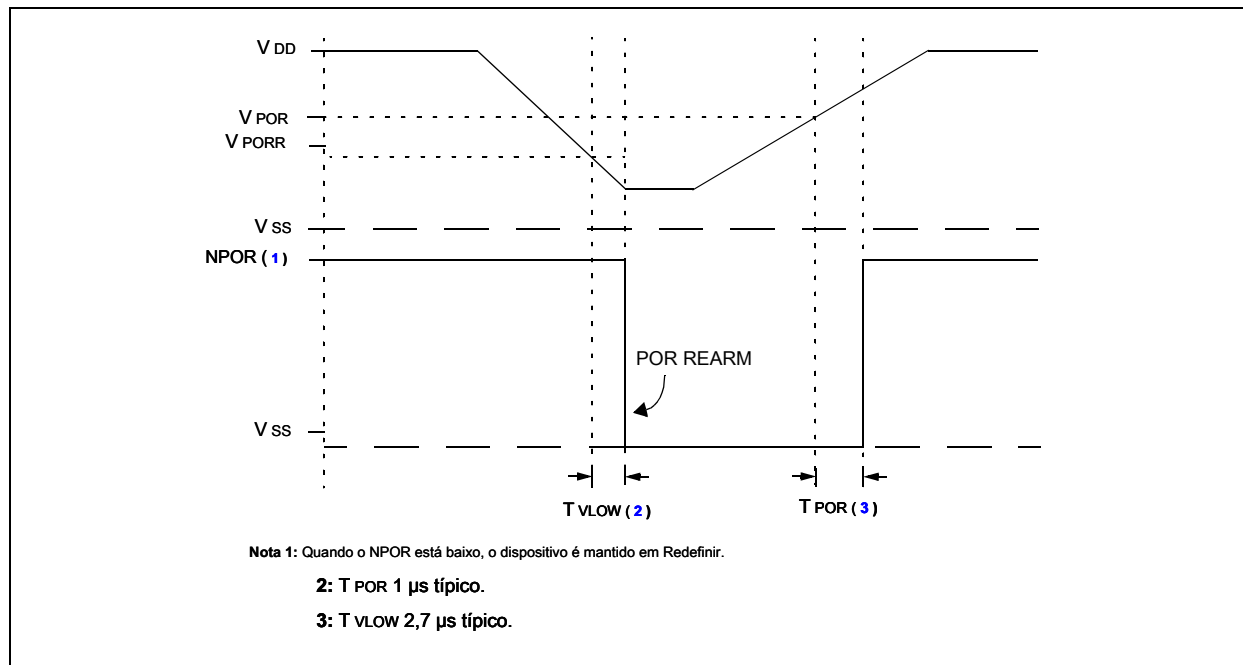


TABELA 3-1: ESPECIFICAÇÕES DO MÓDULO USB

| Características da CC | | Condições de operação (salvo indicação em contrário): 3.0V -- V _{DD} • 5.5V a –40 • C • T _{UMA} • + 85 • C (I-Temp) | | | | | |
|-----------------------|--|--|------|------|-------|----------|--|
| Param. Não. | Característica | Sym. | Min. | Typ. | Máx. | Unidades | Condições |
| D313 | Tensão de Entrada USB para buffer USB | V _{USB} | 3.0 | - | 3.6 | A | tensão V no pino Vusb deve ser nesta faixa para operação USB adequada |
| D314 | Vazamento de entrada no pino | I _{II} | - | - | ± 1 | µA | V _{SS} -- V _{PIN} -- V _{DD} fixar em • alta impedância |
| | Baixa Tensão de Entrada D315 para buffer USB | V _{IUSB} | - | - | 0.8 V | | Para a gama Vusb |
| | Alta tensão de entrada D316 para buffer USB | V _{IHUSB} | 2.0 | - | - V | | Para a gama Vusb |
| | Entrada Diferencial D318 Sensibilidade | V _{difs} | - | - | 0,2 V | | A diferença entre D + e D- deve exceder esse valor enquanto V _{cm} é atingido |
| | D319 Diferencial Comum Faixa de modo | V _{cm} | 0,8 | - | 2,5 V | | |
| | Saída do Driver D320 Impedância (1) | Z _{out} | 28. | - | 44 | • | |
| | Saída de tensão D321 baixa | V _{ol} | 0,0 | - | 0,3 V | | 1,5 k •• carga conectada a 3.6V |
| | Saída de tensão D322 alta | V _{oh} | 2.8 | - | 3,6 V | | 1,5 k •• carga conectada ao terra |

Nota 1: As linhas de sinal D + e D - foram resistências integradas de impedância. Sem resistores externos, capacitores ou componentes magnéticos são necessários nos caminhos de sinal D + / D - entre o dispositivo da família MCP2200 e o cabo USB.

TABELA 3-2: CONSIDERAÇÕES TÉRMICAS

| Condições operacionais padrão (salvo indicação em contrário) | | | | | |
|---|----------------------|---|-------|---|-----------|
| Temperatura de operação: -40 • C • T _{UMA} • + 85 • C (I-Temp) | | | | | |
| Param. Não. | Sym. | Característica | Typ. | Unidades | Condições |
| TH01 | θ_{JA} | Junção de resistência térmica ao ambiente | 36,1 | • Embalagem C / W 20 pinos VQFN 5x5 mm | |
| | | | 85,2 | • Pacote SOIC de 20 pinos C / W | |
| | | | 108,1 | • Pacote SSOP de 20 pinos C / W | |
| TH02 | θ_{JC} | Junção da resistência térmica ao caso | 1,7 | • C / W VQFN de 20 pinos, pacote 5x5 mm 24 | |
| | | | | • Pacote SOIC de 20 pinos C / W 24 | |
| | | | | • Pacote SSOP de 20 pinos C / W | |
| TH03 | T _{JMAX} | Temperatura máxima da junção | 150 | • C | |
| TH04 | PD | Dissipação de energia | - | W PD = P _{INTERNO} + P _{I/O} | |
| TH05 | P _{INTERNO} | Dissipação de energia interna | - | WP _{INTERNO} = Eu _{DD} x V _{DD} (1) | |
| TH06 | P _{I/O} | Dissipação de energia de E / S | - | WP _{E/S} = • (Eu _{OL} • V _{OL}) + • (Eu _{OH} • (V _{DD} - V _{OH})) | |
| TH07 | P _{DER} | Potência reduzida | - | WP _{DER} = PD _{MAX} (T _J - T _{UMA}) / θ_{JA} (2, 3) | |

Nota 1: Eu_{DD} é a corrente para executar o chip sozinho, sem sobrecarregar os pinos de saída.

2: T_A = Temperatura ambiente.

3: T_J = Temperatura da junção.

3.2 Características da CA

3.2.1 SIMBOLOGIA DOS PARÂMETROS DE TEMPO

Os símbolos dos parâmetros de temporização foram criados em um dos seguintes formatos:

| 1. TppS2ppS | 2. TppS | | | | | | | | | | | | | | | | |
|---|--|--------------------------|----|------------|--------|----------------------------|--|--|--|-----|-----------|----|-----------|-----|-----------|---|-----------------|
| <table><tr><td>T</td><td></td></tr><tr><td>F</td><td>Frequência</td></tr><tr><td>E</td><td>Erro</td></tr></table> | T | | F | Frequência | E | Erro | <table><tr><td>T</td><td>Tempo</td></tr></table> | T | Tempo | | | | | | | | |
| T | | | | | | | | | | | | | | | | | |
| F | Frequência | | | | | | | | | | | | | | | | |
| E | Erro | | | | | | | | | | | | | | | | |
| T | Tempo | | | | | | | | | | | | | | | | |
| Letras minúsculas (pp) e seus significados: pp | | | | | | | | | | | | | | | | | |
| <table><tr><td>io</td><td>Pino de entrada ou saída</td></tr><tr><td>rx</td><td>Receber</td></tr><tr><td>bitclk</td><td>RX / TX BITCLK</td></tr><tr><td>drt</td><td>Temporizador de reinicialização do dispositivo</td></tr></table> | io | Pino de entrada ou saída | rx | Receber | bitclk | RX / TX BITCLK | drt | Temporizador de reinicialização do dispositivo | <table><tr><td>osc</td><td>Oscilador</td></tr><tr><td>tx</td><td>Transmite</td></tr><tr><td>RST</td><td>Redefinir</td></tr></table> | osc | Oscilador | tx | Transmite | RST | Redefinir | | |
| io | Pino de entrada ou saída | | | | | | | | | | | | | | | | |
| rx | Receber | | | | | | | | | | | | | | | | |
| bitclk | RX / TX BITCLK | | | | | | | | | | | | | | | | |
| drt | Temporizador de reinicialização do dispositivo | | | | | | | | | | | | | | | | |
| osc | Oscilador | | | | | | | | | | | | | | | | |
| tx | Transmite | | | | | | | | | | | | | | | | |
| RST | Redefinir | | | | | | | | | | | | | | | | |
| Letras maiúsculas e seus significados: S | | | | | | | | | | | | | | | | | |
| <table><tr><td>F</td><td>Outono</td></tr><tr><td>H</td><td>Alto</td></tr><tr><td>Eu</td><td>Inválido (alta impedância)</td></tr><tr><td>eu</td><td>Baixo</td></tr></table> | F | Outono | H | Alto | Eu | Inválido (alta impedância) | eu | Baixo | <table><tr><td>P</td><td>Período</td></tr><tr><td>R</td><td>Subir</td></tr><tr><td>V</td><td>Válido</td></tr><tr><td>Z</td><td>Alta impedância</td></tr></table> | P | Período | R | Subir | V | Válido | Z | Alta impedância |
| F | Outono | | | | | | | | | | | | | | | | |
| H | Alto | | | | | | | | | | | | | | | | |
| Eu | Inválido (alta impedância) | | | | | | | | | | | | | | | | |
| eu | Baixo | | | | | | | | | | | | | | | | |
| P | Período | | | | | | | | | | | | | | | | |
| R | Subir | | | | | | | | | | | | | | | | |
| V | Válido | | | | | | | | | | | | | | | | |
| Z | Alta impedância | | | | | | | | | | | | | | | | |

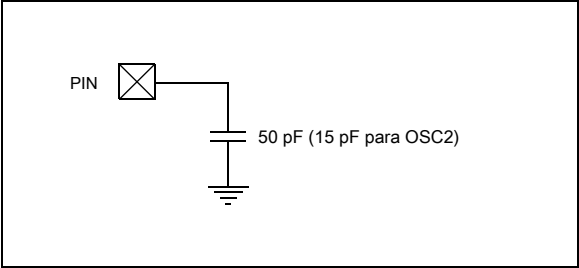
3.2.2. CONDIÇÕES DE TEMPO

A temperatura e tensão de operação especificadas em [Quadro 3-3](#) aplicam-se a todas as especificações de tempo, salvo indicação em contrário. [Figura 3-2](#) especifica as condições de carga para as especificações de tempo.

TABELA 3-3: ESPECIFICAÇÕES DE TEMPERATURA E TENSÃO - CA

| | |
|--------------------|---|
| CARACTERÍSTICAS AC | Condições operacionais padrão (salvo indicação em contrário) |
| | Temperatura de operação: -40°C • T_{UMA} • $+85^{\circ}\text{C}$ Tensão operacional V _{DD} conforme descrito na especificação DC, |
| | Seção 3.1 "Características da CC" . |

FIGURA 3-2: CONDIÇÕES DE CARGA PARA ESPECIFICAÇÕES DE TEMPO DE DISPOSITIVO



3.2.3. ESPECIFICAÇÕES DE TEMPO

TABELA 3-4: RESET, PARÂMETROS DO TEMPORIZADOR DE INICIALIZAÇÃO E DO TIMER DE OSCILADOR

| Condições operacionais padrão (salvo indicação em contrário) | | | | | | |
|--|--------|-------------------------------------|-----------------|----------|----------|-----------|
| Temperatura de operação: -40 ° C ** T UMA ** + 85 ° C | | | | | | |
| Param. Não. (1) | Sym. | Característica | Min. Typ. (2) | Máx. | Unidades | Condições |
| 30 | T RST | Largura de pulso MCLR (baixa) | 2 | - | - µs | |
| 31 | T PWRT | Temporizador de inicialização | 40. | 65 | 140 ms | |
| 32. | T OST | Tempo de inicialização do oscilador | - | 1024 - T | OST | |

Nota 1: Esses parâmetros são caracterizados mas não testados.

2: Dados em "Typ". a coluna está a 5V, 25 ° C, a menos que seja indicado Esses parâmetros são para orientação do projeto somente e não são testados.

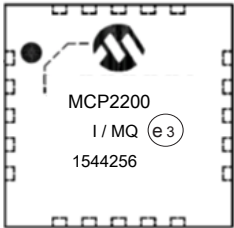
4.0 INFORMAÇÃO DA EMBALAGEM

4.1 Informações sobre a marcação do pacote

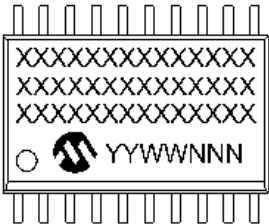
VQFN de 20 derivações (05x05 mm)



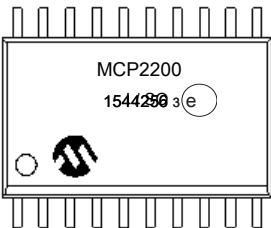
Exemplo:



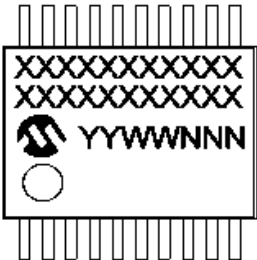
SOIC de 20 derivações



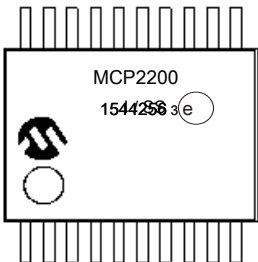
Exemplo:



SSOP de 20 derivações



Exemplo:



Lenda: XX ... X Informações específicas do cliente Y

Código do ano (último dígito do ano civil) AA

Código do ano (últimos 2 dígitos do ano civil) Código da semana da

WW (a semana de 1º de janeiro é a semana '01') NNN Código de rastreabilidade

alfanumérica

(e 3)

JEDEC sem Pb ® designador para Matte Tin (Sn)

*

Este pacote é livre de Pb. O JEDEC sem Pb ® designador (

(e 3))

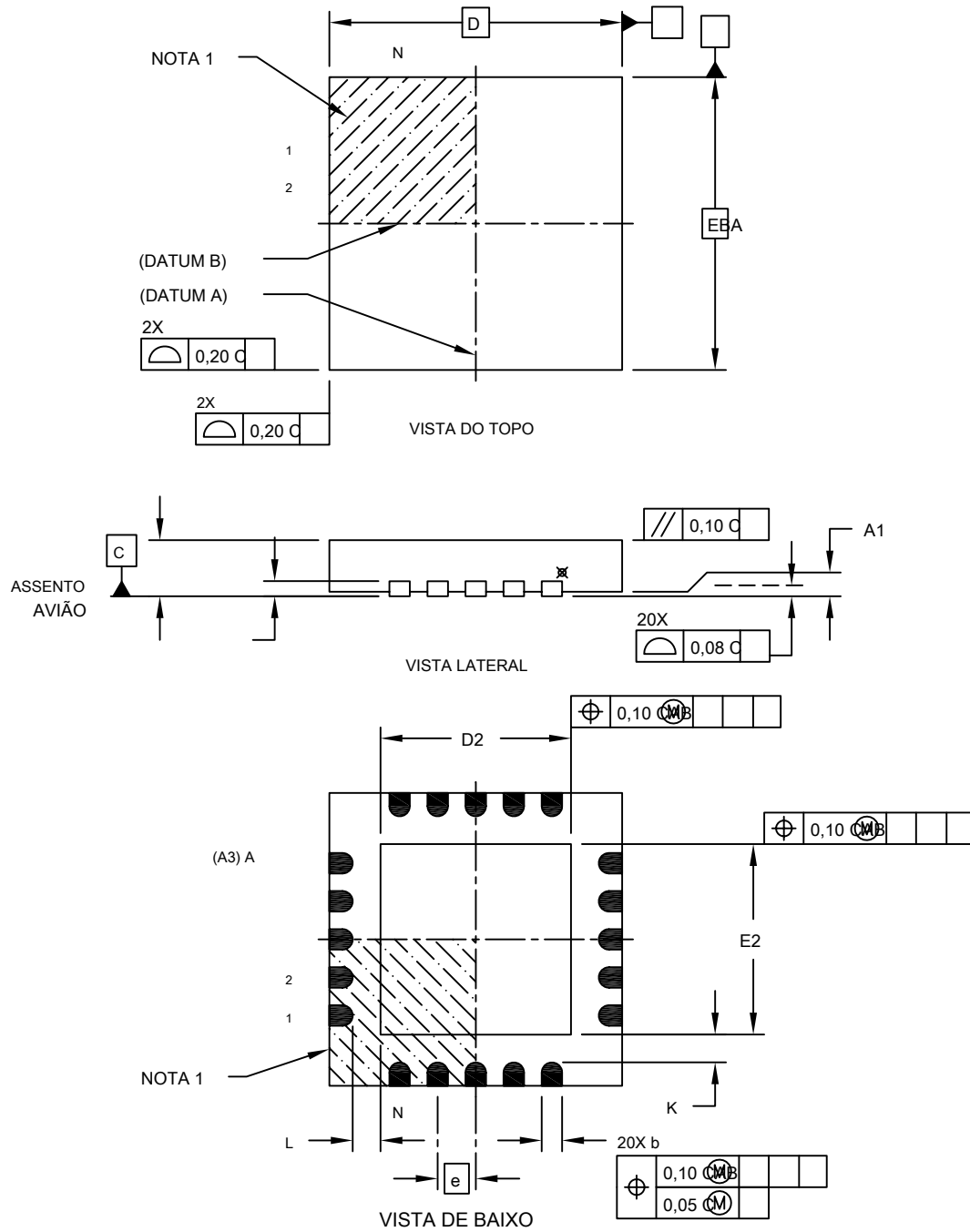
pode ser encontrada na embalagem externa deste pacote.

Nota: Caso o número completo da peça do Microchip não possa ser marcado em uma linha, ele será transferido para a próxima linha, limitando o número de caracteres disponíveis para informações específicas do cliente.

Quadrilátero de plástico de 20 derivações, sem pacote de derivação (MQ) - corpo de 5x5x1,0 mm [VQFN]

Com 0,40 mm de comprimento de contato

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



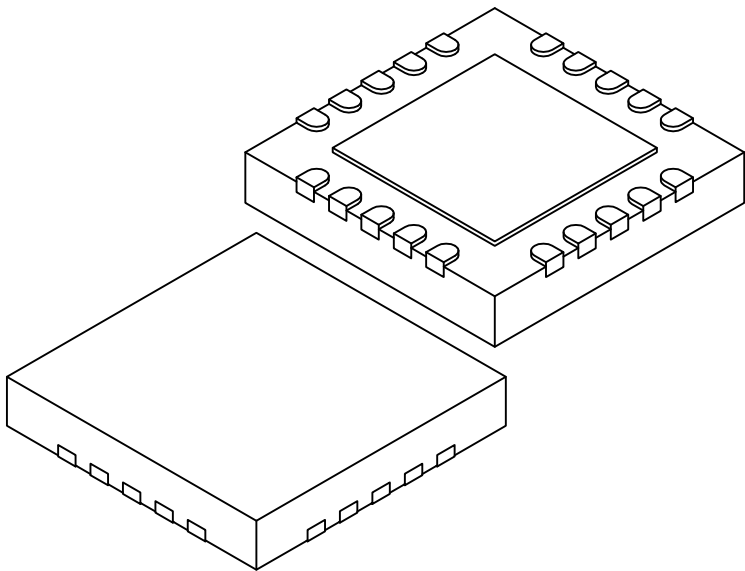
Desenho de tecnologia de microchip C04-139C (MQ) Folha 1 de 2

MCP2200

Quadrilátero de plástico de 20 derivações, sem pacote de derivação (MQ) - corpo de 5x5x1,0 mm [VQFN]

Com 0,40 mm de comprimento de contato

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



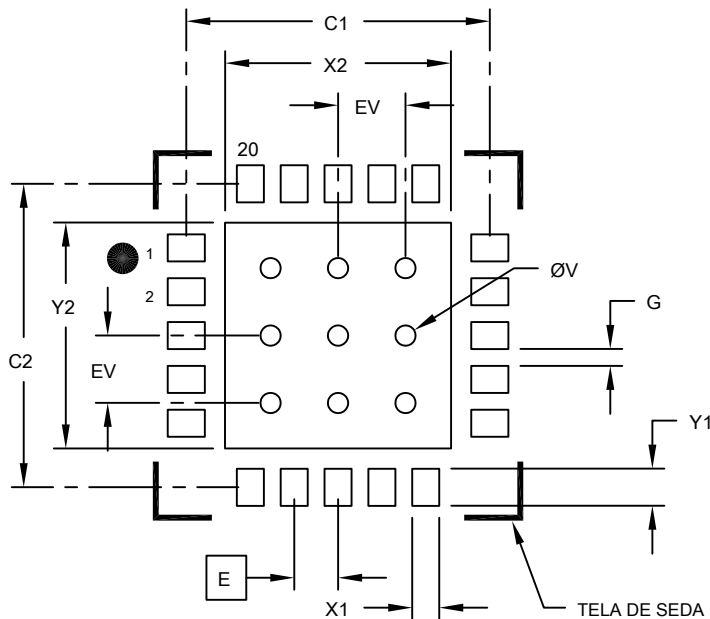
| Unidades | | MILÍMETROS | | |
|-----------------------------------|---|------------|------|------|
| Limites de dimensão | | MIN | NOM | MAX |
| | N | 20 | | |
| altura | | 0,65 BSC | | |
| separador Número de terminais | | | | |
| Passo total da | | 0,80 | 0,90 | 1,00 |
| EspeSSura do contato do | ² ₃ A ¹ | 0,00 | 0,02 | 0,05 |
| | | 0,20 REF | | |
| total | D | 5,00 BSC | | |
| Comprimento do contato | D2 | 3,15 | 3,25 | 3,35 |
| Largura total Largura da | E | 5,00 BSC | | |
| | E2 | 3,15 | 3,25 | 3,35 |
| | b | 0,25 | 0,30 | 0,35 |
| exposta Comprimento do contato | LK | 0,35 | 0,40 | 0,45 |
| Largura do contato com a almofada | | 0,20 | - | - |

- Notas:
- 1 O recurso de índice visual do pino 1 pode variar, mas deve estar localizado dentro da área hachurada. O pacote é visto em
 - 2) separado
 - 3) Dimensionamento e tolerância conforme ASME Y14.5M
- básica. Teoricamente valor exato mostrado, sem tolerâncias.
- REF: dimensão de referência, geralmente sem tolerância, apenas para fins informativos. BSC: dimensão

Quadrilátero de plástico de 20 derivações, sem pacote de derivação (MQ) - corpo de 5x5x1,0 mm [VQFN]

Com 0,40 mm de comprimento de contato

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



PADRÃO DE TERRA RECOMENDADO

| Unidades | | MILÍMETROS | | |
|--|----|------------|------|------|
| Limites de dimensão | | MEU | NOM | MAX |
| Distância do contato | | 0,65 BSC | | |
| Comprimento da almofada de contato | W2 | | | 3,35 |
| Espaçamento da almofada de contato | T2 | | | 3,35 |
| Espaçamento da almofada de contato | C1 | | 4,50 | |
| | C2 | | 4,50 | |
| Largura da almofada de contato (X20) | X1 | | | 0,40 |
| Comprimento da almofada de contato (X20) | Y1 | | | 0,55 |
| Distância Entre Almofadas | G | 0,20 | | |
| Via térmica diâmetro | V | | 0,30 | |
| Thermal Via Pitch | EV | | 1,00 | |

- 1 Dimensionamento e tolerância conforme ASME Y14.5M
- BSC: dimensão básica. Teoricamente valor exato mostrado, sem tolerâncias. Notas:
- 2) Para obter melhores resultados de solda, as vias térmicas, se usadas, devem ser preenchidas ou montadas em barracas para evitar a perda de solda durante o processo de reflow

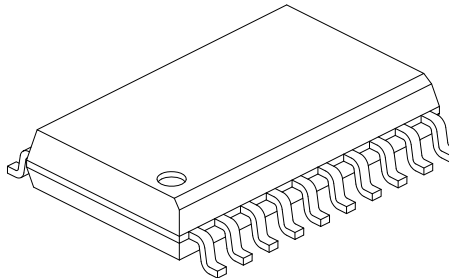
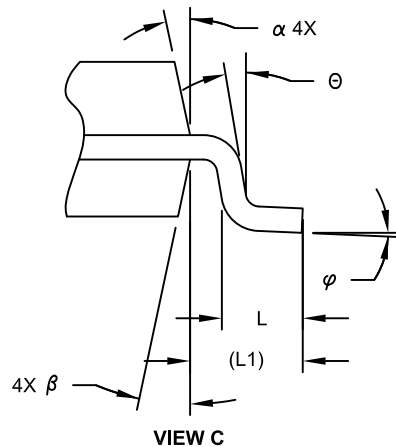
Desenho de tecnologia de microchip C04-2139B (MQ)

| | |
|--------------|--|
| Nota: | Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em http://www.microchip.com/packaging |
|--------------|--|



20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



| Units | | MILLIMETERS | | |
|--------------------------|----------|-------------|-----|------|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Pins | N | 20 | | |
| Pitch | e | 1.27 BSC | | |
| Overall Height | A | - | - | 2.65 |
| Molded Package Thickness | A2 | 2.05 | - | - |
| Standoff § | A1 | 0.10 | - | 0.30 |
| Overall Width | E | 10.30 BSC | | |
| Molded Package Width | E1 | 7.50 BSC | | |
| Overall Length | D | 12.80 BSC | | |
| Chamfer (Optional) | h | 0.25 | - | 0.75 |
| Foot Length | L | 0.40 | - | 1.27 |
| Footprint | L1 | 1.40 REF | | |
| Lead Angle | θ | 0° | - | - |
| Foot Angle | ϕ | 0° | - | 8° |
| Lead Thickness | c | 0.20 | - | 0.33 |
| Lead Width | b | 0.31 | - | 0.51 |
| Mold Draft Angle Top | α | 5° | - | 15° |
| Mold Draft Angle Bottom | β | 5° | - | 15° |

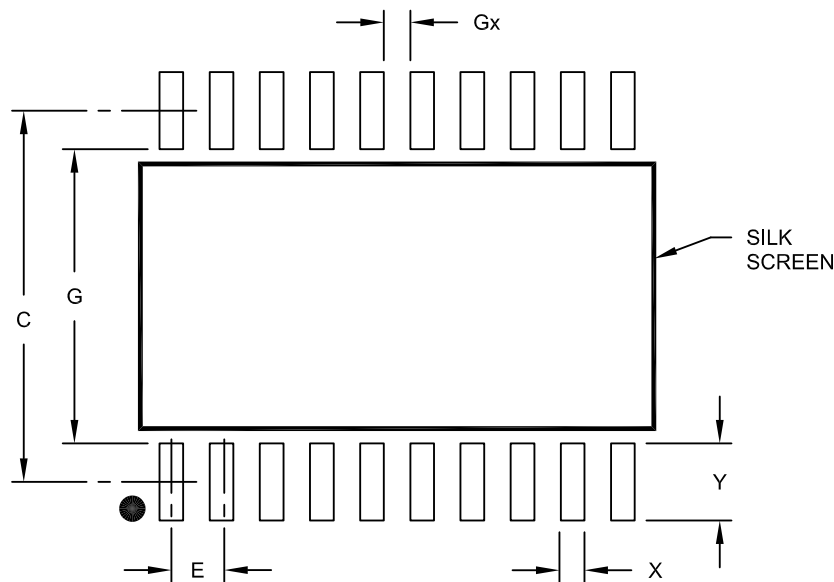
Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.
- Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 of 2

20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|--------------------------|----|-------------|------|------|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | 1.27 BSC | | |
| Contact Pad Spacing | C | | 9.40 | |
| Contact Pad Width (X20) | X | | | 0.60 |
| Contact Pad Length (X20) | Y | | | 1.95 |
| Distance Between Pads | Gx | 0.67 | | |
| Distance Between Pads | G | 7.45 | | |

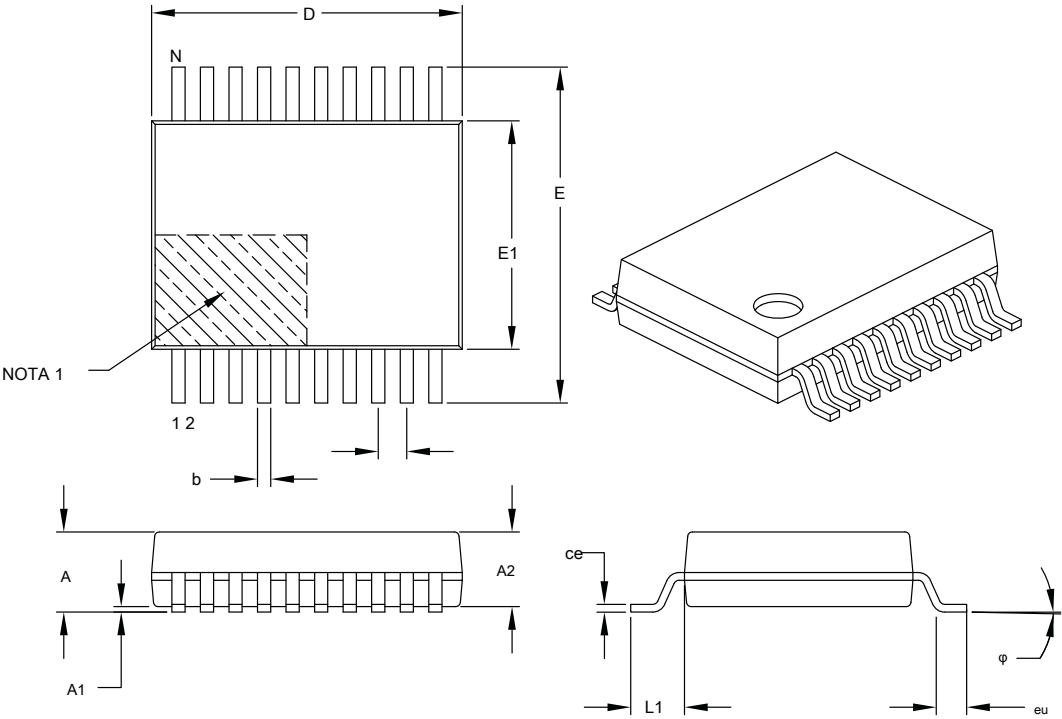
Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

! ??? "

| | | | |
|------|-----------------------|--------------------|-------------------------------|
| # \$ | 1 ???%? &? % ?! ? ? ? | %? ? 2? "(?) ?? '? | %? ? 2? ? \$???% ??? % "??%? |
| | %% 033)))? & ???? | & 3 ?? 2? ? | |



| | | | | |
|-------------------|--------|---------|-----|------|
| | 4 ???% | 55 ?? | | |
| &? ? | 5? &?% | 6 | 6?? | 8 |
| 6! & (??? \$?? ? | 6 | 9-?.? / | | |
| % ?? | | ; | | |
| ?? ?? :? ? % | | 9- | - | <- |
| "" ???? 2? 2? | | - | ; | ; |
| % ?? "? \$\$\$? | | * | <? | <??? |
| ?? ?? =? "%? | * | - | - + | 9? |
| "" ???? 2? =? "%? | | 9 ??? | | - |
| ?? ?? 5? %? | 5 | - - | - | - |
| 1 ???%? 5? %? | 5? | - * 1 | | |
| 1 ???%? ? % | | | ; | - |
| 5? "??? | 2? | | > | <> |
| 1 ???% ??? | | (| ; | +< |
| 5? "? =? "%? | | | | |

\$

1 ????? "#? \$? % !? ? & ????? ' (!%? & !%? (??? % ")?% ??? %? % ?? "???

&? ? "? * ?? "???? % ??? ? ! " ? & ?? " ? \$?? % ? ! ??? " ? \$?? % ? ! ??? %? #? "???? & & ? ? " ?

+ &? ? "?% ?? * , -

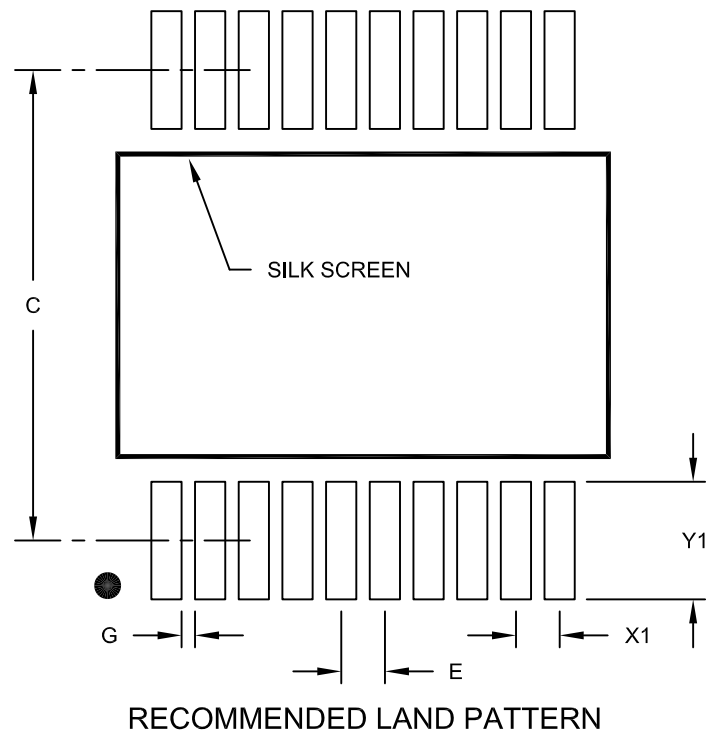
. / 0. ?? &? ? % ??? #% ??? ! ? ?) % ?? ! % ? % ? ?

10? \$? &? ? ' ? ! ! ? ? ? ?) ? % ? ? ! % ? % ? ? ' ? \$?? \$?? & ? % ? ? ? ! ? ?

) / ??? ? .

20-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

Nota: Para obter os desenhos das embalagens mais atuais, consulte a Especificação de embalagem do microchip, localizada em <http://www.microchip.com/packaging>



| Units | | MILLIMETERS | | |
|--------------------------|----|-------------|------|------|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | 0.65 BSC | | |
| Contact Pad Spacing | C | | 7.20 | |
| Contact Pad Width (X20) | X1 | | | 0.45 |
| Contact Pad Length (X20) | Y1 | | | 1.75 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

APÊNDICE A: HISTÓRICO DE REVISÃO

Revisão D (março de 2017)

A seguir está a lista de modificações:

1. Atualizado [Seção 1.8.2 "Redefinição de inicialização \(POR\) "](#) e adicionou novo [Figura 1-6](#) .

Revisão C (dezembro de 2015)

A seguir está a lista de modificações:

1. Windows adicionado 8, Windows 8.1 e Windows 10 para [Características e Seção 1.1 "Sistemas operacionais suportados"](#) .

Revisão B (março de 2011)

A seguir está a lista de modificações:

1. Adicionada nova seção [Seção 1.5.2](#) .
2. Todo atualizado [Seção 2.3 "Configuração e DLL de E / S "](#) .
3. Valores adicionados aos parâmetros TH01 e TH02 para o pacote VQFN de 20 derivações 5x5 no [Quadro 3-2](#) .

Revisão A (março de 2010)

Versão original deste documento.

MCP2200

SISTEMA DE IDENTIFICAÇÃO DO PRODUTO

Para solicitar ou obter informações, por exemplo, sobre preços ou entrega, entre em contato com o escritório de vendas local da Microchip.

| <u>PARTE NO.</u> | <u>IX</u> (1) | <u>X</u> | <u>/ XX</u> |
|--|---------------|-------------|-------------|
| Dispositivo | Tape and Reel | Temperatura | Pacote |
| | Opção | Alcance | |
| <div>Dispositivo: MCP2200: Conversor serial USB-UART MCP2200T: Conversor serial USB-UART (Tape and Reel)</div> <div>Opção de fita e rolo: Em branco = Embalagem padrão (tubo ou bandeja) T = Fita e rolo (1)</div> <div>Faixa de temperatura: Eu = - 40 • C a +85 • C (Industrial)</div> <div>Pacote: MQ = Quad de plástico plano, sem pacote de chumbo Corpo 5x5x1 mm (VQFN), SO de 20 derivações = contorno pequeno de plástico - largo, corpo de 7,50 mm (SO), 20-Lead SS = Contorno pequeno de contração plástica - 5,30 mm Corpo (SS) 20-Lead</div> | | | |

Exemplos:

a) MCP2200- I / MQ: Temperatura industrial, pacote 20LD VQFN.

b) MCP2200T- I / MQ: fita e rolo, Temperatura industrial, pacote 20LD VQFN.

c) MCP2200- I / SO: Temperatura industrial, pacote 20LD SOIC.

d) MCP2200T- I / SO: fita e rolo, Temperatura industrial, pacote 20LD SOIC.

e) MCP2200- I / SS: Temperatura industrial, pacote de 20LD SSOP.

f) MCP2200T- I / SS: Fita e rolo, Temperatura industrial, pacote de 20LD SSOP.

Nota 1: O identificador de fita e bobina aparece apenas na descrição do número de peça do catálogo. Esse identificador é usado para fins de pedido e não é impresso na embalagem do dispositivo. Verifique com o escritório de vendas do Microchip a disponibilidade do pacote com a opção Tape and Reel.

Observe os seguintes detalhes do recurso de proteção de código nos dispositivos Microchip:

- Os produtos de microchip atendem às especificações contidas em sua ficha técnica de microchip.
- A Microchip acredita que sua família de produtos é uma das famílias mais seguras do gênero atualmente no mercado, quando usada da maneira pretendida e em condições normais.
- Existem métodos desonestos e possivelmente ilegais usados para violar o recurso de proteção de código. Todos esses métodos, que sabemos, exigem o uso dos produtos Microchip de maneira fora das especificações operacionais contidas nas Folhas de Dados da Microchip. Provavelmente, a pessoa que está fazendo isso está envolvida em roubo de propriedade intelectual.
- A Microchip está disposta a trabalhar com o cliente preocupado com a integridade do seu código.
- Nem a Microchip, nem qualquer outro fabricante de semicondutores podem garantir a segurança de seu código. A proteção de código não significa que estamos garantindo o produto como "inquebrável".

A proteção de código está em constante evolução. Nós da Microchip estamos comprometidos em melhorar continuamente os recursos de proteção de código de nossos produtos. Tentativas de quebrar o recurso de proteção de código da Microchip podem ser uma violação da Lei de Direitos Autorais do Milênio Digital. Se tais atos permitirem acesso não autorizado ao seu software ou outro trabalho protegido por direitos autorais, você poderá ter o direito de processar por alívio nos termos dessa lei.

As informações contidas nesta publicação sobre aplicativos de dispositivos e similares são fornecidas apenas para sua conveniência e podem ser substituídas por atualizações. É de sua responsabilidade garantir que seu aplicativo atenda às suas especificações. A MICROCHIP NÃO FAZ REPRESENTAÇÕES OU GARANTIAS DE QUALQUER TIPO, EXPRESSO OU IMPLÍCITO, ESCRITO OU ORAL, ESTATUTÁRIO OU DE OUTRA FORMA, RELACIONADO COM O

EM FORMAÇÃO,
INCLUINDO, MAS NÃO SE LIMITANDO À SUA CONDIÇÃO, QUALIDADE, DESEMPENHO, COMERCIALIZAÇÃO OU ADEQUAÇÃO PARA FINS. A Microchip se isenta de qualquer responsabilidade decorrente dessas informações e de seu uso. O uso de dispositivos Microchip em aplicativos de suporte à vida e / ou segurança é de inteira responsabilidade do comprador, e o comprador concorda em defender, indenizar e isentar o Microchip de qualquer e todos os danos, reclamações, ações ou despesas resultantes de tal uso. Nenhuma licença é transmitida,

implícita ou não, sob qualquer Microchip
direitos de propriedade intelectual, salvo indicação em contrário.

A Microchip recebeu a certificação ISO / TS-16949: 2009 por sua sede mundial, instalações de design e fabricação de wafer em Chandler e Tempe, Arizona; Gresham, Oregon e centros de design na Califórnia e na Índia. Os processos e procedimentos do sistema de qualidade da Companhia são para seu PIC® MCUs e dsPIC® DSCs, K EE L OQ® dispositivos de salto de código, EEPROMs seriais, microperiféricos, memória não volátil e produtos analógicos. Além disso, o sistema de qualidade da Microchip para o design e fabricação de sistemas de desenvolvimento é certificado pela ISO 9001: 2000.

SISTEMA DE GESTÃO DA QUALIDADE

CERTIFICADO PELA DNV

== ISO / TS 16949 ==

Marcas comerciais

O nome e o logotipo do Microchip, o logotipo do Microchip, AnyRate, AVR, logotipo AVR, AVR Freaks, BeaconThings, BitCloud, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, K EE L OQ, K EE L OQ logotipo, Klear, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, RightTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI / O e XMEGA são marcas registradas da Microchip Technology Incorporated nos EUA e em outros países.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge e Quiet-Wire são marcas registradas da Microchip Technology Incorporated nos EUA Adjacent Key Supression, AKS, Analog-for-the- Era digital, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, logotipo chipKIT, CodeGuard, CryptoAuthentication, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Correspondência dinâmica média, DAM, ECAN, EtherGREEN, Programação serial em circuito, ICSP, Inter -Chip Connectivity, JitterBlocker, KlearNet, logotipo KlearNet, Mindi, MiWi, motorBench, MPASM, MPF, logotipo certificado MPLAB, MPLIB, MPLINK, MultiTRAK, NetDetach, geração de código onisciente, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, Qatrix , Logotipo RightTouch, REAL ICE, bloqueador de ondulações, SAM-ICE, E / S quádrupla serial, SMART-IS,SQL, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA e ZENA são marcas comerciais da Microchip Technology Incorporated no

EUA e outros países.

SQTP é uma marca de serviço da Microchip Technology Incorporated nos EUA

Silicon Storage Technology é uma marca registrada da Microchip Technology Inc. em outros países. GestIC é uma marca registrada da Microchip Technology Germany II GmbH & Co. KG, uma subsidiária da Microchip Technology Inc., em outros países.

Todas as outras marcas comerciais mencionadas neste documento pertencem a suas respectivas empresas.

© 2011-2017, Microchip Technology Incorporated, todos os direitos reservados.
ISBN: 978-1-5224-1447-6

Vendas e serviços mundiais

AMÉRICAS

Escritório corporativo
2355 West Chandler Blvd. Chandler,
AZ 85224-6199 Tel: 480-792-7200
Fax: 480-792-7277 Suporte técnico:

<http://www.microchip.com/support>

Endereço Web:
www.microchip.com

Atlanta
Duluth, GA Tel:
678-957-9614 Fax:
678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA Tel:
774-760-0087 Fax:
774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071 Fax:
630-285-0075

Dallas
Endereço: RUA JOAQUIM DE
ALMEIDA, S / N Bairro:
CENTRO

Detroit
Novi, MI
Tel: 248-848-4000

Houston, Texas
Tel: 281-894-5983

Indianapolis
Noblesville, IN Tel:
317-773-8323 Fax:
317-773-5453 Tel:
317-536-2380

Los Angeles
Mission Viejo, CA Tel:
949-462-9523 Fax:
949-462-9608 Tel:
951-273-7800

Raleigh, NC
Tel: 919-844-7510

Nova Iorque, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110 Tel:
408-436-4270

Canadá - Toronto
Tel: 905-695-1980 Fax:
905-695-2078

ÁSIA-PACÍFICO

Escritório na Ásia-Pacífico
Suites 3707-14, 37th Floor Tower 6,
The Gateway Harbor City, Kowloon

Hong Kong
Tel: 852-2943-5100 Fax:
852-2401-3431

Austrália - Sydney
Tel: 61-2-9868-6733 Fax:
61-2-9868-6755

China - Pequim
Tel: 86-10-8569-7000 Fax:
86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511 Fax:
86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588 Fax:
86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115 Fax:
86-571-8792-8116

China - RAE de Hong Kong
Tel: 852-2943-5100 Fax:
852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460 Fax:
86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355 Fax:
86-532-8502-7205

China - Xangai
Tel: 86-21-3326-8000 Fax:
86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829 Fax:
86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200 Fax:
86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300 Fax:
86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252 Fax:
86-29-8833-7256

ÁSIA-PACÍFICO

China - Xiamen
Tel: 86-592-2388138 Fax:
86-592-2388130

China - Zhuhai
Tel: 86-756-3210040 Fax:
86-756-3210049

Índia - Bangalore
Tel: 91-80-3090-4444 Fax:
91-80-3090-4123

Índia - Nova Deli
Tel: 91-11-4160-8631 Fax:
91-11-4160-8632

Índia - Pune
Tel: 91-20-3019-1500

Japão - Osaka
Telefone: 81-6-6152-7160
Fax: 81-6-6152-9310

Japão - Tóquio
Telefone: 81-3-6880-3770
Fax: 81-3-6880-3771

Coreia - Daegu
Tel: 82-53-744-4301 Fax:
82-53-744-4302

Coreia - Seul
Tel: 82-2-554-7200 Fax:
82-2-558-5932 ou 82-2-558-5934

Malásia - Kuala Lumpur
Tel: 60-3-6201-9857 Fax:
60-3-6201-9859

Malásia - Penang
Tel: 60-4-227-8870 Fax:
60-4-227-4068

Filipinas - Manila
Tel: 63-2-634-9065 Fax:
63-2-634-9069

Cingapura
Tel: 65-6334-8870 Fax:
65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366 Fax:
886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipé
Tel: 886-2-2508-8600 Fax:
886-2-2508-0102

Tailândia - Bangkok
Tel: 66-2-694-1351 Fax:
66-2-694-1350

EUROPA

Áustria - Wels
Tel: 43-7242-2244-39 Fax:
43-7242-2244-393

Dinamarca - Copenhagen
Tel: 45-4450-2828 Fax:
45-4485-2829

Finlândia - Espoo
Tel: 358-9-4520-820

França - Paris
Tel: 33-1-69-53-63-20 Fax:
33-1-69-30-90-79

França - Saint Cloud
Tel: 33-1-30-60-70-00

Alemanha - Garching
Tel: 49-8931-9700

Alemanha - Haan
Tel: 49-2129-3766400

Alemanha - Heilbronn
Tel: 49-7131-67-3636

Alemanha - Karlsruhe
Tel: 49-721-625370

Alemanha - Munique
Tel: 49-89-627-144-0 Fax:
49-89-627-144-44

Alemanha - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Itália - Milão
Tel: 39-0331-742611 Fax:
39-0331-466781

Itália - Pádua
Tel: 39-049-7625286

Países Baixos - Druen
Tel: 31-416-690399 Fax:
31-416-690340

Noruega - Trondheim
Tel: 47-7289-7561

Polônia - Varsóvia
Tel: 48-22-3325737

Romênia - Bucareste
Tel: 40-21-407-87-50

Espanha - Madrid
Tel: 34-91-708-08-90 Fax:
34-91-708-08-91

Suécia - Gotemburgo
Tel: 46-31-704-60-40

Suécia - Estocolmo
Tel: 46-8-5090-4654

Reino Unido - Wokingham
Tel: 44-118-921-5800 Fax:
44-118-921-5820