



NOVO 204A

Folha de Dados de CriptoAutenticação de Microchip ATSHA204A

Características

- Elemento criptográfico com armazenamento de chave baseado em hardware protegido • Operações de host e cliente de dispositivo de autenticação simétrica segura • Algoritmo hash SHA-256 superior com código de autenticação de mensagem (MAC) e baseado em hash Opções de código de autenticação de mensagem (HMAC) O • melhor da categoria, comprimento de chave de 256 bits; Armazenamento para até 16 chaves • Número de série exclusivo de 72 bits • garantido, gerador de números aleatórios (RNG) interno de alta qualidade • EEPROM de 4,5 kb para chaves e dados • Bits OTP (programáveis uma vez) de 512 bits para informações fixas • Múltiplas opções de E/S
 - Interface de fio único de alta velocidade compatível com UART – Interface I2C de 1 MHz • Faixa de tensão de alimentação de 2,0 V a 5,5 V • Faixa de tensão de comunicação de 1,8 V a 5,5 V • Corrente de suspensão <150 nA • Download seguro e Bota
 - Controle do Ecossistema
 - Segurança de mensagens
 - Anticlonagem •
- SOIC de 8 derivações, TSSOP de 8 derivações (2), SOT23 de 3 derivações, UDFN de 8 pads e pacotes CONTACT de 3 derivações

Formulários

- Download e Inicialização Seguros
- Controle do Ecossistema
- Anticlonagem •

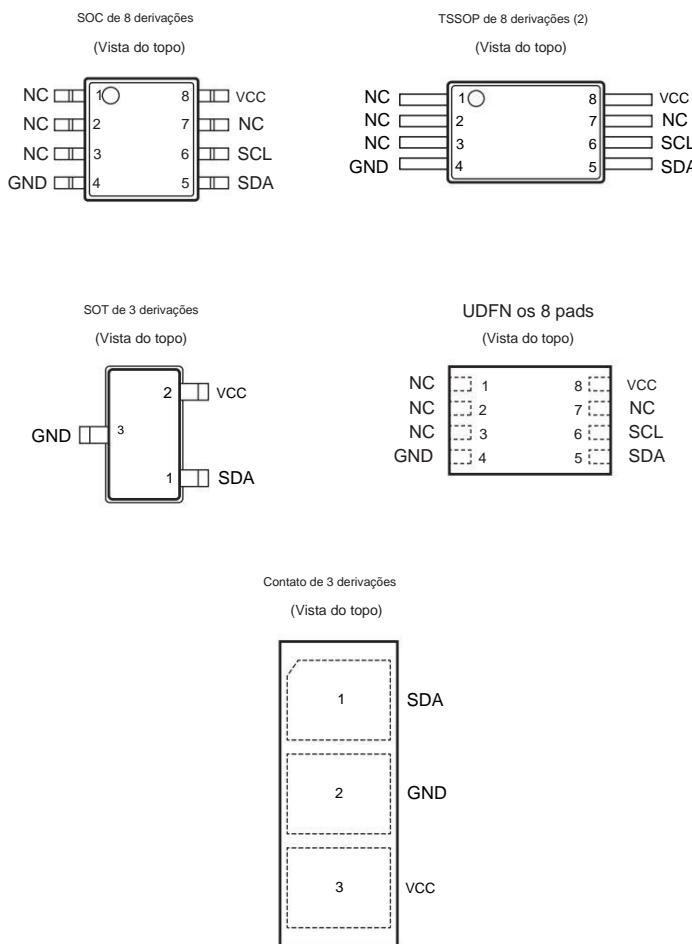
Segurança de mensagens

Tipos de pacote

Tabela 1. Configuração de pinos

Nome do PIN	Função
NC	Sem Conexão
GND	Chão
SDA	Dados Seriais
SCL	Entrada de Relógio Serial
VCC	Fonte de energia

Figura 1. Pinagem(1)



Observação:

1. Os desenhos não estão em escala.
2. Não recomendado para novos projetos.

Índice

Características.....	1
Formulários.....	1
Tipos de embalagem.....	2
1. Introdução.....	5
1.1. Formulários.....	5
do dispositivo.....	5
1.3. Operação criptográfica.....	6
2. Organização do dispositivo.....	8
Organização da EEPROM.....	8
2.2. RAM Estática (SRAM).....	17
3. Recursos de segurança.....	19
3.1. Segurança física.....	19
Gerador de Número Aleatório (RNG).....	19
4. Informações gerais de E/S.....	20
4.1. Ordenação de Bytes e Bits.....	20
5. Interface de fio único.....	22
5.1. Tokens de E/S.....	22
5.2. Sinalizadores de E/S.....	23
5.3. Sincronização.....	24
Compartilhando a interface.....	24
Exemplo de Transação.....	25
Configuração de fiação para interface de fio único.....	26
6. Interface I ² C.....	28
6.1. Condições de E/S.....	28
2C Transmissão para o Dispositivo ATSHA204A.....	30
Transmissão do Dispositivo ATSHA204A.....	32
6.4. Contador de endereços.....	32
6.5. I ² C Sincronização.....	33
Exemplo de Transação.....	34
7. Características elétricas.....	35
7.1. Classificações Máximas Absolutas.....	35
7.2. Confiabilidade.....	35
Parâmetros CA — Todas as Interfaces de E/S.....	35
7.4. Parâmetros CC — Todas as Interfaces de E/S.....	39
8. Comandos de Segurança.....	42

8.1. Blocos de E/S.....	42
8.2. Sequência de sono.....	43
inativa.....	43
falhas.....	43
Comando.....	44
9. Compatibilidade.....	66
10. Mecânica.....	67
10.1. Pinagem.....	67
11. Informações de Marcação da Embalagem.....	68
12. Desenhos de Embalagens.....	69
12.1. UDFN de 8 pads.....	69
derivações.....	72
12.3. TSSOP de 8 derivações.....	75
12.4. 3 Contato do Lead.....	77
12.5. SOT23 de 3 derivações.....	79
13. Referência e Notas de Aplicação.....	83
13.1. SHA-256.....	83
13.2. HMAC/SHA-256.....	83
13.3. Valores-chave	84
14. Histórico de revisões.....	88
O site da Microchip.....	89
Serviço de notificação de alteração do cliente.....	89
Suporte ao cliente.....	89
Sistema de Identificação do Produto.....	90
Recurso de Proteção de Código de Dispositivos Microchip.....	91
Notícia legal.....	91
Marcas Registradas.....	91
Sistema de Gestão da Qualidade Certificado pela DNV.....	92
Vendas e serviços em todo o mundo.....	93

1. Introdução

As seções a seguir apresentam os recursos e funções do dispositivo de elemento criptográfico Microchip ATSHA204A.

1.1

Aplicações O

ATSHA204A é um membro da família Microchip CryptoAuthentication™ de dispositivos de autenticação de hardware de alta segurança. Possui um conjunto de comandos flexível que permite o uso em vários aplicativos, incluindo os seguintes:

- **Antifalsificação** Valida se

um cliente removível, substituível ou consumível é autêntico. Exemplos de clientes podem ser tanques de tinta de impressora, cartões auxiliares eletrônicos, descartáveis médicos ou peças de reposição. O dispositivo também pode ser usado para validar (autenticar) um módulo de software/firmware ou elemento de armazenamento de memória. •

- **Proteção de firmware ou mídia** Valida o

código armazenado na memória flash no momento da inicialização para evitar modificações não autorizadas (isso também é conhecido como inicialização segura), criptografa arquivos de mídia baixados e criptografa exclusivamente imagens de código para serem usadas apenas em um

- **Troca de Chaves de Sessão**

Troca chaves de criptografia de fluxo com segurança e facilidade para uso por um mecanismo de criptografia/descriptografia no microprocessador do sistema para gerenciar um canal de comunicação confidencial, um download criptografado e itens semelhantes.

- **Armazenamento de dados com**

segurança Armazena chaves secretas para uso por aceleradores de criptografia em microprocessadores padrão.

Também pode ser usado para armazenar pequenas quantidades de dados necessários para configuração, calibração, valor ePurse, dados de consumo ou outros segredos. Proteção programável por meio de leituras e gravações criptografadas/autenticadas.

- **Verificando a senha do usuário**

Valida as senhas inseridas pelo usuário sem deixar que o valor esperado se torne conhecido, mapeando senhas simples para complexas e trocando valores de senha com segurança com sistemas remotos.

1.2

Recursos do dispositivo

O dispositivo ATSHA204A inclui uma matriz de memória somente leitura programável apagável eletricamente (EEPROM) que pode ser usada para armazenamento de chaves, dados diversos de leitura/gravação, somente leitura, dados secretos, registro de consumo e configuração de segurança. O acesso às várias seções de memória pode ser restrito de várias maneiras e a configuração pode ser bloqueada para evitar alterações. Consulte a Seção [Organização da EEPROM](#) para obter detalhes.

O ATSHA204A apresenta uma ampla gama de mecanismos de defesa projetados especificamente para evitar ataques físicos ao próprio dispositivo ou ataques lógicos aos dados transmitidos entre o dispositivo e o sistema, consulte a Seção [Recursos de segurança](#) para obter mais detalhes. As restrições de hardware na forma como as chaves são usadas ou geradas fornecem uma defesa adicional contra certos estilos de ataque.

O acesso ao dispositivo é feito através de uma interface padrão I2C com velocidades de até 1 Mb/s. consulte a Seção [I2C Interface](#) para obter detalhes. É compatível com as especificações de interface I2C . O dispositivo também oferece suporte a uma interface de fio único (SWI) que pode reduzir o número de GPIOs necessários no processador do sistema e/ou reduzir o número de pinos nos conectores. Consulte a seção [Interface de fio único](#) para obter mais detalhes.

Usando a interface de fio único, vários dispositivos ATSHA204A podem compartilhar o mesmo barramento, o que economiza o uso do GPIO do processador em sistemas com vários clientes, como tanques de tinta de cores diferentes ou várias peças sobressalentes, como exemplos. Consulte a seção [Compartilhando a interface](#) e a seção [Comando de pausa](#) para obter detalhes sobre como isso é implementado.

Cada ATSHA204A vem com um número de série único garantido de 9 bytes (72 bits). Usando os protocolos criptográficos suportados pelo dispositivo, um sistema host ou servidor remoto pode provar que o número de série é autêntico e não uma cópia. Os números de série geralmente são armazenados em uma EEPROM serial padrão, que pode ser facilmente copiada sem que o host saiba se o número de série é autêntico ou se é um clone. O número de série completo deve ser utilizado para garantir a exclusividade.

O ATSHA204A pode gerar números aleatórios de alta qualidade e empregá-los para qualquer finalidade, inclusive como parte dos protocolos de criptografia deste dispositivo. Como cada número aleatório de 32 bytes (256 bits) não depende de números anteriores gerados neste ou em qualquer outro dispositivo, sua inclusão no cálculo do protocolo garante que os ataques de repetição (por exemplo, retransmitir uma transação anterior bem-sucedida) sempre falhem. Veja a Seção [Gerador de Número Aleatório \(RNG\)](#) e a Seção [Comando Aleatório](#).

A integração do sistema é facilitada por uma ampla faixa de tensão de alimentação (de 2,0 V a 5,5 V) e uma corrente de suspensão ultrabaixa (de <150 nA). Parâmetros CC completos são encontrados na Seção [Características Elétricas](#), que descreve várias opções de pacote, incluindo um pequeno pacote UDFN com tamanho de apenas 2,0 mm x 3,0 mm. Consulte a Seção [Desenhos de embalagens](#) para obter mais detalhes e códigos de pedido.

Consulte a Seção [Compatibilidade](#) para obter informações sobre compatibilidade com o Microchip ATSHA204.

1.3

Operação criptográfica O ATSHA204A

oferece suporte a um protocolo padrão de desafio-resposta para simplificar a programação. Na sua instalação mais básica, o sistema Host envia um desafio (por exemplo, um número) para o dispositivo no Cliente, que combina esse desafio com uma chave secreta usando o comando Message Authentication Code (MAC) do sistema, conforme descrito em Section [MAC](#)

Command e envia essa resposta de volta ao sistema. O dispositivo usa um algoritmo de hash criptográfico para fazer essa combinação (que também é conhecida como resumo). O uso de um algoritmo de hash evita que um observador no barramento deduza o valor da chave secreta, enquanto permite que o destinatário verifique se a resposta está correta realizando o mesmo cálculo combinando o desafio com o segredo para criar um resumo usando um armazenado cópia do segredo.

Esta operação básica pode ser expandida de várias maneiras devido ao conjunto de comandos flexível do ATSHA204A.

Ao usar o comando GenDig (seção [GenDig Command](#)), os valores em outros slots podem ser incluídos no resumo da resposta, o que fornece uma maneira eficaz de provar que um dado lido realmente veio do dispositivo, em vez de ter sido inserido por um homem -in-the-middle atacante. Esse mesmo comando pode ser usado para combinar duas chaves com o desafio, o que é útil quando há várias camadas de autenticação a serem executadas.

O comando DeriveKey (Seção [Comando DeriveKey](#)) implementa um esquema de rolagem de teclas.

Dependendo do parâmetro do modo de comando, a operação resultante pode ser semelhante àquela implementada em um controle remoto de abertura de porta de garagem, por exemplo. Cada vez que a chave é usada, o valor atual da chave é criptograficamente combinado com um valor específico daquele sistema e esse resultado então forma a chave para a próxima operação criptográfica. Mesmo que um invasor obtenha o valor de uma chave, essa chave desaparece para sempre no próximo uso.

DeriveKey também pode ser usado para gerar novas chaves aleatórias que podem ser válidas apenas para um ID de host específico, para um período de tempo específico ou para alguma outra condição restrita. Cada chave gerada é diferente de

qualquer outra chave já gerada em qualquer dispositivo. Ao “ativar” um par Host-Cliente no campo dessa maneira, um clone de um único Cliente não pode funcionar em nenhum outro Host.

Numa configuração Host-Client em que o Host (por exemplo um telemóvel) necessita de verificar um Cliente (por exemplo uma bateria OEM), existe a necessidade de guardar o segredo no Host de forma a validar a resposta do Cliente. O comando CheckMac (Comando [CheckMac da Seção](#)) permite que o dispositivo Host armazene com segurança o segredo do Cliente e oculte o valor correto da resposta dos pinos, retornando apenas uma resposta sim/não ao sistema.

Quando uma senha inserida pelo usuário é necessária, o comando CheckMac também fornece uma maneira de verificar a senha sem expô-la no barramento de comunicação e mapear a senha para um valor armazenado que pode ter uma entropia muito maior. Consulte a seção [Verificação de senha](#) para obter detalhes.

Finalmente, a combinação de hash (por exemplo, resumo) de um desafio e chave secreta pode ser mantida no dispositivo e XORed com o conteúdo de um slot para implementar uma leitura criptografada (Section Read Command), ou pode ser XORed com dados de [entrada](#) criptografados para implementar uma gravação criptografada (Seção [Write Command](#)).

Cada uma dessas operações pode ser protegida contra ataques de repetição incluindo um nonce aleatório (Section [Nonce Command](#)) no cálculo.

Todas as funções de segurança são implementadas usando o algoritmo de hash seguro SHA-256 padrão do setor, que faz parte do conjunto mais recente de algoritmos criptográficos de alta segurança recomendados por várias agências governamentais e especialistas em criptografia. A seção [SHA-256](#) inclui uma referência aos detalhes do algoritmo. Se desejado, o algoritmo SHA-256 também pode ser incluído em uma sequência HMAC (consulte a seção [Comando HMAC](#)). O ATSHA204A emprega chaves secretas de tamanho normal de 256 bits para evitar qualquer tipo de ataque exaustivo.

2.

Organização do dispositivo

O dispositivo contém os seguintes blocos de memória:

- EEPROM
- SRAM

2.1 Organização da EEPROM A

EEPROM contém um total de 664 bytes (5312 bits) e está dividida nas seguintes zonas:

Tabela 2-1. Zonas ATSHA204A

Zona	Descrição	Nomenclatura
Dados	Zona de 512 bytes (4,0 kb) dividida em 16 slots de memória de uso geral somente leitura ou leitura/gravação de 32 bytes (256 bits) cada, que podem ser usados para armazenar chaves, dados de calibração, número do modelo ou outras informações, geralmente referem-se ao item ao qual o dispositivo ATSHA204A está conectado. A política de acesso de cada slot de dados é determinada pelos valores programados nos valores de configuração correspondentes. No entanto, as políticas tornam-se efetivas apenas ao definir o byte LockValue.	Slot <YY> = O inteiro conteúdos armazenados no Slot YY da zona de Dados.
Configuração	Zona de 88 bytes (704 bits) EEPROM que contém o número de série e outras informações de identificação, além de acessar as informações de permissão para cada slot da memória de dados. Os valores programados na zona de configuração determinam a política de acesso de como cada slot de dados responde. A zona de configuração pode ser modificada até que seja bloqueada (LockConfig definido como !=0x55). Para habilitar as políticas de acesso, o byte LockValue deve ser definido. (Veja a seção acima)	SN<a:b> = Um intervalo de bytes dentro de um campo da zona de configuração.
Um tempo Programável (OTP)	Zona de 64 bytes (512 bits) de bits OTP. Antes de bloquear a OTPzone, os bits podem ser escritos livremente usando o comando Write padrão. A zona OTP pode ser usada para armazenar dados somente leitura ou informações de registro de consumo do tipo fusível unidirecional.	OTP<bb> = Um byte dentro da zona OTP, enquanto OTP<aa:bb> indica um intervalo de bytes.

Os termos discutidos neste documento têm os seguintes significados:

Tabela 2-2. Termos do Documento

Prazo	Significado
Bloquear	Uma única área de 256 bits (32 bytes) de uma zona de memória específica. A documentação da indústria SHA-256 usa o termo "bloco" para indicar uma seção de 512 bits da entrada da mensagem. Além disso, a seção de E/S deste documento usa o termo "bloco" para indicar um elemento agregado de comprimento variável transferido entre o sistema e o dispositivo.
slot	Para a zona de dados, os termos "Block" e "Slot" podem ser usados de forma intercambiável. Para a zona OTP e Config existem vários blocos de 32 Bytes cada.
param	Indica um bit de parâmetro ou campo de byte.
SRAM	Contém buffers de entrada e saída, bem como locais de armazenamento de estado. Consulte a seção RAM estática (SRAM)

Na remessa da Microchip, a EEPROM contém dados de teste de fábrica que podem ser usados para teste de placa de valor fixo. Esses dados devem ser substituídos pelo conteúdo desejado antes de bloquear a configuração e/ou as seções de dados do dispositivo. Veja o [site da Microchip](#) para o documento que contém os valores de transporte específicos.

2.1.1 Zona de Dados EEPROM

A zona de dados tem 512 bytes (4 kb), faz parte da matriz EEPROM e pode ser usada para armazenamento seguro propósitos.

Antes de bloquear a seção de configuração usando Lock(Config), a zona de dados fica inacessível e não pode ser lida nem gravada.

Após o bloqueio da configuração, toda a zona de dados pode ser gravada usando o comando Write. Se desejado, os dados a serem gravados podem ser criptografados.

Na tabela a seguir, "Byte Address" é o endereço de byte dentro da zona de dados para o primeiro byte no respectivo slot.

Porque todas as leituras e gravações com o ATSHA204A são executadas com base em uma palavra (4 bytes ou 32 bytes) e o endereço da palavra na tabela abaixo deve ser usado para o parâmetro de endereço passado para os comandos de leitura e gravação.

Tabela 2-3. Slots de zona de dados

Slot	Byte Address (Hex)	Word Address (Hex)	Slot	Byte Address (Hex)	Word Address (Hex)
0	0x0000	0x0000	8	0x0100	0x0040
1	0x0020	0x0008	9	0x0120	0x0048
2	0x0040	0x0010	10	0x0140	0x0050
3	0x0060	0x0018	11	0x0160	0x0058
4	0x0080	0x0020	12	0x0180	0x0060
5	0x00A0	0x0028	13	0x01A0	0x0068
6	0x00C0	0x0030	14	0x01C0	0x0070
7	0x00E0	0x0038	15	0x01E0	0x0078

2.1.2

Zona de configuração

Os 88 bytes (704 bits) na zona de configuração contêm dados de identificação de fabricação, configuração geral do dispositivo e do sistema e valores de controle de restrição de acesso para os slots dentro da zona de dados.

Os valores desses bytes sempre podem ser obtidos usando o comando Read. Os bytes desta zona são organizados conforme mostrado na tabela a seguir.

Tabela 2-4. Zona de Configuração

Palavra	Byte 0	Byte 1	Byte 2	byte 3	Padrão	Gravar	Acessar	Ler	Acesso
0x00		SN<0:3>			01 23 xx xx	Nunca	Sempre		
0x01		Núm Rev.			xx xx xx xx	Nunca	Sempre		
0x02		SN<4:7>			xx xx xx xx	Nunca	Sempre		
0x03	SN<8>	Reservado	I2C_Enable	Reservado	EE 55 xx 00	Nunca	Sempre		
0x04	I2C_Address	CheckMacConfig	Modo OTP		Seletor Modo C8 00 55 00	Se a configuração estiver desbloqueada	Sempre		

NOVO 204A**Organização do dispositivo**

Palavra	Byte 0	Byte 1	Byte 2	byte 3	Padrão	Gravar	Acessar	Ler	Acesso
0x05	SlotConfig 0		SlotConfig 1		8F 80 80 A1	Se a configuração estiver desbloqueada			Sempre
0x06	SlotConfig 2		SlotConfig 3		82 E0 A3 60	Se a configuração estiver desbloqueada			Sempre
0x07	SlotConfig 4		SlotConfig 5		94 40 A0 85	Se a configuração estiver desbloqueada			Sempre
0x08	SlotConfig 6		SlotConfig 7		86 40 87 07	Se a configuração estiver desbloqueada			Sempre
0x09	SlotConfig 8		SlotConfig 9		0F 00 89 F2	Se a configuração estiver desbloqueada			Sempre
0x0A	SlotConfig 10		SlotConfig 11		8A 7A 0B 8B Se o Config estiver desbloqueado				Sempre
0x0B	SlotConfig 12		SlotConfig 13		0C 4C DD 4D Se o Config estiver desbloqueado				Sempre
0x0C	SlotConfig 14		SlotConfig 15		C2 42 AF 8F Se Config estiver desbloqueado				Sempre
0x0D UseFlag 0	UpdateCount 0 UseFlag 1		UpdateCount 1 FF 00 FF 00			Se a configuração estiver desbloqueada			Sempre
0x0E UseFlag 2	UpdateCount 2 UseFlag 3		UpdateCount 3 FF 00 FF 00			Se a configuração estiver desbloqueada			Sempre
0x0F UseFlag 4	UpdateCount 4 UseFlag 5		UpdateCount 5 FF 00 FF 00			Se a configuração estiver desbloqueada			Sempre
0x10 UseFlag 6	UpdateCount 6 UseFlag 7		UpdateCount 7 FF 00 FF 00			Se a configuração estiver desbloqueada			Sempre
0x11 LastKeyUse 0 LastKeyUse 1		LastKeyUse 2 LastKeyUse 3 FF FF FF FF Se Config estiver desbloqueado							Sempre
0x12 LastKeyUse 4 LastKeyUse 5 LastKeyUse 6 LastKeyUse 7 FF FF FF FF Se Config estiver desbloqueado									Sempre
0x13 LastKeyUse 8 LastKeyUse 9 LastKeyUse 10 LastKeyUse 11 FF FF FF FF Se Config estiver desbloqueado									Sempre
0x14 LastKeyUse 12 LastKeyUse 13 LastKeyUse 14 LastKeyUse 15 FF FF FF FF Se Config estiver desbloqueado									Sempre
0x15 Usuário Extra	Seletor	LockValue1	LockConfig	00 00 55 55		Através UpdateExtra Comando Apenas			Sempre

Observação:

1. LockValue era anteriormente conhecido como LockData.

2.1.2.1 I2C_Enable**Parte 7-1:**

Ignorado e definido pelo Microchip.

Bit 0:	0 = Modo de interface de fio único. 1= Modo de interface I2C .
---------------	---

2.1.2.2 I2C_Endereço I2C**Modo I2C_Enable<0> = 1**

Bits 7 – 1:	I 2C endereço do dispositivo
Parte 3:	Ativar TTL (bit de dupla finalidade) Parte do endereço I2C e define o nível de limite.
	0= O nível de entrada usa uma referência fixa. 1 = O nível de entrada usa o VCC como referência.
Bit 0:	Ignorado.

Modo de fio único I2C_Enable<0> = 0

Partes 7–4:	Ignorado.
Parte 3:	Ativar TTL 0= O nível de entrada usa uma referência fixa. 1 = O nível de entrada usa o VCC como referência.
Bits 2–0:	Ignorado.

2.1.2.3 CheckMacConfig Este byte

se aplica somente aos comandos CheckMac, Read e Write:

- **Leitura e gravação:** CheckMacConfig<0> controla os slots 0 e 1, CheckMacConfig<1> controla os slots 2 e 3 e assim por diante. Qualquer comando de leitura ou gravação criptografado falhará se o valor em TempKey.SourceFlag não corresponder ao bit correspondente neste byte. Este byte é ignorado para leituras e gravações de texto não criptografado.
- **CheckMac:** CheckMacConfig<0> controla o slot 1, CheckMacConfig<1> controla o slot 3 e assim por diante. A função de cópia só pode ser habilitada se o valor CheckMacSource correspondente ao slot de destino corresponder ao valor do bit 2 do modo do comando CheckMac. O comando falhará se o bit 2 do modo não corresponder a TempKey.SourceFlag, portanto, isso equivale a exigir que o bit correspondente neste byte corresponda a TempKey.SourceFlag.

2.1.2.4 Modo OTP

0xAA (modo somente leitura) = Quando a zona OTP está bloqueada, as gravações são desabilitadas e as leituras de todas as palavras são permitidas.

0x55 (modo de consumo) = Grava na zona OTP quando a zona OTP está bloqueada faz com que os bits transitem apenas de um para zero. Leituras de todas as palavras são permitidas.

0x00 (modo legado) = Quando a zona OTP está bloqueada, as gravações são desativadas, as leituras das palavras 0 e 1 e as leituras de 32 bytes são desativadas.

Todos os outros modos são reservados.

2.1.2.5 Modo Seletor

Se for 0x00, o seletor será atualizado com UpdateExtra.

Todos os outros valores só podem permitir que o Seletor seja atualizado se seu valor for zero.

2.1.2.6 Configuração do Slot

Consulte Tabela [SlotConfig Bits \(por slot\)](#).

2.1.2.7 UseFlag

Para usos com "slots de uso limitado". A quantidade de bits "1" representa o número de vezes que os slots 0 a 7 podem ser utilizados antes de serem desabilitados.

2.1.2.8 Contagem de Atualizações

Indica quantas vezes os slots de 0 a 7 foram atualizados com DeriveKey.

2.1.2.9 LastKeyUse

Usado para controlar o uso limitado do Slot 15. Cada bit "1" representa um uso restante do Slot 15. Aplica-se somente se SlotConfig<5> LimitedUse estiver definido.

2.1.2.10 UserExtra

Para uso geral do sistema, pode ser modificado através do comando UpdateExtra.

2.1.2.11 Seletor

Seleciona qual dispositivo permanece no modo ativo após a execução do comando Pause.

2.1.2.12 Valor de Bloqueio

Controla as zonas de dados e OTP são desbloqueadas e podem ser escritas livremente, mas não lidas.

0x55 = As zonas de dados e OTP estão desbloqueadas e têm acesso de gravação.

0x00 = As zonas Dados e OTP estão bloqueadas e assumem as políticas de acesso definidas na zona de configuração. Os slots na zona de dados só podem ser modificados com base nos campos WriteConfig correspondentes. A zona OTP só pode ser modificada com base no modo OTP.

2.1.2.13 Acesso à

zona de Configuração LockConfig .

0x55 = A zona de configuração tem acesso de gravação (desbloqueado).

0x00 = A zona de configuração não tem acesso de gravação (bloqueado).

2.1.2.14 SlotConfig (Bytes 20 – 51)

Os 16 elementos SlotConfig configuram as proteções de acesso para cada um dos 16 slots dentro do ATSHA204A.

Cada elemento de configuração consiste em 16 bits, que controlam o uso e acesso para aquele slot ou chave em particular. O campo SlotConfig é interpretado de acordo com a tabela abaixo quando a Zona de dados está bloqueada. Quando a zona de dados é desbloqueada, essas restrições não se aplicam e todos os slots podem ser gravados livremente e nenhum pode ser lido.

Tabelas 2-5. Bits o SlotConfig (por slot)

Nome do bit	Descrição
15-12 WriteConfig	Consulte a definição detalhada da função para uso.
11-8 WriteKey	Slot da chave a ser usada para validar gravações criptografadas.

Nome do bit	Descrição
7 é secreto	0 = O slot não é secreto e permite leitura clara, gravação limpa, sem verificação de MAC e nenhum comando Derivekey. 1 = O slot é secreto. Leituras e gravações, se permitidas, devem ser criptografadas.
6 EncryptRead	0 = Leituras claras são permitidas. 1 = Requer que o slot seja Segredo e leitura criptografada para acessar.
5 Uso Limitado(1)	0 = Sem limite no número de vezes que a chave pode ser usada. 1 = Limite o número de vezes que a chave pode ser usada com base no UseFlag (ou LastKeyUse) para o slot.
4 CheckOnly	0 = Este slot pode ser usado para todos os comandos criptográficos. 1 = Este slot só pode ser usado para CheckMac e GenDig seguidos por comandos CheckMac.
3-0 ReadKey	Slot da chave a ser usada para leituras criptografadas. Se 0x0, então este slot pode ser usado como o slot de origem para o comando CheckMac/Copy.

Observação:

1. O bit LimitedUse era anteriormente denominado SingleUse.

Tabela 2-6. Gravar Bits de Configuração — Comando Derivekey

Bit 15	Bit 14	Bit 13	Bit 12	Source Key(1)	Descrição
0	x	1	0	Alvo	O comando DeriveKey pode ser executado sem autorização do MAC (Roll).
1	x	1	0	Alvo	MAC de autorização necessário para o comando DeriveKey (Roll).
0	x	1	1	Bom	O comando DeriveKey pode ser executado sem autorização do MAC (Criar).
1	x	1	1	Bom	É necessário autorizar o MAC para o comando DeriveKey (Criar).
x	x	0	x	—	Os slots com esse valor no campo WriteConfig não podem ser usados como destino do comando DeriveKey.

Observação:

1. A chave de origem para o cálculo executado pelo comando DeriveKey pode ser a chave especificado diretamente em Param2 (o "Target") ou a chave em SlotConfig<Param2>. WriteKey (o "Pai").

Consulte a seção [Valores-chave](#) para obter mais detalhes.

Tabela 2-7. Gravar bits de configuração — Comando de gravação

Bit 15	Bit 14	Bit 13	Caminho	Nome	Descrição
0	0	0	Sempre		As gravações de texto não criptografado são sempre permitidas neste slot. Slots definidos como “always” nunca devem ser usados como armazenamento de chaves. 4 ou 32 bytes podem ser escritos neste slot.
x	0	1	Nunca		As gravações nunca são permitidas neste slot usando o comando Write Slots definidos como “nunca” ainda podem ser usados como armazenamento de chaves.
1	0	x	Nunca		As gravações nunca são permitidas neste slot usando o comando Write Slots definidos como “nunca” ainda podem ser usados como armazenamento de chaves.
x	1	x	criptografar		As gravações neste slot requerem um MAC computado corretamente e os dados de entrada devem ser criptografados pelo sistema com WriteKey usando o algoritmo de criptografia documentado na Seção de descrição do comando de gravação (8.5.18 Comando de gravação). Gravações de 4 bytes neste slot são proibidas.

O campo WriteConfig de 4 bits é interpretado pelo comando Write conforme mostrado na Tabela [Write Configuration Bits —Write Command](#), onde X significa não importa.

As tabelas se sobreponem. Por exemplo, um código de 0b0110 indica que um slot pode ser gravado em formato criptografado usando o comando Write e também pode ser o destino de um comando DeriveKey não autorizado com o destino como origem.

O bit IsSecret controla os circuitos internos necessários para a segurança adequada dos slots nos quais leituras e/ou gravações devem ser criptografadas ou totalmente proibidas. Também deve ser definido para todos os slots que serão usados como chaves, incluindo aqueles criados ou modificados com DeriveKey. Especificamente, para permitir a operação adequada do dispositivo, esse bit deve ser definido, a menos que WriteConfig seja “Always”. Acessos de 4 bytes são proibidos de/para slots nos quais este bit é definido.

Os slots usados para armazenar valores de chave sempre devem ter IsSecret definido como um e EncryptRead definido como zero (leituras proibidas) para segurança máxima. Para valores de chave fixa, WriteConfig deve ser definido como “Nunca”. Quando configurado dessa forma, não há como ler ou gravar a chave após o bloqueio da zona de dados. Só pode ser usado para operações de criptografia.

Algumas políticas de segurança exigem que os segredos sejam atualizados periodicamente. O ATSHA204A suporta esta capacidade da seguinte forma:

- WriteConfig para o slot específico deve ser definido como “Encrypt” e SlotConfig.WriteKey deve apontar para o mesmo slot definindo WriteKey para o ID do slot. Um comando de gravação padrão pode então ser usado para gravar um novo valor neste slot, desde que o MAC de autenticação seja calculado usando o valor de chave antigo (atual).

2.1.2.15 Valores de Memória Especial na Zona de Configuração (Bytes 0 – 12)

Várias informações fixas estão incluídas no ATSHA204A que nunca podem ser escritas em nenhuma circunstância e sempre podem ser lidas, independentemente do estado dos bits de bloqueio.

- **SerialNum**

Nove bytes (SN<0:8>) que juntos formam um valor único que nunca se repete para nenhum dispositivo da família CryptoAuthentication. O número de série é dividido em dois grupos:

1.1. **SN<0:1> e SN<8>**

Os valores desses bits são fixados no momento da fabricação na maioria das versões do ATSHA204A. Seu valor padrão é (0x01 0x23 0xEE). Esses 24 bits são sempre incluídos nos cálculos SHA-256 feitos pelo ATSHA204A.

1.2. SN<2:7>

Os valores destes bits são programados pela Microchip durante o processo de fabricação e são diferentes para cada matriz. Esses 6 bytes (48 bits) são incluídos opcionalmente em alguns cálculos SHA-256 feitos pelo ATSHA204A

- Núm Rev.

Quatro bytes de informação que são usados pela Microchip para fornecer informações de revisão de fabricação. Esses bytes podem ser lidos livremente como RevNum<0:3>, mas nunca devem ser usados pelo software do sistema, pois podem sofrer alterações devido a uma revisão do silício.

2.1.3 Zona programável uma vez (OTP)

A zona OTP de

64 bytes (512 bits) faz parte da matriz EEPROM e pode ser usada para armazenamento somente leitura.

Antes de bloquear a seção de configuração usando Lock(LockConfig), a zona OTP fica inacessível e não pode ser lida nem gravada. Após o bloqueio da configuração, mas antes do bloqueio da zona OTP usando Lock(LockValue), toda a zona OTP pode ser gravada usando o comando Write. Se desejado, os dados a serem gravados podem ser criptografados. Quando desbloqueado, a zona OTP não pode ser lida.

Depois que a zona OTP é bloqueada, o byte do modo OTP na zona de configuração controla as permissões dessa zona, da seguinte forma:

- Modo somente leitura

Os dados não podem ser modificados e seriam usados para armazenar números fixos de modelos, informações de calibração, histórico de fabricação e/ou outros dados que nunca devem ser alterados. O comando Write sempre retorna um erro e deixa a memória inalterada. Todos os 64 bytes na seção OTP estão sempre disponíveis para leitura usando leituras de 4 ou 32 bytes.

- Modo de consumo

Os bits funcionam como fusíveis unidirecionais e podem ser usados para rastrear o consumo ou uso do item ao qual o ATSHA204A está conectado. Por exemplo, em uma bateria, eles podem ser usados para rastrear ciclos de carga ou tempo de uso; em um cartucho de tinta de impressora, eles podem rastrear a quantidade de material consumido; em um dispositivo médico, eles podem rastrear o número de usos permitidos para um item de uso limitado. Nesse modo, o comando Write só pode fazer com que os bits façam a transição de um para zero. Logicamente, isso significa que o valor dos dados na lista de parâmetros de entrada é AND'ed com o valor atual na(s) palavra(s) e o resultado gravado de volta na memória. Por exemplo, escrever um valor de 0xFF resulta em nenhuma alteração no byte e escrever um valor de 0x00 faz com que o byte na memória vá para zero, independentemente do valor anterior. Uma vez que um bit tenha transitado para zero, ele nunca poderá voltar para um. • Modo legado A operação da

zona OTP é consistente

com a matriz de fusíveis no Microchip (anteriormente Atmel)

ATSA102S. As leituras das palavras zero e um são sempre proibidas, enquanto as leituras das 14 palavras restantes são sempre permitidas. Somente leituras de 4 bytes (32 bits) são permitidas e qualquer tentativa de executar uma leitura de 32 bytes (256 bits) resulta em um código de retorno de erro. Todas as operações de gravação na zona OTP são proibidas. Veja a Seção 9. Compatibilidade para mais detalhes de compatibilidade do Microchip ATSA102S.

Todos os bits de zona OTP têm um valor de um no envio da fábrica da Microchip.

Tabela 2-8. zona OTP

Palavra (HEX)	Endereço (HEX)	Padrão
0x00	0x00	0xFFFFFFFF
0x01	0x04	0xFFFFFFFF
0x02	0x08	0xFFFFFFFF

Palavra (HEX)	Endereço (HEX)	Padrão
0x03	0x0C	0xFFFFFFFF
0x04	0x10	0xFFFFFFFF
0x05	0x14	0xFFFFFFFF
0x06	0x18	0xFFFFFFFF
0x07	0x1C	0xFFFFFFFF
0x08	0x20	0xFFFFFFFF
0x09	0x24	0xFFFFFFFF
0x0A	0x28	0xFFFFFFFF
0x0B	0x2C	0xFFFFFFFF
0x0C	0x30	0xFFFFFFFF
0x0D	0x34	0xFFFFFFFF
0x0E	0x38	0xFFFFFFFF
0x0F	0x3C	0xFFFFFFFF

2.1.4

Dispositivo de bloqueio

Existem dois bytes de bloqueio separados para o dispositivo:

- Um para bloquear a zona de configuração (que é controlada por LockConfig, byte 87).
- Um para bloquear as zonas de Dados e OTP (que são controladas por LockValue, byte 86). Isso permite as políticas de acesso para cada slot de zona de dados com base na configuração do slot.

Esses bloqueios são armazenados em bytes separados na zona de configuração e podem ser modificados apenas por meio do comando Lock.

Depois que uma zona de memória é bloqueada, não há como desbloqueá-la. O bloqueio da zona de Dados/OTP não significa que os slots não possam ser modificados. Os slots podem ser modificados com base nas políticas de acesso definidas pela configuração do Slot.

O dispositivo deve ser personalizado no fabricante do sistema com as informações de configuração desejadas e a zona de configuração deve ser bloqueada. Quando esse bloqueio estiver concluído, todas as gravações necessárias de informações públicas e secretas nos slots EEPROM devem ser executadas usando gravações criptografadas, se apropriado.

Após a conclusão das gravações nos dados e nas zonas OTP, o byte LockValue deve ser gravado nas zonas Data e OTP.

É vital que o byte LockValue seja definido como bloqueado antes da liberação do sistema que contém o dispositivo para o campo, a fim de proteger os dados armazenados nas zonas de dados e OTP. O não bloqueio dessas zonas pode permitir a modificação de quaisquer chaves secretas e pode levar a outros problemas de segurança.

Qualquer tentativa de ler ou gravar as seções Data ou OTP antes de bloquear a seção de configuração faz com que o dispositivo retorne um erro.

Entre em contato com a Microchip para obter serviços opcionais de personalização segura.

2.1.4.1 Bloqueio da zona de configuração Certos

bytes dentro da zona de configuração não podem ser modificados, independentemente do estado de LockConfig.

O acesso ao restante dos bytes dentro da zona é controlado usando o byte LockConfig na zona de configuração, conforme mostrado na tabela abaixo. Ao longo deste documento, se LockConfig for 0x55, a zona de configuração será desbloqueada; caso contrário, ele está bloqueado.

Tabela 2-9. Bloqueio de zona de configuração

estado de bloqueio	Acesso de leitura	Acesso de gravação
LockConfig == 0x55 (desbloqueado)	Ler	Escrever
LockConfig != 0x55 (bloqueado)	Ler	<nunca>

2.1.4.2 Bloqueio de zona de dados e OTP Ao

longo deste documento, se LockValue for 0x55, diz-se que as zonas de dados e OTP estão desbloqueadas; caso contrário, eles estão bloqueados.

Não há acesso de leitura nem gravação às zonas de dados e OTP antes do bloqueio da configuração zona.

Tabela 2-10. Dados e restrições de acesso à zona OTP

estado de bloqueio	Acesso de leitura	Acesso de gravação
LockValue == 0x55 (desbloqueado)	<nunca>	Escrever
LockValue != 0x55 (bloqueado)	leia(1)	Escreva(1)

Observação:

1. Com base na configuração do slot para um determinado slot.

2.1.4.3 Bloqueio de Zona OTP

As leituras e gravações da zona OTP dependem do estado dos bytes de modo LockConfig, LockValue e OTP na zona de configuração.

2.2**RAM estática (SRAM)**

O dispositivo inclui uma matriz SRAM que é usada para armazenar o comando de entrada ou resultado de saída, valores de computação intermediários e/ou uma chave efêmera. Todo o conteúdo desta memória é sempre invalidado sempre que o dispositivo entra em modo de hibernação ou a energia é removida. A chave efêmera é denominada TempKey e pode ser usada como uma entrada para os comandos MAC, HMAC, CheckMac, GenDig e DeriveKey. Também é usado como a chave de proteção de dados (criptografia ou descriptografia) pelos comandos de leitura e gravação. Consulte a seção [TempKey](#).

2.2.1 TempKey TempKey

é um registro de armazenamento na matriz SRAM que pode ser usado para armazenar um valor de resultado efêmero dos comandos Nonce, GenDig, CheckMac ou SHA. O conteúdo deste registrador nunca pode ser lido do dispositivo (embora o próprio dispositivo possa ler e usar o conteúdo internamente).

Este registro contém os elementos mostrados na tabela abaixo.

Tabela 2-11. Registro de armazenamento de chave temporária

Nome	Descrição do comprimento do bit	
TempKey	256 (32 bytes)	Nonce (faça o comando Nonce) ou Digest (faça o comando GenDig).
LockID	4	Se TempKey foi gerado por GenDig (veja os bits GenData e CheckFlag), esses bits indicam qual chave foi usada em seu cálculo. Os quatro bits representam um dos slots da zona de dados.

Nome	Descrição do comprimento do bit
SourceFlag	1 A fonte da aleatoriedade em TempKey: 0 = Número aleatório gerado internamente (Rand). 1 = Somente semente de entrada, sem geração aleatória interna (entrada).
GenData	1 0 = TempKey.SlotID não é significativo e é ignorado. 1 = O conteúdo de TempKey foi gerado pelo GenDig usando um dos slots na zona de dados (e TempKey.SlotID é significativo).
CheckFlag	1 0 = O conteúdo de TempKey foi gerado usando um Nonce, SHA ou GenDig sem uma restrição de chave CheckMac. 1 = O conteúdo de TempKey foi gerado pelo comando GenDig e pelo menos uma das chaves utilizadas nessa geração está restrita ao comando CheckMac (SlotConfig.CheckOnly é uma delas)
Válido	1 0 = As informações em TempKey são inválidas. 1 = As informações em TempKey são válidas.

Nesta especificação, o nome "TempKey" refere-se ao conteúdo do registro de dados de 32 bytes (256 bits). Os campos de bits restantes são referidos como TempKey.SourceFlag, TempKey.GenData e assim por diante.

O bit TempKey.Valid é zerado em qualquer uma das seguintes circunstâncias:

- Inicialização, hibernação, queda de energia, expiração do watchdog ou detecção de adulteração. O conteúdo de TempKey no entanto, são retidos quando o dispositivo entra no modo inativo.
- Após a execução de qualquer comando que não seja Nonce ou GenDig, independentemente de a execução do comando ser bem-sucedida ou não. Pode ser limpo pelo comando CheckMac, a menos que ocorra uma cópia bem-sucedida. Não é resolvido se houver um problema de comunicação, conforme evidenciado por um erro de verificação de redundância cíclica (CRC).
- Um erro durante a análise ou execução de um comando GenDig e/ou Nonce.
- A execução do GenDig substitui qualquer saída anterior do comando Nonce pela saída do comando GenDig. A execução do comando Nonce também substitui qualquer saída anterior do comando GenDig.

3. Recursos de segurança

3.1 Segurança física

O ATSHA204A incorpora vários recursos de segurança física projetados para proteger o conteúdo da EEPROM contra exposição não autorizada. As medidas de segurança incluem:

- Um escudo ativo sobre a peça
- Criptografia de memória interna
- Modos de teste seguros
- Proteção contra falhas
- Detecção de adulteração de tensão •
- Detecção de adulteração de temperatura

As chaves de transporte pré-programadas armazenadas no ATSHA204A são criptografadas de forma a tornar muito difícil a recuperação de seus valores usando análise externa.

Tanto o clock lógico quanto a tensão de alimentação lógica são gerados internamente, impedindo qualquer ataque direto a esses dois sinais usando os pinos do dispositivo.

3.2 Gerador de Números Aleatórios (RNG)

O ATSHA204A inclui um RNG de alta qualidade que retorna um número aleatório de 32 bytes ao sistema. O dispositivo combina esse número gerado com um número de entrada separado para formar um nonce que é armazenado no dispositivo em TempKey e pode ser usado por comandos subsequentes.

O sistema pode usar este RNG para qualquer finalidade. Uma finalidade comum seria como o desafio de entrada para o comando MAC em um dispositivo CryptoAuthentication separado. O dispositivo fornece um comando aleatório especial para tais fins, que não afeta o nonce armazenado internamente.

Para simplificar o teste do sistema, antes de bloquear a zona de configuração, o RNG sempre retorna o seguinte valor de 32 bytes:

0xFF FF 00 00 FF FF 00 00 ...

onde 0xFF é o primeiro byte lido do dispositivo e é usado para a mensagem SHA.

Para evitar ataques de repetição em dados criptografados que são passados de ou para o ATSHA204A, o dispositivo requer que um novo nonce gerado internamente seja incluído como parte da sequência de criptografia usada para proteger os dados que estão sendo lidos ou gravados. Para implementar esse requisito, a chave de proteção de dados gerada pelo GenDig e usada pelo comando Read ou Write deve usar o RNG interno durante a criação do nonce.

Números aleatórios são gerados a partir de uma combinação da saída de um RNG de hardware e um valor de semente interno, que não é acessível externamente. A semente interna é armazenada na EEPROM e normalmente é atualizada uma vez após cada inicialização ou ciclo de suspensão/ativação. Após a atualização, esse valor de semente é retido nos registros SRAM dentro do dispositivo que são invalidados se o dispositivo entrar no modo de hibernação ou se a energia for removida.

4. Informações gerais de E/S

A comunicação com o ATSHA204A é obtida através de um dos dois protocolos diferentes (I2C ou Single-Wire) e é selecionado com base no dispositivo solicitado:

- **Interface de fio único**

Usa uma única conexão GPIO no microprocessador do sistema conectado ao pino SDA no dispositivo. Ele permite o menor número de pinos conectores para qualquer entidade removível/substituível. A taxa de bits é de até 25,6 kb/s e é compatível com a sinalização UART padrão.

- **Interface 2C**

Este modo é compatível com a interface Microchip AT24C16 Serial EEPROM. São necessários dois pinos, Serial Data (SDA) e Serial Clock (SCL). A interface I2C suporta uma taxa de bits de até 1 Mb/s.

Os níveis mais baixos dos protocolos de E/S são descritos na Seção [Interface de fio único](#) e na Seção [I2C Interface](#). Além do nível do protocolo de E/S, ambas as interfaces transmitem exatamente os mesmos bytes de e para o dispositivo para implementar os comandos criptográficos e os códigos de erro documentados na seção [Comandos de segurança](#).

O dispositivo implementa um temporizador de vigilância interno à prova de falhas que o força a um modo de baixo consumo de energia após um determinado intervalo de tempo, independentemente de qualquer atividade atual. A programação do sistema deve levar isso em consideração. Consulte a seção [Watchdog Failsafe](#) para obter detalhes.

4.1

Ordenação de Bytes e Bits

Os dispositivos CryptoAuthentication usam um esquema de ordenação comum para bytes e também para a forma como os números e matrizes são representados nesta folha de dados:

- Todos os elementos agregados multibyte são tratados como matrizes de bytes e são processados na ordem recebida ou transmitida com o índice #0 primeiro.
- Números inteiros de 2 bytes (16 bits), normalmente Param2, aparecem primeiro no barramento LSB.

A ordem dos bits é diferente dependendo do canal de E/S usado:

- Na interface de fio único, os dados são transferidos de/para o ATSHA204A LSb primeiro no barramento.
- Na interface I2C , os dados são transferidos de/para o ATSHA204A MSb primeiro no barramento.

4.1.1 Exemplo de Saída

Os seguintes bytes são retornados nesta ordem no barramento por uma leitura de 32 bytes da seção de configuração com um endereço de entrada de 0x0000:

```
SN<0>, SN<1>, SN<2>, SN<3>, RevNum<0>, RevNum<1>, RevNum<2>, RevNum<3>, SN<4>, SN<5>, SN<6>, SN<7>, SN<8>, reservado,  
I2C_Enable, reservado, I2C_Address, OTPmode, SelectorMode, SlotConfig<0>.Read, SlotConfig<0>.Write, SlotConfig<1>.Read,  
SlotConfig<1>.Write, SlotConfig<2>.Read, SlotConfig<2>.Write, SlotConfig<3>.Read, SlotConfig<3>.Write,  
SlotConfig<4>.Read, SlotConfig<4>.Write, SlotConfig<5>.Read, SlotConfig<5>.Write
```

4.1.2 Exemplo de Mensagem MAC

Os bytes a seguir são passados para o mecanismo SHA para um comando MAC usando um valor de modo de 0x71 e um SlotID de slot x. No exemplo abaixo, K<x> indica o SlotID do slot x na zona de dados, com K<0> sendo o primeiro byte no barramento para uma leitura ou gravação nesse slot. OTP<0> indica o primeiro byte no barramento para uma leitura da zona OTP no endereço zero e assim por diante.

K<0>, K<1>, K<2>, K<3> ... K<31>, TempKey<0>, TempKey<1>, TempKey<2>, TempKey<3> ...
TempKey<31>, Opcode (=0x08), Mode (=0x71), Param2(LSB = 0xYY), Param2(MSB = 0x00), OTP<0>, OTP<1>,
OTP<2>, OTP<3>, OTP<4>, OTP<5>, OTP<6>, OTP<7>, OTP<8>, OTP<9>, OTP<10>, SN<8>, SN<4>, SN<5> ,
SN<6>, SN<7>, SN<0>, SN<1>, SN<2>, SN<3>.

Para obter mais detalhes sobre mensagens MAC, consulte a Seção [Comando MAC](#).

5. Interface de fio único

No modo de interface de fio único, as comunicações de e para o ATSHA204A ocorrem no pino SDA, um único fio cronometrado assincronamente e o pino SCL é ignorado.

Os valores de especificação de corrente de suspensão são garantidos somente se o pino SCL for mantido baixo ou deixado desconectado.

A estrutura geral de comunicações é uma hierarquia: A tabela abaixo mostra os tokens usados para a interface de fio único com uma porta RS-232 padrão. A porta UART do host deve ser configurada para palavras de dados de 7 bits e taxa de dados de 230,4 kBaud.

Tabela 5-1. Tokens de ativação e E/S

Tipo de token	Token Valor	Iniciar (1)	Token de ativação LSb: MSb							Parar (1)
			b0	b1	b2	b3	b4	b5	b6	
acordar (2)	0x00 0		0	0	0	0	0	0	0	1
Lógica 0 (3)	0x7D0		1	0	1	1	1	1	1	1
Lógica 1 (3)	0X7F 0		1	1	1	1	1	1	1	1

Observação:

1. Todos os Tokens devem ser precedidos por um LOW Start Pulse para sincronizar a captura de dados e terminar com um Valor de parada ALTO.
2. Um token de ativação cria um pulso baixo grande o suficiente para ativar o dispositivo.
3. Tokens de E/S Lógica 0, Lógica 1 representam um único bit de dados. 8 tokens de E/S seriam necessários para criar um único byte de dados.

Sinalizadores de E/S - Os sinalizadores consistem em oito tokens (bits) que transmitem a direção e o significado do próximo grupo de bits (se houver) que pode ser transmitido. Os sinalizadores são sempre transmitidos LSb primeiro.

Blocos - Blocos de dados seguem o comando e transmitem flags. Eles incorporam uma contagem de bytes e uma soma de verificação para garantir a transmissão de dados adequada.

Pacotes - Pacotes de bytes formam o núcleo do bloco (menos a contagem de bytes e CRC). Eles são os parâmetros de entrada ou saída de um comando CryptoAuthentication ou informações de status do ATSHA204A.

5.1 Tokens de E/S

Há vários tokens de E/S que podem ser transmitidos pela interface de fio único:

- **Entrada** (para o ATSHA204A)
 - Ativar: ativa o dispositivo do estado de suspensão ou ocioso.
 - Zero: envia um único bit do sistema para o dispositivo com valor zero.
 - One: envia um único bit do sistema para o dispositivo com valor um.
- **Saída** (do ATSHA204A)
 - ZeroOut: envia um único bit do dispositivo para o sistema com valor zero.
 - OneOut: envia um único bit do dispositivo para o sistema com valor um.

As formas de onda são as mesmas em qualquer direção. Existem algumas diferenças no tempo; no entanto, com base na expectativa de que o Host tenha um relógio muito preciso e consistente, enquanto o ATSHA204A tem

variabilidade significativa de peça a peça em seu gerador de relógio interno, devido à fabricação normal e flutuações ambientais.

O tempo de bit foi projetado para permitir que um UART padrão rodando a 230,4 kBaud transmita e receba os tokens com eficiência. Cada byte transmitido ou recebido pela UART corresponde a um único bit recebido ou transmitido pelo dispositivo. O UART precisa ser configurado com 7 bits de dados tendo 0x7F correspondendo a uma Lógica 1 e 0x7D correspondendo a uma Lógica 0.

O token Wake é especial porque requer um pulso baixo extra longo de tWLO no pino SDA (consulte a Tabela [AC Parameters – All I/O Interfaces](#)), que não pode ser confundido com os pulsos baixos mais curtos que ocorrem durante um token Data (Zero , One, ZeroOut ou OneOut). Os dispositivos que estão no estado ocioso ou suspenso ignoram todos os tokens de dados até que recebam um token de ativação legal. Não envie um token Wake para dispositivos que estão ativados, pois eles perdem a sincronização porque a forma de onda não pode ser resolvida nem como um legal nem como zero. Consulte a seção [Procedimentos de sincronização](#) para obter o procedimento para recuperar a sincronização.

5.2

Flags de E/

S O sistema é sempre o barramento mestre; portanto, antes de qualquer transação de I/O, o sistema deve enviar um flag de 8 bits para o dispositivo para indicar a operação de I/O a ser realizada posteriormente, conforme tabela abaixo.

Tabela 5-2. Sinalizadores de E/S

Nome	Valor	Significado
Suspensão (baixo consumo de energia)	0xCC	O ATSHA204A entra no modo de hibernação de baixo consumo de energia e ignora todas as transições de E/S subsequentes até o próximo sinalizador Wake. Todo o estado volátil do dispositivo é redefinido.
Parado	0xBB	O ATSHA204A entra no estado inativo e ignora todas as transições de E/S subsequentes até o próximo sinalizador Wake. O conteúdo dos registros de semente TempKey e RNG são retidos.
Comando	0x77	Escreva os bytes subsequentes em endereços sequenciais no buffer de comando de entrada.
Reservado	Todos os outros valores	Esses sinalizadores não devem ser enviados ao dispositivo.
Transmite	0x88	Comunica ao dispositivo para aguardar um tempo de retorno do barramento e então iniciar a transmissão de sua resposta ao bloco de comando transmitido anteriormente. Quando dados válidos estão no buffer de saída, o sinalizador de transmissão pode ser emitido repetidamente para o dispositivo para reenviar o buffer ao sistema.
Acordar	Consulte Interface Desperte o dispositivo do modo de baixo consumo de energia e redefina o contador do watchdog.	

5.2.1

Sinalizador de

transmissão O sinalizador de transmissão é usado para inverter o barramento para que o ATSHA204A possa enviar dados de volta ao sistema. Os bytes que o dispositivo retorna ao sistema dependem do estado atual do dispositivo e podem incluir status, código de erro ou resultados de comando.

Quando o dispositivo está ocupado executando um comando, ele ignora o pino SDA e quaisquer sinalizadores enviados pelo sistema. Veja a Seção [Códigos de Operação de Comando, Descrições Breves e Tempos de Execução](#) para atrasos de execução no dispositivo para cada tipo de comando. O sistema deve observar esses atrasos antes de tentar se comunicar com o dispositivo após o envio de um comando.

5.3**Sincronização**

Como o protocolo de comunicação é half-duplex, existe a possibilidade de que o sistema e o ATSHA204A possam perder a sincronização entre si. Para acelerar a recuperação, o dispositivo implementa um tempo limite que o força a dormir em determinadas circunstâncias.

5.3.1**Tempo Limite de E/S**

Após o recebimento de uma transição inicial para qualquer token de dados, o ATSHA204A espera que os bits restantes do token sejam recebidos adequadamente pelo dispositivo dentro do intervalo tTIMEOUT . A falha no envio de bits suficientes ou a transmissão de um token ilegal (um pulso baixo excedendo tZLO) faz com que o dispositivo entre no estado de suspensão após o intervalo tTIMEOUT .

O mesmo timeout se aplica durante a transmissão do bloco de comando. Após a transmissão de um sinalizador de comando legal, o circuito de tempo limite de E/S é ativado até que o último bit de dados esperado seja recebido.

Nota: O contador de tempo limite é redefinido após cada token legal e o tempo total para transmitir o comando pode exceder o intervalo tTIMEOUT enquanto o tempo entre os bits pode não.

O circuito de tempo limite de E/S é desabilitado quando o dispositivo está ocupado executando um comando.

5.3.2**Procedimentos de sincronização** Se o

dispositivo não estiver ocupado quando o sistema enviar um sinalizador de transmissão, o dispositivo deverá responder dentro de tTURNAROUND. Se o tempo tEXEC ainda não tiver passado, o dispositivo pode estar ocupado e o sistema deve pesquisar ou aguardar até que o tempo máximo tEXEC tenha decorrido. Se o dispositivo ainda não responder a um segundo sinalizador de transmissão dentro de tTURNAROUND, pode estar fora de sincronização. Neste ponto, o sistema pode tomar as seguintes medidas para restabelecer a comunicação:

1. Aguarde tTIMEOUT.
2. Envie o sinalizador de transmissão.
3. Se o dispositivo responder dentro de tTURNAROUND, o sistema poderá prosseguir com mais comandos.
4. Envie um token de ativação.
5. Aguarde tWHI.
6. Envie o sinalizador de transmissão.
7. O dispositivo deve responder com um status 0x11 dentro de tTURNAROUND, momento em que o sistema pode prossiga com os comandos.

Qualquer resultado de comando no buffer de E/S pode ser perdido quando o sistema e o dispositivo perdem a sincronização.

5.4**Compartilhando a interface**

Vários dispositivos CryptoAuthentication podem compartilhar a mesma interface, como segue:

1. O sistema emite um token Wake (Seção [Watchdog Failsafe](#)) para ativar todos os dispositivos.
2. O sistema emite o comando Pause para colocar todos os dispositivos, exceto um, no modo ocioso. Apenas o dispositivo restante então vê todos os comandos que o sistema envia. Quando o sistema terminar de falar com o dispositivo ativo, ele enviará um sinalizador ocioso, que os dispositivos ociosos ignorarão, mas colocará o único dispositivo ativo restante no modo ocioso. Consulte a Seção [Comando de Pausa](#) para obter mais detalhes.

As etapas 1 e 2 são repetidas para cada dispositivo no fio. Se o sistema tiver concluído a comunicação com o dispositivo final, ele deverá ativar todos os dispositivos e, em seguida, colocar todos os dispositivos em repouso para reduzir o consumo total de energia.

NOVO 204A

Interface de fio único

O dispositivo usa o byte seletor dentro da zona de configuração para determinar qual dispositivo permanece ativo. Somente aquele dispositivo com um valor de seletor que corresponda ao parâmetro de entrada do comando Pause permanece ativo. Para facilitar a configuração tardia de sistemas que usam o modo de compartilhamento de vários dispositivos, há suporte para os três recursos de atualização do byte seletor:

1. Atualizações ilimitadas

A qualquer momento pode ser executado o comando UpdateExtra para escrever o valor no campo seletor da zona Configuração. Para ativar este modo, defina o byte SelectorMode na zona de configuração para zero.

2. Atualização de campo única

Se o byte SelectorMode for definido como um valor diferente de zero e o byte seletor for definido como um valor zero antes de bloquear a zona de configuração. Então, a qualquer momento após o bloqueio da zona de Configuração, o comando UpdateExtra pode ser usado uma vez para definir o Seletor para um valor diferente de zero. O comando UpdateExtra não é afetado pelo byte LockValue.

3. Valor do seletor fixo

O byte do seletor nunca pode ser modificado após a zona de configuração ser bloqueada se SelectorMode e Selector estiverem definidos para valores diferentes de zero. O comando UpdateExtra sempre retorna um código de erro.

5.5

Exemplo de Transação

Wake (fio único)		
Hospedar		Dispositivo
Acordar	ÿ	
Transmite	ÿ	
	ÿ	Dados

Exemplo (fio único)		
Hospedar		Dispositivo
Acordar	ÿ	
Transmite	ÿ	
	ÿ	Dados
Comando	ÿ	
Dados	ÿ	
Transmite	ÿ	
	ÿ	Dados
Ocioso / eu sou	ÿ	

Tabela 5-3. Exemplo (fio único)

	Token de ativação 0x00	Transmitir 0x88	Contagem 0x04	Estado 0x11
Hospedar		0 0 0 1 0 0 0 1		
Dispositivo			0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0	

	CRC-16 0x33	CRC-16 0x43	Comando 0x77	Contar
Hospedar			1 1 1 0 1 1 1 0	
Dispositivo 1 1 0 0 1	1 0 0 1 1 0 0 0 0 1 0			
	Código de operação	Param1	Param2	Param2
Hospedar				
Dispositivo				
	Dados (0 – N)	Transmitir 0x88	Contar	Dados (1 – N)
Hospedar		0 0 0 1 0 0 0 1		
Dispositivo			XXXXXXXXXXXXXXXXXX	
	CRC-16	CRC-16	Parado	
Hospedar			1 1 0 1 1 1 0 1	
Dispositivo XXXXXXXXXXXXXXXXXX				

5.6

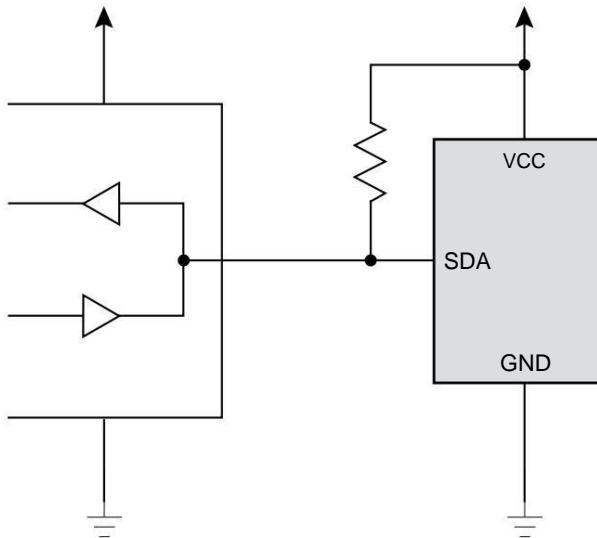
Configuração de fiação para interface de fio único A

interface de fio único permite a conexão do ATSHA204A a um host usando apenas um único pino SDA para transferir dados em ambas as direções. Esta interface não usa o pino SCL. O ATSHA204A não requer um capacitor de desvio quando conectado nesta configuração se a impedância dos sinais de alimentação e terra de volta à fonte de alimentação for baixa. A Microchip recomenda que um capacitor de desvio seja sempre usado para a melhor confiabilidade.

Para evitar a polarização direta do diodo interno e o consumo de corrente nos planos de energia do sistema, o pull-up do resistor no pino SDA deve ser conectado à mesma fonte conectada ao pino VCC ou a um trilho de tensão mais baixa .

Se os níveis de sinal para SDA forem diferentes da tensão VCC , consulte a seção de especificações paramétricas deste documento para garantir que os níveis de sinal sejam tais que a corrente de fuga excessiva seja minimizada quando em modos de hibernação. Esta situação pode ocorrer se o dispositivo ATSHA204A estiver fisicamente distante do dispositivo mestre do barramento ou se a tensão de alimentação do mestre do barramento for diferente da tensão de alimentação do ATSHA204A.

Figura 5-1. Configuração de 3 fios para interface de fio único



5.6.1

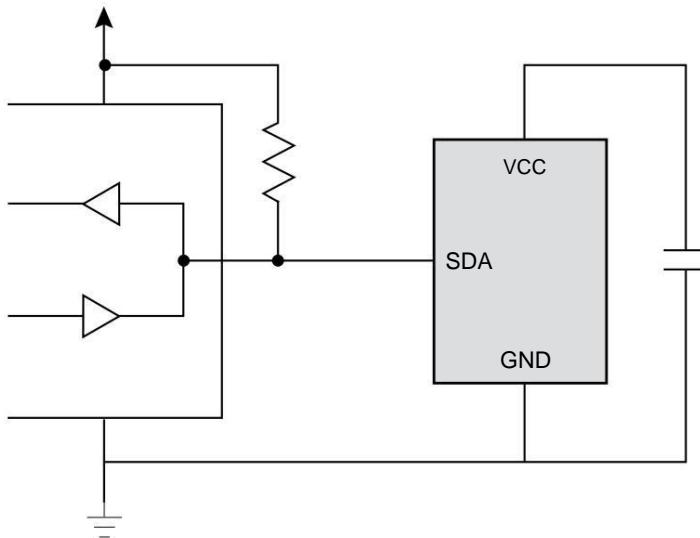
Configuração de 2

derivações Há uma chave interna que é conectada entre os pinos SDA e VCC que permite ao ATSHA204A roubar energia do pino SDA e armazená-la no capacitor de bypass. Neste caso, o pino VCC não precisa ser conectado à fonte de alimentação do Host. Esta configuração permite que a placa contendo o ATSHA204A e um capacitor de bypass seja conectada ao microprocessador do Host usando apenas dois cabos (ou seja, SDA e GND).

Se o nível de tensão de alimentação do sistema for de pelo menos 3 V, o resistor pull-up não deve ser maior que 1 K e o capacitor não deve ser menor que 0,03 μ F. O dispositivo opera adequadamente mantendo o VCC igual ou superior ao nível de especificação de 2V. Entre em contato com a Microchip para obter outras informações de configuração.

Em uma configuração de 2 derivações, o pino SDA deve ser direcionado para VCC usando um driver ativo capaz de fornecer ICC durante toda a execução de qualquer comando e um driver de totem deve ser usado para enviar dados ao dispositivo. A linha SDA deve depender apenas do resistor pull-up durante a transmissão de dados do ATSHA204A para o sistema

Figura 5-2. Configuração de 2 derivações para interface de fio único



6.**Interface 2C A**

interface I2C usa os pinos SDA e SCL para indicar vários estados de E/S para o ATSHA204A. Esta interface foi projetada para ser compatível em nível de protocolo com outros dispositivos I2C operando até 1 MHz.

O pino SDA deve ser puxado para cima com um resistor pull-up externo, pois o ATSHA204A inclui apenas um driver de dreno aberto em seu pino de saída. O mestre do barramento pode ser dreno aberto ou totem e, se for o último, deve ser tri-estado quando o ATSHA204A estiver conduzindo resultados no barramento. O pino SCL é uma entrada e deve ser acionado tanto alto quanto baixo o tempo todo por um dispositivo externo ou pull-up.

6.1**Condições de E/S**

O dispositivo ATSHA204A responde às seguintes condições de E/S descritas nas seções [Device is Asleep](#) e [Device is Awake](#).

6.1.1**O dispositivo está em**

suspensão Quando o dispositivo está em suspensão, ele ignora tudo, menos a condição de ativação.

- **Wake:** Se o SDA for mantido baixo por um período superior a tWLO, o dispositivo sai do modo de baixo consumo de energia e, após um atraso de tWHI, está pronto para receber comandos I2C. O dispositivo ignora quaisquer níveis ou transições no pino SCL quando o dispositivo está inativo ou em modo de espera e durante tWLO. Em algum ponto durante o tWHI, o pino SCL é ativado e as condições listadas na seção [Device is Awake](#) são atendidas.

A condição Wake requer que o processador do sistema conduza manualmente o pino SDA baixo para tWLO, ou que um byte de dados de 0x00 seja transmitido a uma taxa de clock suficientemente lenta para que SDA seja baixo por um período mínimo de tWLO. Quando o dispositivo está ativo, o hardware e/ou software I2C do processador normal pode ser usado para comunicações do dispositivo até e incluindo a sequência de E/S necessária para colocar o dispositivo de volta no modo de baixo consumo de energia (por exemplo, hibernação).

Quando há vários dispositivos ATSHA204A no barramento e a interface I2C é executada a 133 KHz ou mais lenta, a transmissão de determinados padrões de dados (como 0x00) faz com que todos os dispositivos ATSHA204A no barramento sejam ativados. Como os endereços de dispositivo subsequentes transmitidos ao longo do barramento podem corresponder apenas aos dispositivos desejados, os dispositivos não utilizados permanecem inativos e não causam conflitos de barramento.

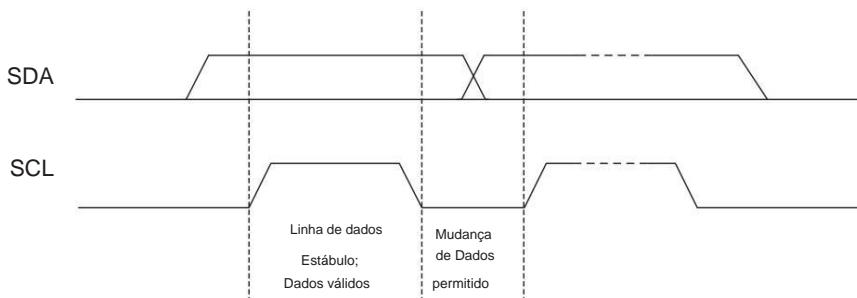
Não modo I2C , o dispositivo ignora uma sequência de ativação que é enviada quando o dispositivo já está ativado.

6.1.2**O dispositivo está ativo**

Quando o dispositivo está ativado, ele respeita as condições listadas abaixo:

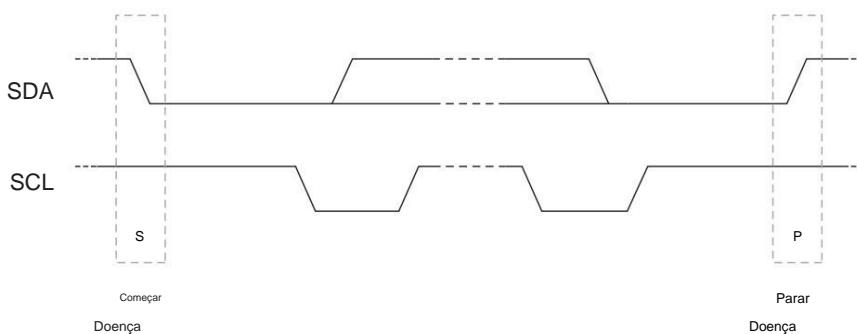
- **Data Zero:** se SDA é baixo e estável enquanto SCL vai de baixo para alto para baixo, então um bit zero está sendo transferido no ônibus. SDA pode mudar enquanto SCL está baixo.
- **Data One:** se SDA é alto e estável enquanto SCL vai de baixo para alto para baixo, então um bit está sendo transferido no ônibus. SDA pode mudar enquanto SCL está baixo.

Figura 6-1. Transferência de bits de dados na interface I2C

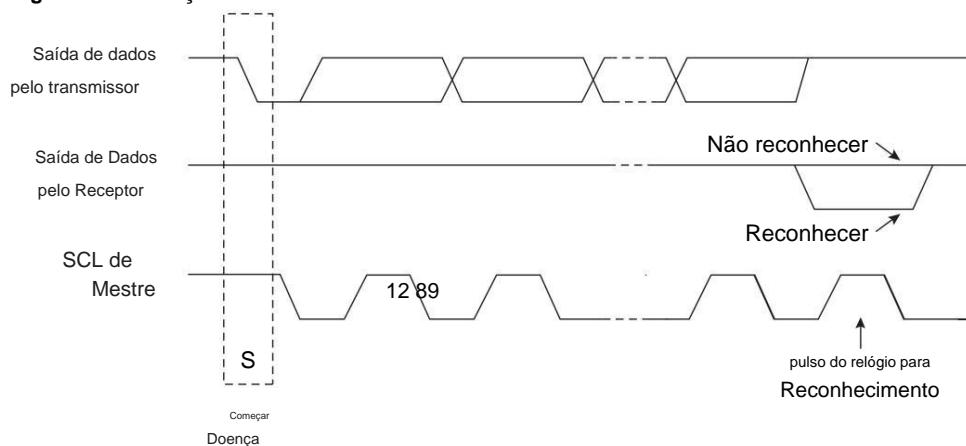


- **Condição inicial:** uma transição de alto para baixo de SDA com SCL alto é uma condição inicial, que deve preceder todos os comandos.
- **Condição de parada:** uma transição de baixo para alto de SDA com SCL alto é uma condição de parada. Depois disto é recebida pelo dispositivo, a transação de E/S atual termina. Na entrada, se o dispositivo tiver bytes suficientes para executar um comando, o dispositivo fará a transição para o estado ocupado e iniciará a execução. A Microchip recomenda que uma condição Stop seja enviada após o envio de qualquer pacote ao dispositivo, embora nem sempre seja necessário. O dispositivo inicia quando o número correto de bytes é recebido. Em caso de erro no barramento, o dispositivo zera o watchdog timer.

Figura 6-2. Condições de partida e parada na interface I2C



- **Reconhecimento (ACK):** no nono ciclo de clock após cada endereço ou byte de dados ter sido transferido, o receptor puxa o pino SDA para baixo para confirmar a recepção adequada do byte. • **Not Acknowledge (NACK):** alternativamente, no nono ciclo de clock após cada endereço ou byte de dados ter sido transferido, o receptor pode deixar o pino SDA alto para indicar que houve um problema com a recepção do byte ou que este byte foi concluído a transferência do bloco.

Figura 6-3. Condições NACK e ACK na interface I2C

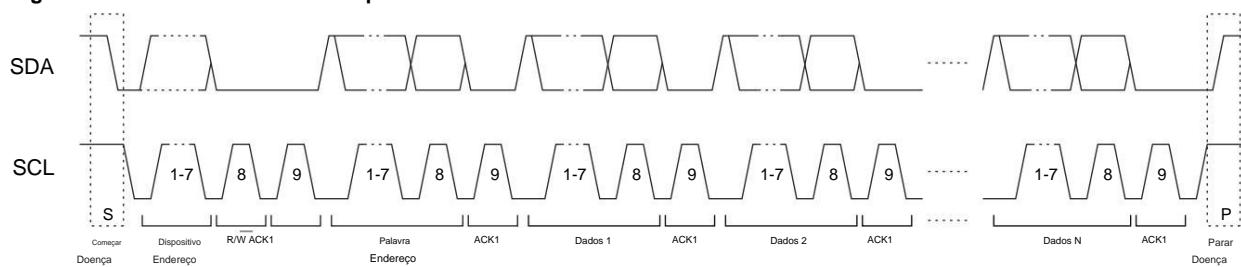
Vários dispositivos ATSHA204A podem compartilhar a mesma interface I2C se o byte I2C_Address for programado de forma diferente para cada dispositivo no barramento. Como seis dos bits do endereço do dispositivo são programáveis, o ATSHA204A também pode compartilhar a interface I2C com qualquer dispositivo I2C padrão, incluindo qualquer EEPROM serial. O bit 3 (também conhecido como TTL Enable) deve ser programado de acordo com os limites de entrada desejados e é fixado em uma determinada aplicação.

6.2

I2C Transmissão para o Dispositivo ATSHA204A

A transmissão de dados do sistema para o ATSHA204A está resumida na figura abaixo. A ordem de transmissão é a seguinte:

1. Condição inicial
2. Byte de endereço do dispositivo
3. Byte de endereço de palavra
4. Bytes de dados opcionais (1 a N)
5. Condição de parada

Figura 6-4. Transmissão I2C normal para um ATSHA204A

Nota: o SDA é reduzido pelo ATSHA204A durante os períodos de ACK

A tabela a seguir rotula os bytes da transação de E/S. A coluna de nome I2C fornece os nomes dos bytes conforme descritos na folha de dados AT24C16.

Tabela 6-1. Transmissão I2C para o ATSHA204A

ATSHA204A I2C Nome Direção Descrição		
Endereço do dispositivo	Dispositivo Endereço	para escravo

Este byte seleciona um determinado dispositivo na interface I2C. O ATSHA204A é selecionado se os bits 1 a 7 deste byte corresponderem aos bits 1 a

ATSHA204A I2C Nome Direção Descrição			
			7 do byte I2C_Address na zona de configuração. O bit 0 deste byte é o bit I2C R/W padrão e deve ser zero para indicar uma operação de gravação (os bytes que seguem o endereço do dispositivo viajam do mestre para o escravo).
Dados	Dados 1,N	To Slave	O bloco de entrada.

Como o dispositivo trata o buffer de entrada de comando como FIFO, o bloco de entrada pode ser enviado ao dispositivo em um ou mais blocos de comando I2C. O primeiro byte enviado ao dispositivo é a contagem, portanto, após o dispositivo receber esse número de bytes, ele ignora todos os bytes recebidos subsequentemente até que a execução seja concluída.

O sistema deve enviar uma condição de parada após o último byte de comando para garantir que o ATSHA204A inicie a computação do comando. A falha em enviar uma condição de parada pode eventualmente resultar em perda de sincronização (consulte a Seção [I2C Sincronização](#) para obter os procedimentos de recuperação).

6.2.1 Valores de endereço de palavra

Durante um pacote de escrita I2C, o ATSHA204A interpreta o segundo byte enviado como a palavra endereço, que indica a função do pacote, conforme descrito na tabela abaixo.

Tabela 6-2. Valores de endereço de palavra

Nome	Valor	Descrição
Reiniciar	0x00	Redefina o contador de endereços. A próxima transação de leitura ou gravação começa com o início do buffer de E/S.
Dormir (Potência baixa)	0x01	O ATSHA204A entra no modo de hibernação de baixo consumo de energia e ignora todas as transições de E/S subsequentes até o próximo sinalizador Wake. Todo o estado volátil do dispositivo é redefinido.
Parado	0x02	O ATSHA204A entra no estado inativo e ignora todas as transições de E/S subsequentes até o próximo sinalizador Wake. O conteúdo dos registros TempKey e RNG Seed são retidos.
Comando	0x03	Grave os bytes subsequentes em endereços sequenciais no buffer de comando de entrada que seguem as gravações anteriores. Esta é a operação normal.
Reservado	0x04 - 0xFF	Esses endereços não devem ser enviados para o dispositivo.

6.2.2 Polling de conclusão de comando

Após um comando completo ter sido enviado ao ATSHA204A, o dispositivo ficará ocupado até que a computação do comando seja concluída. O sistema tem duas opções para este atraso:

- **Polling**

O sistema deve aguardar tEXEC (típico) e então enviar uma sequência de leitura (Consulte a Seção [I2C Transmissão do Dispositivo ATSHA204A](#)). Se o dispositivo NACKs o endereço do dispositivo, então ele ainda está ocupado. O sistema pode atrasar por algum tempo ou enviar imediatamente outra sequência de leitura, novamente fazendo um loop no NACK. Após um atraso total de tEXEC (max), o dispositivo terá concluído a computação e poderá retornar os resultados. •

Atraso Único

O sistema deve esperar tEXEC (max), após o qual o dispositivo terá concluído a execução e o resultado poderá ser lido do dispositivo usando uma sequência de leitura normal.

6.3

I2C Transmissão do Dispositivo ATSHA204A

Quando o ATSHA204A está ativo e não ocupado, o barramento mestre pode recuperar o conteúdo do buffer atual do dispositivo usando uma leitura I2C . Se os resultados do comando válido estiverem disponíveis, o tamanho do bloco retornado é determinado pelo comando específico que foi executado (consulte a seção [Comandos de segurança](#)); caso contrário, o tamanho do bloco (e do primeiro byte retornado) é sempre quatro: contagem, status/erro e CRC de 2 bytes. A temporização do barramento é mostrada na Figura [I2C Sincronização de dados síncronos](#).

Tabela 6-3. Transmissão I2C de ATSHA204A

Nome	I2C Nome	Direção	Descrição
Dispositivo Endereço	Dispositivo Endereço	para escravo	Este byte seleciona um determinado dispositivo na interface I2C e o ATSHA204A é selecionado se os bits 1 a 7 deste byte corresponderem aos bits 1 a 7 do byte I2C_Address na zona de configuração. O bit 0 deste byte é o pino I2C R/W padrão e deve ser um para indicar que os bytes que seguem o endereço do dispositivo viajam do escravo para o mestre (lido).
Dados	Dados 1,N	Dominar	O bloco de saída, consistindo no byte de contagem e status/erro ou no pacote de saída seguido pelo CRC de 2 bytes conforme a Seção 8.2.

As saídas de status, erro ou comando podem ser lidas repetidamente pelo mestre. Cada vez que um comando Read é enviado ao ATSHA204A pela interface I2C , o dispositivo transmite o próximo byte sequencial no buffer de saída. Consulte a seção a seguir para obter detalhes sobre como o dispositivo lida com o contador de endereços.

Se o ATSHA204A estiver ocupado, ocioso ou adormecido, ele fará NACK no endereço do dispositivo em uma sequência de leitura. Se um comando parcial tiver sido enviado ao dispositivo, ele fará NACK no endereço do dispositivo, mas flutuará no barramento durante os intervalos de dados.

6.4 Contador de endereços

As gravações e/ou leituras do buffer de E/S ATSHA204A na interface I2C são tratadas como se o dispositivo fosse um FIFO. Tanto o byte I2C quanto os protocolos de leitura/gravação de bloco podem ser usados. O número de bytes transferidos com cada sequência de blocos não afeta a operação do dispositivo.

O primeiro byte transmitido ao dispositivo é tratado como o byte de contagem. Qualquer tentativa de enviar mais do que este número de bytes ou qualquer tentativa de gravar além do final do buffer de E/S (84 bytes) faz com que o ATSHA204A execute NACK esses bytes.

Após o host gravar um único byte de comando no buffer de entrada, os comandos de leitura do dispositivo do host são proibidos até que o dispositivo conclua a execução do comando. As tentativas de leitura do dispositivo antes do envio do último byte de comando resultam em um ACK do endereço do dispositivo, mas todos (0xFF) no barramento. Se o mestre tentar enviar um byte de leitura para o dispositivo durante a execução do comando, o dispositivo irá NACK o endereço do dispositivo.

Os dados podem ser lidos do dispositivo nas três condições a seguir:

- Ao ligar, o byte único, 0x11 (consulte a seção [Códigos de operação de comando, breves descrições e tempos de execução](#)), pode ser lido dentro de um bloco de quatro bytes.
- Se um bloco completo foi recebido pelo dispositivo, mas há algum erro na análise ou execução do comando, um único byte de código de erro está disponível, também dentro de um bloco de quatro bytes. • Após a conclusão da execução do comando, de 1 a 32 bytes do resultado do comando estão disponíveis para serem lido dentro de um bloco de 4 a 35 bytes.

Qualquer tentativa de ler além do final do buffer de saída válido retorna 0xFF para o sistema e o contador de endereços não retorna ao início do buffer.

Pode haver situações em que o sistema deseja reler o buffer de saída, por exemplo, quando a verificação do CRC revela um erro. Neste caso, o mestre deve enviar uma sequência de dois bytes para o ATSHA204A consistindo no endereço correto do dispositivo e um endereço de palavra de 0x00 (Reset, conforme Tabela Tabela 6-2) , seguido de uma condição de parada. Isso faz com que o contador de endereços seja zerado e permite que os dados sejam reescritos (releidos) para (do) dispositivo. Essa sequência de redefinição de endereço não proíbe operações de leitura subsequentes se os dados estiverem disponíveis para leitura no buffer de E/S antes da execução da sequência.

Após uma ou mais operações de leitura para recuperar os resultados de uma execução de comando, a primeira operação de gravação redefine o contador de endereços para o início do buffer de E/S.

6.5

I2C Sincronização É

possível que o sistema perca a sincronização com a porta de E/S no ATSHA204A, por exemplo, devido a uma reinicialização do sistema, ruído de E/S ou alguma outra condição. Nessas circunstâncias, o ATSHA204A pode não responder como esperado, pode estar inativo ou pode estar transmitindo dados durante um intervalo em que o sistema espera enviar dados. Qualquer resultado de comando no buffer de E/S pode ser perdido quando o sistema e o dispositivo perdem a sincronização. Para ressincronizar, o seguinte procedimento deve ser seguido:

1. Para garantir uma redefinição do canal de E/S, o sistema deve enviar a sequência de redefinição do software I2C padrão, do seguinte modo:

- Uma condição inicial.
- Nove ciclos de SCL com SDA mantido alto.
- Outra condição de início.
- Uma condição de parada.

Deve então ser possível enviar uma sequência de leitura e se a sincronização for concluída corretamente, o ATSHA204A confirmará o endereço do dispositivo. O dispositivo retorna dados ou deixa o barramento flutuante (que o sistema interpreta como um valor de dados de 0xFF) durante os períodos de dados.

Se o dispositivo confirmar o endereço do dispositivo, o sistema deve redefinir o contador de endereço interno para forçar o ATSHA204A a ignorar qualquer comando de entrada parcial que possa ter sido enviado. Isso pode ser feito enviando uma sequência de gravação para o endereço de palavra 0x00 (Redefinir), seguido por uma condição de parada.

2. Se o dispositivo não responder ao endereço do dispositivo com um ACK, ele pode estar inativo. Nesse caso, o sistema deve enviar um token de ativação completo e aguardar tWHI após a borda de subida. O sistema pode então enviar outra sequência de leitura e, se a sincronização for concluída, o dispositivo confirmará o endereço do dispositivo.
3. Se o dispositivo ainda não responder ao endereço do dispositivo com um ACK, ele pode estar ocupado executando um comando. O sistema deve aguardar o maior tEXEC (max) e então enviar a sequência de leitura, que é confirmada pelo dispositivo.

6.6**Exemplo de Transação****Tabela 6-4. Acorde (I2C)**

Acorde (I2C)		
Hospedar		Dispositivo
Começar	ÿ	
Acordar	ÿ	
Parar	ÿ	
Começar	ÿ	
Endereço Escravo / R	ÿ	
	ÿ	Dados
Parar	ÿ	

Tabela 6-5. Exemplos de transação

Exemplo (I2C)		
Hospedar	ÿ	Dispositivo
Começar	ÿ	
Acordar	ÿ	
Parar	ÿ	
Começar	ÿ	
Endereço Escravo / R	ÿ	
	ÿ	Dados
Parar	ÿ	
Começar	ÿ	
Endereço Escravo / W	ÿ	
Comando	ÿ	
Dados	ÿ	
Parar	ÿ	
Começar	ÿ	
Endereço Escravo / R	ÿ	
	ÿ	Dados
Parar	ÿ	
Começar	ÿ	
Endereço Escravo / W	ÿ	
Inativo / Dormir	ÿ	
Parar	ÿ	

7. características elétricas

7.1 Classificações Máximas Absolutas

Temperatura de operação	ÿ40°C a +85°C
Temperatura de armazenamento	ÿ65°C a + 150°C
Tensão operacional máxima	6,0 V
Corrente de Saída DC	5,0 mA
Tensão em qualquer pino	0,5V a (VCC + 0,5V)
Classificações ESD:	
Modelo de Corpo Humano (HBM) ESD	>4kV
Modelo de Dispositivo de Carga (CDM) ESD	>1kV

Observação: tensões além das listadas em “Classificações máximas absolutas” podem causar danos permanentes ao dispositivo. Esta é apenas uma classificação de estresse e a operação funcional do dispositivo nessas ou em qualquer outra condição além das indicadas nas seções operacionais desta especificação não está implícita. A exposição a condições de classificação máxima absoluta por períodos prolongados pode afetar a confiabilidade do dispositivo.

7.2 Confiabilidade

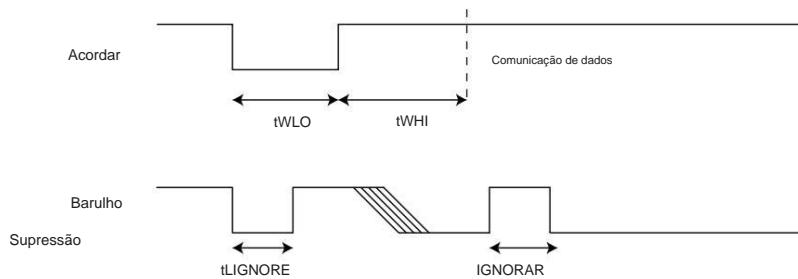
O ATSHA204A é fabricado com a alta confiabilidade de uma tecnologia de fabricação Microchip CMOS EEPROM.

Tabela 7-1. Confiabilidade EEPROM

Parâmetro	min.	típica	máximo	Unidades
Write Endurance (cada byte a 25°C)	100.000			Ciclos de Gravação
Retenção de dados (a 55°C)	10			Anos
Retenção de dados (a 35°C)	30	50		Anos
Ler Resistência	Ilimitado		Ciclos de leitura	

7.3 Parâmetros CA — Todas as Interfaces de E/S

Figura 7-1. Diagrama de temporização CA - todas as interfaces de E/S



NOVO 204A

características elétricas

Tabela 7-2. Parâmetros CA — Todas as interfaces de E/S

Parâmetro	Símbolo	Direção		Min	Type	Max	Unidade	Notas
Wake Low Duração	tWLO		Para cripto Autenticação	60			—	ys SDA pode ser estável em níveis altos ou baixos durante intervalos prolongados de sono.
Atraso de inicialização tPU			Para cripto Autenticação	100(1)			ys	Tempo mínimo entre VCC > VCC min antes da medição de tWLO.
Wake High Atraso para comunicação de dados	tWHI		Para cripto Autenticação	2.5			ms	SDA deve ser estável alto por toda esta duração.
Falha lateral alta Filtrar em Ativo	tIGNORE_A		Para Criptografia Autenticação	45			ns	Pulsos menores que isso em largura são ignorados pelo dispositivo, independentemente de seu estado quando ativo.
Falha do lado inferior Filtrar em Ativo	tIGNORE_A		Para Criptografar Autenticação	45			ns	Pulsos mais curtos que isso em width são ignorados pelo dispositivo, independentemente de seu estado quando ativo.
Falha lateral alta Filtrar em modo de suspensão	tIGNORE_S		Para Criptografia Autenticação	15			ys	Pulsos menores do que isso em largura são ignorados pelo dispositivo quando em modo de hibernação.
Falha do lado inferior Filtrar em modo de suspensão	tIGNORE_S		Para Criptografar Autenticação	15			ys	Pulsos menores do que isso em largura são ignorados pelo dispositivo quando em modo de hibernação.
Watchdog redefinir tWATCHDOG para criptografia Autenticação				0,7(1)	1,3	1,7	s	máx. tempo desde a ativação até que o dispositivo seja forçado a entrar no modo de hibernação (consulte a seção Watchdog Failsafe).

Observação:

1. Estes parâmetros são garantidos por caracterização, mas não testados.

7.3.1 Parâmetros CA — Interface de fio único

Figura 7-2. Diagrama de temporização AC - Interface de fio único

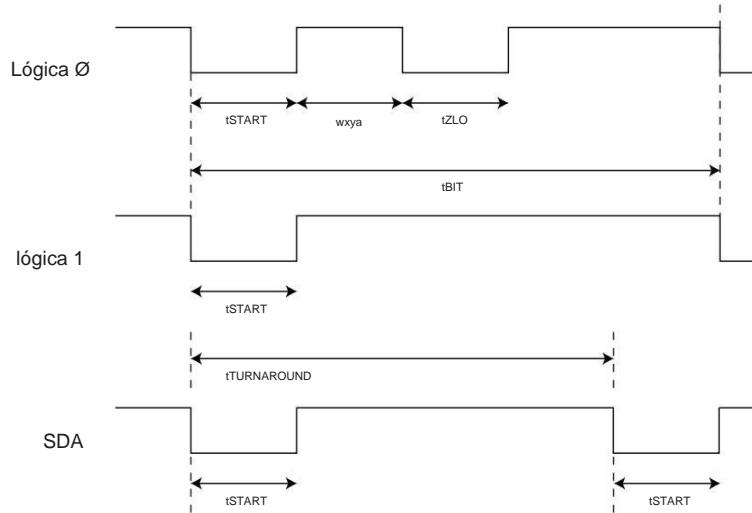


Tabela 7-3. Parâmetros CA — Interface de fio único A menos

que especificado de outra forma, aplicável de TA = $\bar{y}40^{\circ}\text{C}$ a $+85^{\circ}\text{C}$, VCC = +2,0 V a +5,5 V, CL =100 pF.

Parâmetro	Símbolo	Direção	Min	Type	Max	Unidade	Notas
pulso inicial Duração(1)	tSTART	Para cripto Autenticação	4,10	4,34	4,56	ÿs	
		De cripto Autenticação	4,60	6,00	8,60	ÿs	
Transmissão zero pulso alto(1)	wxya	Para cripto Autenticação	4,10	4,34	4,56	ÿs	
		De cripto Autenticação	4,60	6,00	8,60	ÿs	
Transmissão zero Pulso baixo(1)	tZLO	Para cripto Autenticação	4,10	4,34	4,56	ÿs	
		De cripto Autenticação	4,60	6,00	8,60	ÿs	
Tempo de bits (1)	tBIT	Para cripto Autenticação	37	39	—	ÿs	Se o tempo de bit exceder tTIMEOUT, o ATSHA204A pode entrar no estado de hibernação. Consulte a Seção Tempo Limite de E/S para obter detalhes específicos.
		De cripto Autenticação	41	54	78	ÿs	
Atraso de retorno tTURNAROUND		De cripto Autenticação	64	80	131	ÿs	O ATSHA204A inicia a primeira transição baixa após este intervalo de tempo após o início do último bit (tBIT) do sinalizador de transmissão.

NOVO 204A

características elétricas

Parâmetro	Símbolo	Direção	Min	Type	Max	Unidade	Notas
		Para cripto Autenticação	93			ÿs	Após o ATSHA204A transmite o último bit de um bloco, o sistema deve esperar este intervalo antes de enviar o primeiro bit de um sinalizador.
Tempo Limite de E/S	tTIMEOUT	Para cripto Autenticação	45	65	85	ms	O ATSHA204A pode transitar para o estado de hibernação se o barramento estiver inativo por mais tempo do que esta duração. Consulte a Seção Tempo Limite de E/S para obter detalhes específicos.

Observação:

1. tSTART, tZLO, tZHI e tBIT são projetados para serem compatíveis com um UART padrão rodando em 230,4 kBaud para transmissão e recepção. O UART deve ser definido para sete bits de dados, sem paridade e um bit de parada.

7.3.2 Parâmetros CA — Interface I2C Figura 7-3.

Temporização de dados síncronos I2C tHIGH

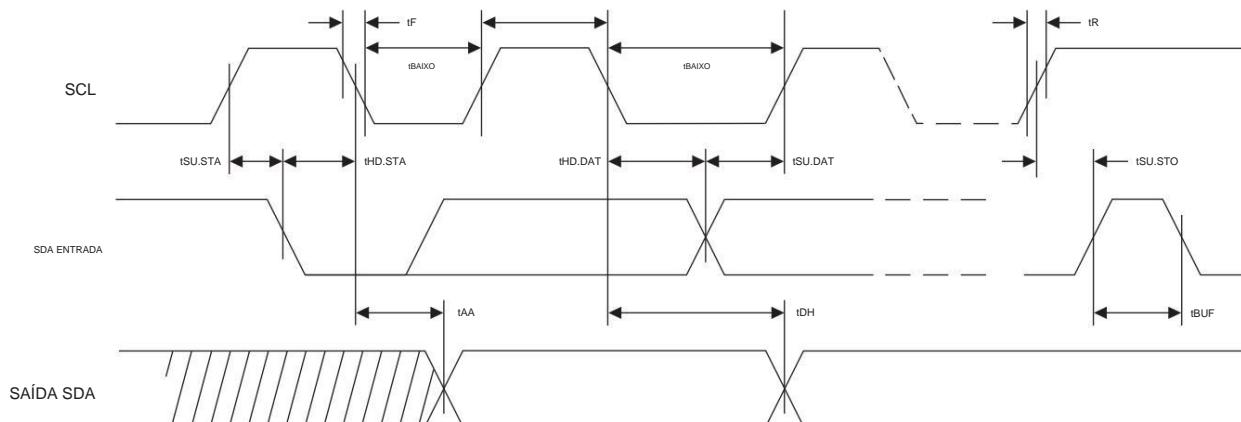


Tabela 7-4. Características CA da interface I2C A menos que

especificado de outra forma, aplicável na faixa operacional recomendada de TA = $\bar{y}40^{\circ}\text{C}$ a $+85^{\circ}\text{C}$, VCC = +2,0 V a +5,5 V, CL = 1 portão TTL e 100 pF.

Parâmetro do símbolo	Unidades Mín. Máx.
fSCK	Frequência do Relógio SCK
	Ciclo de trabalho do relógio SCK
coxa	SCK Tempo Máximo
tBAIXO	SCK tempo baixo
tSU.STA	Hora de inicio da configuração
iHD.STA	Iniciar tempo de espera
Tempo de configuração de parada tSU.STO	

NOVO 204A**características elétricas**

Parâmetro do símbolo		Unidades Mín. Máx.	
Dados tSU.DAT no tempo de configuração		100	ns
Dados tHD.DAT em tempo de espera		0	ns
tR	Tempo de subida de entrada (1)	300	ns
tF	Tempo de queda de entrada (1)	100	ns
tAA	Relógio baixo para saída de dados válida	50 550	ns
tDH	Tempo de espera de saída de dados	50	ns
tBUF	O barramento de tempo deve estar livre antes que uma nova transmissão possa começar.(1)	500	ns

Observação:

1. Os valores são baseados na caracterização, mas não são testados.

Condições de medição CA:

- RL (conecta entre SDA e VCC): 1,2 k Ω (para VCC +2,0V a +5,0V)
- Tensões de pulso de entrada: 0,3VCC a 0,7VCC
- Tempos de subida e descida de entrada: \geq
- 50 ns Tensão de referência de temporização de entrada e saída: 0,5VCC

7.4 Parâmetros DC — Todas as Interfaces de E/S**Tabela 7-5. Parâmetros CC em todas as interfaces de E/S**

Parâmetro	Símbolo	Min	Max	Unidade	Notas
Ambiente Operacional Temperatura	VOLTOPO PARA	-40	85°C		
Tensão de alimentação VCC		2.0	5,5V		
Fonte de alimentação ativa Atual	ICC	500		μ A	0°C \geq +70°C, VCC = 3,3V.
		— 2 mA	-40°C \geq +85°C, VCC = 5,5V.		
Fonte de alimentação ociosa Atual	EU OCIOSO	200		μ A	Quando o dispositivo está no modo inativo, VCC = 3,3 V, VSDA e VSCL < 0,3 V ou > VCC-0,3.
Sono atual	EU DURMO	30 150 nA			Quando o dispositivo está no modo de hibernação, VCC \geq 3,6 V, VSDA e VSCL < 0,3 V ou > VCC-0,3, TA e 55°C
Baixa Tensão de Saída	VOL		0,4V		Quando o dispositivo está no modo ativo, VCC = 2,5 – 5,5V.
Corrente baixa de saída	CAVALO		4mA		Quando o dispositivo está no modo ativo, VCC = 2,5 – 5,5 V, VOL = 0,4 V.

NOVO 204A

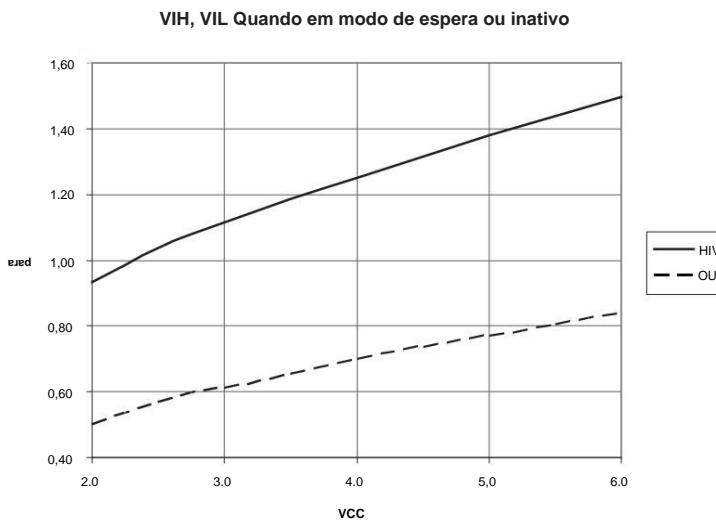
características elétricas

7.4.1

Especificações de VIH e VIL

Os limites de tensão de entrada quando em modo de espera ou ocioso dependem do nível VCC , como mostrado não gráfico em VIH e VIL Quando em modo de espera ou ocioso.

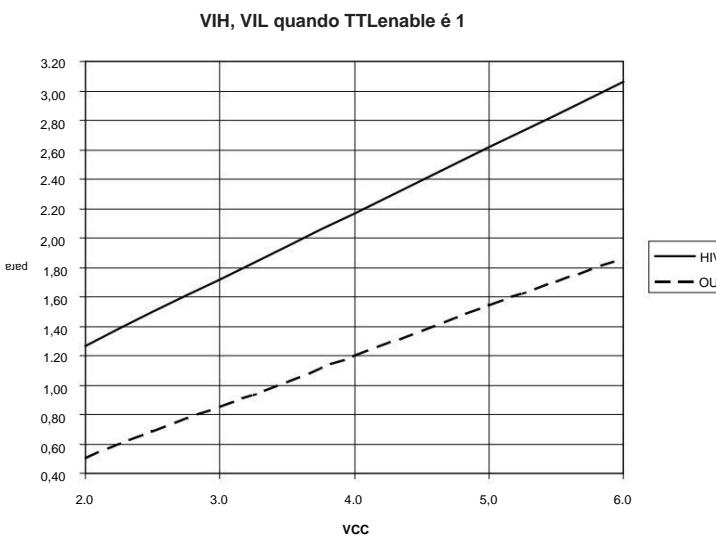
Figura 7-4. VIH e VIL quando em modo de espera ou inativo



Quando o dispositivo está ativo (por exemplo, não em modo de espera ou inativo), os limites de tensão de entrada são diferentes, dependendo do estado de TTLenable (bit 3) dentro do byte I2C_Address armazenado na zona de configuração da EEPROM. Quando uma tensão comum é usada para o pino ATSHA204A VCC e o resistor pull-up de entrada, esse bit deve ser definido como um, o que permite que os limites de entrada rastreiem a alimentação, conforme mostrado na [Figura 7-5](#).

Se a tensão fornecida ao pino VCC do ATSHA204A for diferente da tensão do sistema à qual o resistor pull-up de entrada está conectado, o projetista do sistema pode optar por definir TTLenable como zero. Isso permite um limite de entrada fixo e o sinal de entrada deve atender aos níveis de limite conforme mostrado na [Tabela 7-6](#).

Figura 7-5. VIH e VIL Quando TTLenable = 1 em todas as interfaces de E/S



NOVO 204A

características elétricas

Tabela 7-6. VIL e VIH Quando TTLenable = 0 em todas as interfaces de E/S

Parâmetro	Símbolo	Min	Tipo	Max	Unidade	Notas
VIL de baixa tensão de entrada			GND - 0,5		0,5	EM Quando o dispositivo está ativo e o bit TTLenable na memória de configuração é zero; caso contrário, veja acima.
VIH de alta tensão de entrada			1,5		VCC + 0,5 V	Quando o dispositivo está ativo e o bit TTLenable está ativado a memória de configuração é zero; caso contrário, veja acima.

8. Comandos de segurança

8.1 Blocos de E/S

Independentemente do protocolo de I/O utilizado (por exemplo, Single-Wire ou I2C), os comandos são enviados ao dispositivo e as respostas recebidas do dispositivo, dentro de um bloco que é construído da seguinte forma:

Tabela 8-1. Blocos

Nome do byte	Significado
0	Contar
Dados 1 a N-2	Se a entrada do dispositivo; comandos e parâmetros. Se a saída do dispositivo; resposta do dispositivo com base no comando que está sendo chamado.
N-1 a N	Soma de verificação CRC-16. O polinômio CRC é 0x8005.

O ATSHA204A foi projetado de forma que o valor de contagem no bloco de entrada seja consistente com os requisitos de tamanho especificados nos parâmetros de comando. Se o valor da contagem for inconsistente com o opcode do comando e/ou os parâmetros do pacote, o ATSHA204A responde de maneiras diferentes, dependendo do comando específico. A resposta inclui uma indicação de erro ou alguns bytes de entrada são silenciosamente ignorados.

8.1.1 Códigos de status/erro

O dispositivo não possui um registrador de status dedicado, então a saída FIFO é compartilhada entre os resultados de status, erro e comando. Todas as saídas do dispositivo são retornadas ao sistema como blocos completos.

Depois que o dispositivo recebe o primeiro byte de um bloco de comando de entrada, o sistema não pode ler nada do dispositivo até que o sistema tenha enviado todos os bytes para o dispositivo.

Após o despertar e após a execução de um comando, pode haver bytes de erro, status ou resultado no registrador de saída do dispositivo que podem ser recuperados pelo sistema. Quando o comprimento desse bloco é de quatro bytes, os códigos retornados são detalhados na tabela a seguir. Alguns comandos retornam mais de quatro bytes quando são executados com sucesso. A descrição do pacote resultante está listada na seção de comando abaixo.

Erros CRC sempre são retornados antes de qualquer outro tipo de erro. Indicam que ocorreu algum tipo de erro de I/O e que o comando pode ser reenviado para o dispositivo. Se um comando inclui erros de análise e execução, não há nenhuma precedência específica aplicada; portanto, um erro de execução pode ocorrer antes de um erro de análise e/ou o inverso pode ocorrer.

Tabela 8-2. Códigos de status/erro em blocos de 4 bytes

Descrição do estado	Erro/	Descrição de status
Comando bem sucedido Execução	0x00	Comando executado com sucesso.
Checkmac erro de comparação	0x01	O comando CheckMac foi enviado corretamente para o dispositivo, mas a resposta do Cliente de entrada não correspondeu ao valor esperado.
Erro de análise	0x03	O comando foi recebido corretamente, mas o comprimento, opcode do comando ou os parâmetros são ilegais, independentemente do estado (configuração volátil e/ou EEPROM) do ATSHA204A.

Descrição do estado	Erro/ Código de status	Descrição de status
		Alterações no valor dos bits de comando devem ser feitas antes de tentar novamente.
Erro de execução	0x0F	O comando foi recebido corretamente, mas não pode ser executado pelo dispositivo em seu estado atual. Alterações no estado do dispositivo ou no valor dos bits de comando devem ser feitas antes de tentar novamente.
Após o despertar, antes do primeiro Comando	0x11	Indicação de que o ATSHA204A recebeu um token de ativação adequado.
CRC ou outro Erro de comunicação	0xFF	O comando não foi recebido corretamente pelo ATSHA204A e deve ser retransmitido pelo driver de E/S no sistema. Nenhuma tentativa foi feita para analisar ou executar o comando.

8.2

Sequência de suspensão

Após a conclusão do uso do sistema do ATSHA204A, o sistema deve emitir uma sequência de suspensão para colocar o dispositivo no modo de baixo consumo de energia. Usando a interface I2C , essa sequência consiste no endereço do dispositivo adequado seguido pelo sinalizador de suspensão seguido por uma condição de parada. Essa transição para o estado de baixo consumo de energia causa uma redefinição completa do mecanismo de comando interno do dispositivo e do buffer de entrada/saída. Ele pode ser enviado para o dispositivo a qualquer momento quando estiver ativo e não ocupado.

8.3

Sequência ociosa Se

a sequência total de comandos necessários exceder tWATCHDOG, o dispositivo entrará automaticamente no modo de suspensão e perderá todas as informações armazenadas nos registradores voláteis. Essa ação pode ser evitada colocando o dispositivo no estado inativo antes da conclusão do intervalo de vigilância. Quando o dispositivo recebe o token Wake, ele reinicia o cronômetro de vigilância e a execução pode continuar.

Usando a interface I2C , essa sequência inativa consiste no endereço do dispositivo adequado seguido pelo valor de 0x02 como o endereço da palavra seguido por uma condição de parada. Ele pode ser enviado para o dispositivo a qualquer momento quando estiver ativo e não ocupado.

Se TempKey foi criado como resultado do modo de cópia do comando CheckMac, ele não será retido quando a peça entrar em estado ocioso.

8.4 Watchdog Failsafe Um contador

watchdog começa dentro do dispositivo depois que o ATSHA204A recebe um token Wake. Depois de tWATCHDOG, o dispositivo entra no modo de hibernação independentemente de alguma transmissão de E/S ou execução de comando estar em andamento. Não há outra maneira de redefinir o contador a não ser colocar o dispositivo no modo de suspensão ou ocioso e, em seguida, ativá-lo novamente.

O temporizador watchdog é implementado como um mecanismo à prova de falhas para que não importa o que aconteça no lado do sistema ou dentro do dispositivo, incluindo qualquer problema de sincronização de E/S, o consumo de energia cai automaticamente para o nível de hibernação ultrabaixo.

O dispositivo redefine os valores armazenados na SRAM e nos registros de status internos quando faz a transição para o estado de hibernação, no entanto, se o dispositivo for explicitamente colocado no modo inativo por meio do I/O apropriado

sequência, o dispositivo retém o conteúdo dos dois registradores SRAM (por exemplo, TempKey e RNG seed).

Normalmente, todas as sequências de comando devem ser concluídas no tWATCHDOG se exigirem um estado armazenado nos registradores SRAM. O software do sistema pode usar esse mecanismo de modo inativo entre os comandos para implementar uma sequência de comando mais longa do que pode ser concluída durante um único intervalo de vigilância.

8.5 Sequência de Comando

8.5.1 Pacotes de Comando

O pacote de comando é dividido conforme mostrado na tabela a seguir:

Tabela 8-3. Pacotes de comando

Byte # Nome		Significado
0	Comando	Sinalizador de comando (consulte Valores de endereço de palavra para o modo de operação I2C e sinalizadores de E/S para o modo de interface de fio único). Não incluído no campo Contagem ou CRC.
1	Contar	Tamanho do pacote: Inclui Count, Opcode, Param1, Param2, Data e CRC. Não inclui sinalizador de comando.
2	Operação Opcode	ATSHA204A sendo chamada.
3	Param1	Primeiro Parâmetro. Um byte sempre presente.
4 – 5	Param2	Segundo Parâmetro. Dois bytes sempre presentes.
	Dados	Dados opcionais baseados no comando que está sendo chamado.
Soma de verificação N-1 a N		CRC-16. O polinômio CRC é 0x8005. Inclui Count, Opcode, Param1, Param2 e Data. Não inclui sinalizador de comando.

Após o ATSHA204A receber todos os bytes em um bloco, o dispositivo passa para o estado ocupado e tenta executar o comando. Nem o status nem os resultados podem ser lidos do dispositivo quando ele está ocupado.

Durante este tempo, a interface I/O do dispositivo ignora todas as transições SDA independentemente da interface I/O selecionada. Os atrasos de execução do comando estão listados na seção [Operações de leitura na zona de dados](#).

Se um número insuficiente de bytes for enviado ao dispositivo quando ele estiver no modo de um fio, o dispositivo fará a transição automaticamente para o estado de suspensão de baixa energia após o intervalo tTIMEOUT . No modo I2C , o dispositivo continua a aguardar os bytes restantes até que o limite do timer de vigilância, tWATCHDOG, seja atingido ou uma condição Iniciar/Parar seja recebida pelo dispositivo.

Nas descrições de comandos individuais na Tabela 8-8 até a Tabela 8-41, a coluna de tamanho descreve o número de bytes no parâmetro documentado em cada linha específica. Se o tamanho do bloco de entrada para um determinado comando estiver incorreto, o dispositivo responderá de forma diferente, dependendo do comando. Ele pode não retornar uma indicação de erro em todas as circunstâncias (consulte a seção [Status/Códigos de erro](#)).

8.5.2 Códigos de Operação de Comando, Descrições Breves e Tempos de Execução

Durante a análise dos parâmetros e a subsequente execução de um comando recebido corretamente, o dispositivo está ocupado e não pode responder às transições nos pinos. O intervalo durante o qual o dispositivo está ocupado varia dependendo do comando e seus valores de parâmetro, estado do dispositivo, condições ambientais e outros fatores de acordo com a tabela a seguir.

Na maioria dos casos, mas não em todos, os comandos com falha retornam com relativa rapidez, geralmente bem antes do tempo de execução típico.

Tabela 8-4. Códigos de operação de comando, descrições curtas e tempos de execução

Descrição do código de operação do comando			Tipo. Exec. Tempo(1), ms	máx. Executivo Tempo(2), ms
DeriveKey	0x1C	Derive um valor de chave de destino da chave de destino ou pai.	14	62
DevRev	0x30	Retorna as informações de revisão do dispositivo.	0,4	2
De novo	0x15	Gere um resumo de proteção de dados a partir de uma semente aleatória ou de entrada e uma chave.	11	43
HMAC	0x11	Calcule a resposta da chave e outros dados internos usando HMAC/ SHA-256.	27	69
CheckMac	0x28	Verifique um MAC calculado em outro dispositivo Microchip CryptoAuthentication.	12	38
Trancar	0x17	Impede outras modificações em uma zona do dispositivo.	5	24
MAC	0x08	Calcule a resposta da chave e outros dados internos usando SHA-256.	12	35
Nonce	0x16	Gere um número aleatório de 32 bytes e um armazenado internamente nonce.	22	60
Quebrar	0x01	Coloque seletivamente apenas um dispositivo em um barramento compartilhado no estado ocioso.	0,4	2
aleatório	0x1B	Gera um número aleatório.	11	50
Ler	0x02	Leia quatro bytes do dispositivo, com ou sem autenticação e criptografia.	0,4	4
BEBIDA	0x47	Calcula um resumo SHA256 para qualquer finalidade do sistema.	11	22
UpdateExtra 0x20		Atualize os bytes 84 ou 85 na zona de configuração após o bloqueio da zona de configuração.	8	12
Escrever	0x12	Escreva 4 ou 32 bytes no dispositivo, com ou sem autenticação e criptografia.	4	42

Observação:

- Os tempos de execução típicos são representativos da duração para executar o comando assumindo condições sem erro, configuração de modo mais rápido, sem ações internas opcionais, como teclas de uso limitado e condições ambientais favoráveis. Para obter o melhor desempenho, atrasse esse intervalo e, em seguida, inicie a pesquisa para determinar a conclusão real do comando.
- Os tempos máximos de execução são representativos da duração mais longa de um comando bem-sucedido execução com todos os modos e ações internas habilitadas sob estatísticas e ambientais estendidas condições. O tempo de execução pode se estender além desses valores em situações extremas.

8.5.3 Codificação de zona

O valor em Param1 tanto para o comando Read quanto para o comando Write controla qual zona o acesso de comando. Consulte a seção [Bloqueio de zona de configuração](#) para obter mais informações sobre o que controla os estados “bloqueado” e “desbloqueado” para cada zona. Todos os outros valores de zona são reservados e devem não ser usado.

Tabela 8-5. Codificação de Zona (Param1)

Zona Nome	Param1 Valor	Tamanho lido			Escrever
configuração	0	704 bits 88 bytes 3 slots	Sempre disponível.		Parcialmente, quando desbloqueado. Nunca quando bloqueado. Nunca criptografado.
OTP	1	512 bits 64 bytes 2 slots	Nunca quando desbloqueado. Sempre quando bloqueado, exceto modo não legado. Consulte a seção Zona programável de tempo único (OTP) .		Não permitido quando LockConfig está desbloqueado. Tudo gravável quando LockConfig está bloqueado e LockValue está desbloqueado. Controlado pelo modo OTP quando LockValue está bloqueado. Consulte a seção Programável uma vez (OTP) Zona .
Dados	2	4096 bits Nunca quando desbloqueado; 512 bytes caso contrário, controlado por 16 slots IsSecret e EncryptRead.			Não permitido quando LockConfig está desbloqueado. Tudo gravável quando LockConfig está bloqueado e LockValue está desbloqueado. Controlado por WriteConfig quando LockValue está bloqueado. Consulte a seção Bloqueio do dispositivo .

8.5.4 Codificação de Endereço

Param2 inclui um único endereço que indica a memória a ser acessada. Todas as leituras e gravações estão em unidades de palavras (4 bytes). O byte mais significativo de um endereço ATSHA204A válido é sempre zero. Todos os bits de endereço não utilizados devem sempre ser definidos como zero. Os bits menos significativos no endereço descrevem o deslocamento para a primeira palavra a ser acessada dentro do Bloco/Slot, enquanto os bits superiores especificam o número do Slot conforme tabela abaixo:

Tabela 8-6. Codificação de Endereço (Param2)

Zona	Byte 0 (primeiro byte no barramento)								Byte 1							
	7	6	5	4	3	2	1	0 7 6 5				4	3	2	1	0
Dados	0				Bloquear				Desvio			0	0	0	0	0
configuração	0	0	0		Bloquear				Desvio			0	0	0	0	0
OTP	0	0	0	0	Bloquear				Desvio			0	0	0	0	0

Dentro de cada zona, existem várias restrições de acesso conforme a tabela abaixo:

Tabela 8-7. Valores legais de bloco/slot

Zona	Bloco legal/slot (Até)	Notas
Dados	0 – 15	Todos os deslocamentos em todos os slots disponíveis para leitura e gravação. Acesso de 4 bytes permitido em um slot específico somente se SlotConfig.IsSecret for zero.
configuração	0 – 2	Palavras acima de 16 (bloco 2, deslocamento 6) nunca podem ser lidas.

Zona	Bloco legal/slot (Até)	Notas
		<p>Palavras acima de 10 (bloco 2, offset 0) lidas e escritas devem estar no modo Word (4 bytes).</p> <p>Palavras abaixo de 04 (bloco 0, offset 4) e acima de 15 (bloco 2, offset 5) nunca podem ser escritas.</p>
OTP	0 – 1	<p>Quando o modo OTP é somente leitura, todos os deslocamentos em ambos os blocos estão disponíveis para uso com leituras de 4 bytes e 32 bytes.</p> <p>Se o modo OTP for consumo, as gravações também serão permitidas em todos os deslocamentos.</p> <p>Consulte a seção Zona programável de tempo único (OTP) se o modo OTP for legado.</p>

8.5.5 Comando CheckMac

O comando CheckMac calcula uma resposta MAC que foi gerada em um dispositivo CryptoAuthentication e compara a resposta MAC com algum valor de entrada. Ele retorna um resultado booleano para indicar o sucesso ou falha da comparação.

Antes de executar este comando, os comandos Nonce e/ou GenDig podem ter sido executados opcionalmente para criar e carregar uma chave ou valor nonce em TempKey. O parâmetro mode determina a fonte da “chave” (os primeiros 32 bytes da mensagem SHA) e “challenge/nonce” (os segundos 32 bytes da mensagem SHA).

Mode<2> controla o requisito para um nonce aleatório se TempKey fizer parte do valor calculado. Se Mode<2> = 1, então TempKey deve ser gerado usando Nonce(Fixed); se Mode<2> = 0, então TempKey deve ser gerado usando Nonce(Random).

Definir Mode<2> como um pode permitir ataques de repetição em algumas situações.

Se a comparação corresponder, o valor do slot de destino poderá ser copiado para TempKey. Se SlotID for par, então o slot de destino é SlotD+1, ou então o slot de destino é SlotID. Para que a cópia ocorra, as seguintes condições devem ser verdadeiras. Se eles não forem todos verdadeiros, o ATSHA204A retornará o resultado da comparação, mas não copiará o valor da chave.

1. O parâmetro de modo para CheckMac deve ter um valor de 0x01 ou 0x05.
2. SlotConfig.ReadKey para a chave de destino deve ser zero.
3. O bit em Config.CheckMacSource correspondente aos slots de chave deve ter um valor que corresponda Modo <2>.

Tabela 8-8. Parâmetros de entrada

	Nome	Notas de tamanho
Opcode CheckMac 1 0x28		
Modo Param1	1	<p>Bits 7-6: Deve ser zero.</p> <p>8 bytes de mensagem SHA.</p> <p>Parte 5: 0: zeros</p> <p>1: zona OTP</p> <p>Bit 4-3: Defina como zero.</p> <p>Parte 2: Se TempKey for usado, esse bit deverá corresponder ao valor de TempKey.SourceFlag.</p> <p>Parte 1: Origem dos primeiros 32 bytes da mensagem SHA.</p>

	Nome	Notas de tamanho
		0: Slot<SlotID> 1: TempKey Bit 0: Origem dos segundos 32 bytes da mensagem SHA. 0: parâmetro ClientChal 1: TempKey
Param2 SlotID		2 Qual slot interno deve ser usado para gerar a resposta. Apenas os bits 3-0 são usados.
Data1		ClientChal 32 Desafio enviado ao Cliente. (Deve aparecer no fluxo de entrada).
		Data2 ClientResp 32 Resposta gerada pelo Cliente.
		Data3 OtherData 13 Dados constantes restantes necessários para o cálculo da resposta.

Tabela 8-9. Parâmetro de saída

Nome	Tamanho	Observações
Resultado 1		Retorna um valor de 1 byte de zero se ClientResp corresponder ao resumo calculado internamente, um se houver uma incompatibilidade.

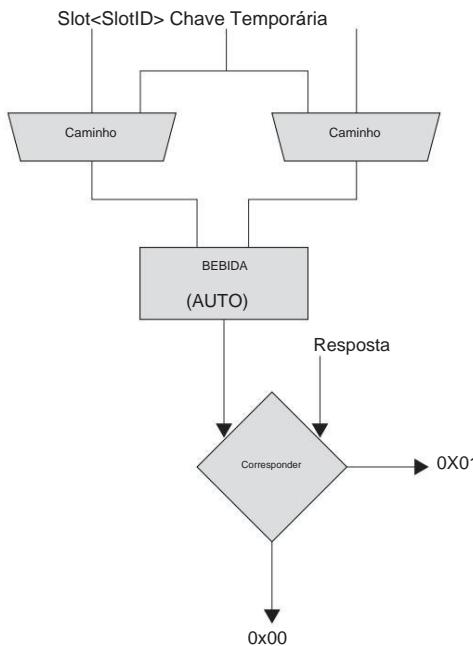
A mensagem que é hash com o algoritmo SHA-256 consiste nas seguintes informações:

32 bytes	key<SlotID> ou TempKey (dependendo do modo)
32 bytes	ClientChal ou TempKey (dependendo do modo)
4 bytes	OutrosDados<0:3>
8 bytes	OTP<0:7> ou zeros (dependendo do modo)
3 bytes	OutrosDados<4:6>
1 byte	SN<8>
4 bytes	OutrosDados<7:10>
2 bytes	SN<0:1>
2 bytes	OutrosDados<11:12>

O objetivo de OtherData é construir uma mensagem SHA-256 que corresponda identicamente à mensagem MAC que foi usada para produzir ClientResp. Ao comparar a mensagem usada para o SHA-256 do comando MAC, OtherData é analisado da seguinte forma:

Tabela 8-10. Outros dados

Size CheckMac	MAC	Notas
1 Outros Dados<0>	Código de operação	MAC OpCode = 0x08
1 Outros Dados<1>	Caminho	Modo usado para comando MAC.
2 Outros Dados<2:3>	LockID	SlotID usado para comando MAC.
3 Outros Dados<4:6>	OTP<8:10> OTP<8:10> usado para comando MAC. (Útil para legado.)	
4 Outros Dados<7:10>	SN<4:7>	SN<4:7> usado para comando MAC. (Único por cliente.)
2 Outros Dados<11:12>	SN <2:3>	SN<2:3> usado para comando MAC. (Único por cliente.)

Figura 8-1. Fluxo de dados para o comando CheckMac

8.5.6 Comando DeriveKey

O dispositivo combina o valor atual de uma chave com o nonce armazenado em TempKey usando SHA-256 e coloca o resultado no slot de chave de destino. SlotConfig<TargetKey>.Bit13 deve ser definido ou DeriveKey retorna um erro.

Se SlotConfig<TargetKey>.Bit12 for zero, a chave de origem combinada com TempKey é a chave de destino especificada na linha de comando (operação Roll-Key). Se SlotConfig<TargetKey>.Bit12 for um, a chave de origem é a chave pai da chave de destino, que é encontrada em SlotConfig<TargetKey>.WriteKey (operação Criar chave).

Antes da execução do comando DeriveKey, o comando Nonce deve ter sido executado para criar um nonce válido em TempKey. Dependendo do estado do bit dois do modo de entrada, este nonce teria sido criado com o RNG interno ou teria sido corrigido.

Se SlotConfig<TargetKey>.Bit15 for definido, um MAC de entrada deve estar presente e deve ser calculado da seguinte forma:

SHA-256(ParentKey, Opcode, Param1, Param2, SN<8>, SN<0:1>) onde o ID ParentKey é sempre SlotConfig<TargetKey>.WriteKey.

Se SlotConfig<TargetKey>.Bit12 ou SlotConfig<TargetKey>.Bit15 para definir e SlotConfig<ParentKey>.LimitedUse também for definido, DeriveKey retornará um erro se UseFlag<ParentKey> for 0x00. DeriveKey ignora LimitedUse e UseFlag para a chave de destino se SlotConfig<TargetKey>.Bit12 e SlotConfig<TargetKey>.Bit15 forem ambos zero.

Somente para slots de 0 a 7, se a análise de entrada e a verificação de MAC opcional forem bem-sucedidas, UseFlag<TargetKey> será definido como 0xFF e UpdateCount<TargetKey> será incrementado. Se UpdateCount tiver atualmente um valor de 0xFF, ele retornará a zero. Se o comando falhar por qualquer motivo, esses bytes não poderão ser atualizados. O valor de UpdateCount pode ser corrompido se a energia for interrompida durante a execução de DeriveKey.

Nota: Se as chaves de origem e destino forem as mesmas, existe o risco de perda permanente do valor da chave se a energia for interrompida durante a operação de gravação. Se os bits de configuração permitirem, então o slot de chave pode ser recuperado usando uma gravação autenticada e criptografada com base na chave pai.

Tabela 8-11. Parâmetros de entrada

	Nome	Notas de tamanho
Opcode DeriveKey		1 0x1C
Param1 Aleatório	1	<p>Bits 7-3: Deve ser zero.</p> <p>Parte 2: O valor desse bit deve corresponder ao valor em TempKey.SourceFlag ou o comando retornará um erro.</p> <p>Bits 1-0: Deve ser zero.</p>
Param2 TargetKey		2 Slot de chave a ser escrito.
Dados Mac		0 ou 32 MAC opcional usado para validar a operação.

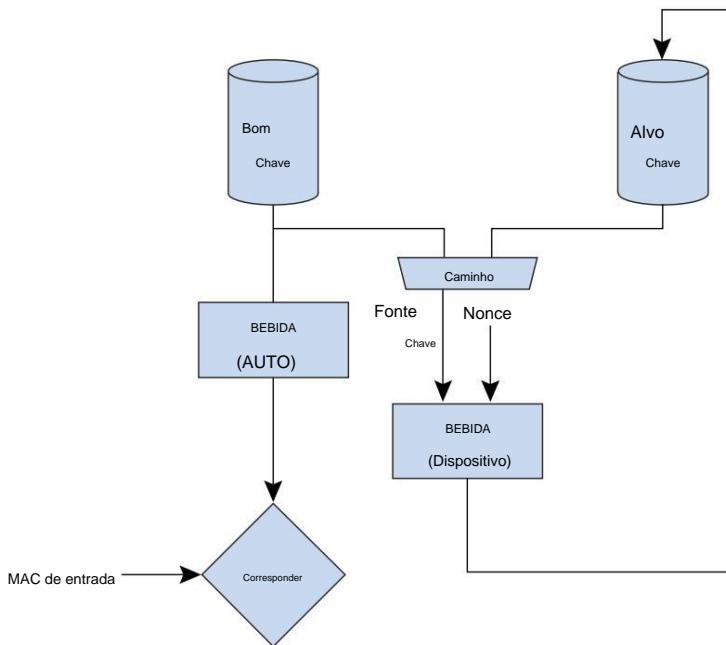
Tabela 8-12. Parâmetro de saída

Nome	Notas de tamanho
Sucesso	1 Após a conclusão bem-sucedida, o ATSHA204A retorna um valor de zero.

A chave gravada no slot de destino é o resultado de um SHA-256 da seguinte mensagem:

32 bytes	Alvo ou chave pai (dependendo do SlotConfig Bit12)
1 byte	Código de operação
1 byte	Param1
2 bytes	Param2
1 byte	SN<8>
2 bytes	SN<0:1>
25 bytes	zeros
32 bytes	TempKey.value

O fluxo de dados para este comando é mostrado graficamente na figura abaixo:

Figura 8-2. Fluxo de dados para o comando DeriveKey

8.5.7 Comando DevRev

O comando DevRev retorna uma única palavra de quatro bytes representando o número de revisão do dispositivo.

O software não deve depender desse valor porque ele pode mudar de tempos em tempos.

Tabela 8-13. Parâmetros de entrada

	Nome	Tamanho	Notas
Código de operação	DevRev	1	0x30.
Param1	Caminho	1	Deve ser zero.
Param2	—	2	Deve ser zero.
Dados	—	0	—

Tabela 8-14. Parâmetros de Saída

Nome	Tamanho	Notas
Sucesso	4	O número de revisão do dispositivo atual.

8.5.8 Comando GenDig

O comando GenDig usa SHA-256 para combinar um valor armazenado com o conteúdo de TempKey, que deve ser válido antes da execução desse comando. O valor armazenado pode vir de um dos slots de dados, das páginas OTP, das duas primeiras páginas da zona de configuração ou recuperado da matriz de chaves de transporte de hardware. O resumo resultante é retido em TempKey e pode ser usado de uma das três maneiras a seguir: Pode ser incluído como parte da mensagem usada pelos

1. comandos MAC, CheckMac ou HMAC.

Como a saída de resposta MAC incorpora os dados usados no cálculo do GenDig e a chave secreta do comando MAC, ela serve para autenticar os dados armazenados nas zonas de Dados e/ou OTP.

2. Um comando Read ou Write subsequente pode usar o resumo para fornecer autenticação e/ou confidencialidade para os dados, caso em que é conhecido como resumo de proteção de dados.
3. Este comando pode ser usado para personalização segura usando um valor da matriz de chave de transporte. O resumo de proteção de dados resultante seria então usado pelo comando de gravação.

Se a zona for dois (dados) e o SlotID for menor ou igual a 15, o comando GenDig define TempKey.GenData como um e TempKey.SlotID como o SlotID de entrada; caso contrário, TempKey.GenData é definido como zero.

Independentemente de como o resumo resultante é calculado, ele nunca pode ser lido no dispositivo.

Se TempKey.Valid for inválido, esse comando retornará um erro. Após a conclusão do comando, o bit TempKey.Valid é definido, indicando que um resumo foi carregado e está pronto para uso. O bit TempKey.Valid é limpo quando o próximo comando é executado. Consulte a seção [RAM estática \(SRAM\)](#) para obter detalhes.

Para todos os valores de SlotID menores que 0x8000, o dispositivo usa os quatro bits menos significativos de SlotID para determinar o número do slot do qual recuperar o valor da chave da zona de dados da EEPROM.

Valores de SlotID acima de 0x8000 chaves de referência armazenadas nas máscaras do projeto. Em qualquer caso, todos os 16 bits de SlotID como entrada para o dispositivo são usados como Param2 no cálculo SHA-256.

Se o parâmetro de zona apontar para a zona de configuração, este comando retornará um erro se a zona de configuração estiver desbloqueada.

Quando a chave especificada na entrada para GenDig tem o bit CheckOnly definido, GenDig pode ser usado para gerar chaves efêmeras correspondentes àquelas geradas nos dispositivos Client CryptoAuthentication usando o comando DeriveKey. As chaves que possuem o bit CheckOnly definido representam situações nas quais o dispositivo está agindo como um host. Nesse caso, os bytes de opcode e parâmetro que normalmente seriam incluídos no cálculo de SHA são substituídos por bytes do fluxo de entrada.

Tabela 8-15. Parâmetros de entrada

	Nome	Notas de tamanho
Opcode GenDig		1 0x15
área Param1	1	<p>Se 0x00 (Config), use SlotID para especificar o primeiro (SlotID=0) ou segundo (SlotID = 1) bloco de 256 bits da zona de configuração.</p> <p>Se 0x01 (OTP), use SlotID para especificar o primeiro ou o segundo bloco de 256 bits da zona OTP.</p> <p>Se 0x02 (dados), SlotID especifica um slot na zona de dados ou uma chave de transporte na matriz de hardware.</p> <p>Todos os outros valores são reservados e não devem ser usados.</p>
Param2 SlotID		2 Número de identificação da chave a ser utilizada, ou seleção de qual bloco OTP.
Dados	Outros Dados 4 ou 0	Quatro bytes de dados para cálculo de SHA ao usar uma chave CheckOnly; caso contrário ignorado.

Tabela 8-16. Parâmetro de Saída

	Nome	Notas de tamanho
Sucesso		1 Após a execução bem-sucedida, o ATSHA204A retorna um valor de zero.

Se a zona for Data e SlotConfig<SlotID>.CheckOnly for um, o corpo da mensagem SHA-256 usado para criar a nova TempKey resultante consiste nos seguintes bytes:

32 bytes	Slot<SlotID>
4 bytes	Outros dados
1 byte	SN<8>
2 bytes	SN<0:1>
25 bytes	zeros
32 bytes	TempKey.value

Em todos os outros casos, a mensagem usada para criar TempKey é a seguinte:

32 bytes	Config<SlotID> ou OTP<SlotID> ou Data.slot<SlotID> ou TransportKey<SlotID>
1 byte	Código de operação
1 byte	Param1
2 bytes	Param2
1 byte	SN<8>
2 bytes	SN<0:1>
25 bytes	zeros
32 bytes	TempKey.value

8.5.9 Comando HMAC

O comando HMAC calcula um resumo HMAC/SHA-256 de uma chave armazenada no dispositivo, um desafio e outras informações no dispositivo. A saída desse comando é a saída do algoritmo HMAC calculado sobre essa chave e mensagem. Se a mensagem incluir o número de série do dispositivo, a resposta é dita “diversificada”.

O fluxo de comando normal para usar este comando é o seguinte:

1. Execute o comando Nonce para carregar o desafio de entrada e, opcionalmente, combine-o com um número aleatório gerado. O resultado dessa operação é um nonce armazenado internamente no dispositivo.
2. Opcionalmente, execute o comando GenDig para combinar um ou mais locais de EEPROM armazenados no dispositivo com o nonce. O resultado é armazenado internamente no dispositivo.
3. Execute este comando HMAC para combinar a saída da etapa 1 (e da etapa 2, se desejado) com um Tecla EEPROM para gerar uma resposta de saída.

O passo 2 aborda os modelos de uso múltiplo. Se os dados na EEPROM forem uma chave, o GenDig tem o efeito de autenticar o desafio com várias chaves secretas. Alternativamente, se o conteúdo do slot for dados (que não precisam necessariamente ser secretos), o GenDig tem o efeito de autenticar o valor armazenado naquele local.

Tabela 8-17. Parâmetros de entrada

Nome	Tamanho	Observações
Opcode HMAC	1	0x11.
Param1	Modo 1	Controla quais campos dentro do dispositivo são usados na mensagem.
Param2	SlotID	2 Qual chave deve ser usada para gerar a resposta.

	Nome	Tamanho	Observações
			Param2<3:0> são usados apenas para selecionar um slot, mas todos os 16 bits são usados na mensagem HMAC.
Dados — 0 —			

Tabelas 8-18. Parâmetros de saída

Nome	Tamanho	Notas
Resposta	32	resumo HMAC

O resumo HMAC é calculado usando a chave em SlotID como a chave HMAC em uma mensagem que consiste nas seguintes informações:

32 bytes	zeros
32 bytes	TempKey
1 byte	Opcode (sempre 0x11)
1 byte	Caminho
2 bytes	LockID
8 bytes	OTP<0:7> ou zeros (Ver tabela abaixo)
3 bytes	OTP<8:10> ou zeros (Ver tabela abaixo)
1 byte	SN<8>
4 bytes	SN<4:7> ou zeros (Ver tabela abaixo)
2 bytes	SN<0:1>
2 bytes	SN<2:3> ou zeros (Ver tabela abaixo)

Consulte a Seção [HMAC/SHA-256](#) para obter os documentos reguladores que contêm uma descrição completa de como os vários resumos são calculados usando SHA-256, a chave HMAC e o preenchimento apropriado.

Tabela 8-19. Modo de Codificação

Significado dos Bits	
7	Deve ser zero.
6	0 = 48 bits correspondentes a SN<2:3> e SN<4:7> são definidos como zero. 1 = Inclua os 48 bits SN<2:3> e SN<4:7> na mensagem
5	0 = 64 bits correspondentes a OTP<0> a OTP<7> são definidos como 0. 1 = Inclua os primeiros 64 bits OTP (8 Bytes) OTP<0> a OTP<7> na mensagem. Se Mode<4> for definido, o valor desse bit de modo será ignorado.
4	0 = 88 bits (11 bytes) correspondentes a OTP<0> a OTP<10> são definidos como 0. 1 = Incluir os primeiros 88 bits OTP (11 Bytes) OTP<0> a OTP<10> na mensagem.
3	Deve ser zero.

Significado dos Bits	
2	O valor deste bit deve corresponder ao valor em TempKey.SourceFlag ou o comando retornará um erro.
1 – 0	Deve ser 0b00.

8.5.10 Comando de Bloqueio

Escreva LockConfig ou LockValue para 0x00, para alterar as permissões na zona designada.

Este comando falha se a zona designada já estiver bloqueada.

Antes de bloquear o dispositivo, o ATSHA204A usa o algoritmo CRC-16 para gerar um resumo da(s) zona(s) designada(s). O cálculo é feito de forma idêntica ao CRC calculado sobre os blocos de entrada e saída.

- **Zona de Configuração:** O CRC é calculado sobre todos os 88 bytes.

- **Zona de Dados e OTP:** Seus conteúdos são concatenados nessa ordem para criar a entrada para o CRC algoritmo.

Se o resumo de entrada não corresponder ao calculado no dispositivo, um erro é retornado e o processo de personalização deve ser repetido.

Tabela 8-20. Parâmetros de entrada

	Nome	Tamanho	Observações
Bloqueio do código de operação		1	0x17.
área Param1		1	<p>0 = CRC é verificado quando a zona está bloqueada.</p> <p>Parte 7: 1 = A verificação do CRC é ignorada e a zona é bloqueada, independentemente do estado da memória. A Microchip não recomenda a utilização deste modo.</p> <p>Bits 6-1: Todos os bits devem ser zero.</p> <p>Bit 0: 0 = Zona de configuração 1= Zona de dados e OTP</p>
Param2	Summary 2	Resumo	das zonas designadas, ou deve ser 0x0000 se Zone<7> for definido.
Dados -		0 —	

Tabela 8-21. Parâmetro de saída

Nome	Notas de tamanho
Sucesso	1 Após a execução bem-sucedida, o ATSHA204A retorna um valor de zero.

8.5.11 Comando MAC

O comando MAC calcula um resumo SHA-256 de uma chave armazenada no dispositivo, um desafio e outras informações no dispositivo. A saída desse comando é o resumo dessa mensagem. Se a mensagem incluir o número de série do dispositivo, a resposta é dita “diversificada”.

O fluxo de comando normal para usar este comando é o seguinte:

1. Execute o comando Nonce para carregar o desafio de entrada e, opcionalmente, combine-o com um número aleatório gerado. O resultado dessa operação é um nonce armazenado internamente no dispositivo em tempkey.

2. Opcionalmente, execute o comando GenDig para combinar um ou mais locais de EEPROM armazenados no dispositivo com o nonce. O resultado é armazenado internamente no dispositivo em tempkey. Esse recurso permite que duas ou mais chaves sejam usadas como parte da geração de resposta.
3. Execute este comando MAC para combinar a saída da Etapa 1 (e Etapa 2, se desejado) com uma chave EEPROM para gerar uma resposta de saída (ou resumo).

Tabela 8-22. Parâmetros de entrada

	Nome	Notas de tamanho	
Opcode MAC		1	0x08.
Modo Param1		1	Controla quais campos dentro do dispositivo são usados na mensagem.
Param2 SlotID		2	Qual chave interna deve ser usada para gerar a resposta. Os bits 3-0 são usados apenas para selecionar um slot, mas todos os 16 bits são usados na mensagem SHA-256.
Dados	Desafio	0 ou 32	Parte de entrada da mensagem a ser digerida, ignorada se Mode<0> for um.

Tabela 8-23. Parâmetro de saída

Nome	Tamanho	Notas
Resposta	32	Resumo SHA-256.

A mensagem que é hash com o algoritmo SHA-256 consiste nas seguintes informações:

32 bytes	key<SlotID> ou TempKey (consulte a tabela abaixo)
32 bytes	Desafio ou TempKey (veja a tabela abaixo)
1 byte	Opcode (Sempre 0x08.)
1 byte	Caminho
2 bytes	Param2
8 bytes	OTP<0:7> ou zeros (Ver tabela abaixo)
3 bytes	OTP<8:10> ou zeros (Ver tabela abaixo)
1 byte	SN<8>
4 bytes	SN<4:7> ou zeros (Ver tabela abaixo)
2 bytes	SN<0:1>
2 bytes	SN<2:3> ou zeros (Ver tabela abaixo)

Tabela 8-24. Modo de Codificação

Significado dos Bits	
7	Deve ser zero.
6	0 = Defina os bits correspondentes a SN<2:3> e SN<4:7> como 0.
	1= Inclua os 48 bits SN<2:3> e SN<4:7> na mensagem.
5	Se Mode<4> for definido, o valor desse bit de modo será ignorado.
	0 = Defina os 64 bits OTP correspondentes como 0.

Significado dos Bits	
	1 = Incluir os primeiros 64 bits OTP (OTP<0> a OTP<7>) na mensagem.
4	0 = Defina os 88 bits OTP correspondentes como 0. 1 = Incluir os primeiros 88 bits OTP (OTP<0> a OTP<10>) na mensagem.
3	Deve ser zero.
2	Se Mode<0> ou Mode<1> forem definidos, Mode<2> deverá corresponder ao valor em TempKey.SourceFlag ou o comando retornará um erro.
1	0 = Os primeiros 32 bytes da mensagem SHA são carregados de um dos slots de dados. 1 = Os primeiros 32 bytes são preenchidos com TempKey.
0	0 = Os segundos 32 bytes da mensagem SHA são obtidos do parâmetro Challenge de entrada. 1 = Os segundos 32 bytes são preenchidos com o valor em TempKey. Este modo é recomendado para todos os usos.

8.5.12 Comando Nonce

O comando Nonce gera um nonce para uso por um comando GenDig, MAC, HMAC, Read ou Write subsequente combinando um número aleatório gerado internamente com um valor de entrada do sistema. O nonce resultante é armazenado internamente em TempKey e o número aleatório gerado é retornado ao sistema.

O valor de entrada é projetado para evitar ataques de repetição contra o Host e deve ser gerado externamente pelo sistema e passado para o dispositivo usando este comando. Pode ser qualquer valor que mude consistentemente, como um contador não volátil, hora real do dia e assim por diante; ou pode ser um número aleatório gerado externamente.

Para fornecer um valor nonce para comandos criptográficos subsequentes, o número de entrada e o número aleatório de saída são agrupados de acordo com as informações listadas abaixo. O resumo resultante (nonce) é sempre armazenado no registro TempKey, TempKey.Valid é definido e TempKey.SourceFlag é definido como "Rand". O nonce pode ser usado por um comando GenDig, Read, Write, HMAC ou MAC subsequente, portanto, o sistema deve computar externamente esse valor resumido e armazená-lo externamente para concluir a execução desses comandos.

Como alternativa, esse comando também pode ser executado em um modo de passagem se um nonce fixo for necessário para comandos subsequentes. Nesse caso, o valor de entrada deve ter 32 bytes e é passado diretamente para TempKey sem modificação. Nenhum cálculo SHA-256 é executado e TempKey.SourceFlag é definido como "Input". O valor nonce em TempKey não pode ser usado com comandos Read ou Write. Se operado neste modo e com um valor de número de entrada repetido, o dispositivo não oferece proteção contra ataques de repetição.

Antes da seção de configuração ser bloqueada, o RNG produz um valor de 32 bytes de 0xFF FF 00 00 FF FF 00 00... para facilitar o teste. Este valor de teste é combinado com o valor de entrada da maneira descrita acima.

Tabela 8-25. Parâmetros de entrada

	Nome	Notas de tamanho	
OpcodeNonce		1	0x16.
Modo Param1		1	Controla o mecanismo do RNG interno e atualização de sementes.
Param2 Zero		2	Deve ser 0x0000.
Dados	NumIn	20,32	Valor de entrada do sistema.

Tabela 8-26. Parâmetro de Saída

Nome	Notas de tamanho
RandOut 1 ou 32 A saída do RNG ou um único byte com valor zero se Mode<0:1> for três.	

Se Mode<1:0> for 0b00 ou 0b01, o parâmetro NumIn de entrada deve ter 20 bytes de comprimento e o corpo da mensagem SHA-256 usado para criar o nonce armazenado internamente em TempKey consiste no seguinte:

32 bytes	RandOut
20 bytes	NumIn do fluxo de entrada
1 byte	Opcode (sempre 0x16)
1 byte	Caminho
1 byte	LSb de Param2 (deve ser sempre 0x00)

Após a conclusão do comando, TempKey.SourceFlag é definido como "Rand".

Se Mode<1:0> for 0b11, este comando opera em modo de passagem, o parâmetro de entrada (NumIn) deve ter 32 bytes de comprimento e TempKey é carregado com NumIn. Nenhum cálculo SHA-256 é executado, nenhum dado é retornado ao sistema e TempKey.SourceFlag é definido como "Input".

Se Modo<1:0> for 0b01, a atualização automática de propagação é suprimida. Consulte a seção [Gerador de números aleatórios \(RNG\)](#) para obter mais detalhes. A Microchip recomenda que Mode<1:0> seja definido como 0b00 para maior segurança.

Tabela 8-27. Modo de Codificação

Significado dos Bits	
7 – 2	Deve ser zero.
1 – 0	<p>00 = Combine o novo número aleatório com NumIn, armazene em TempKey. Atualize automaticamente a semente EEPROM somente se necessário antes da geração de números aleatórios. Recomendado para maior segurança.</p> <p>01 = Combine o novo número aleatório com NumIn, armazene em TempKey. Gere um número aleatório usando a semente EEPROM existente, não atualize a semente EEPROM.</p> <p>10 = Status inválido</p> <p>11 = Operar no modo de passagem e escrever TempKey com NumIn. (Deve ter 32 bytes).</p>

8.5.13 Comando de pausa

Todos os dispositivos no barramento para os quais o byte do seletor de configuração não corresponde ao parâmetro do seletor de entrada entram no estado inativo. Este comando é usado para evitar conflitos de barramento em um sistema que inclui vários dispositivos ATSHA204A compartilhando o mesmo barramento.

O comando Pause difere do sinalizador/sequência ocioso porque dispositivos individuais no barramento de pino único podem ser selecionados para entrar no estado ocioso, em oposição ao sinalizador ocioso que faz com que todos os dispositivos CryptoAuthentication no barramento entrem no estado ocioso.

Se o byte do seletor de EEPROM não corresponder ao parâmetro do seletor de entrada, o dispositivo irá imediatamente para o estado inativo. Se o parâmetro do seletor de entrada corresponder ao byte do seletor de configuração, o dispositivo retornará um código de sucesso de 0x00.

O comando Pause não pode ser usado para colocar os dispositivos no estado de suspensão.

Tabela 8-28. Parâmetros de entrada

	Nome	Notas de tamanho	
Código de operação	Quebrar	1	0x01.
Param1	Seletor	1	Todos os dispositivos que não correspondem a esse valor vão para o estado inativo.
Param2	Zero	2	Deve ser 0x0000.
Dados —		0 —	

Tabela 8-29. Parâmetro de Saída

Nome	Tamanho	Observações
Sucesso 1		<p>Se o comando indicar que algum outro dispositivo deve ficar inativo, o ATSHA204A retorna um valor de 0x00.</p> <p>Se este dispositivo ficar ocioso, nenhum valor será retornado.</p>

8.5.14 Comando Aleatório

O comando Random gera um número aleatório para ser usado pelo sistema.

Números aleatórios são gerados por meio de uma combinação da saída de um RNG de hardware e um valor de semente interno armazenado na EEPROM ou SRAM. O sistema externo pode optar por atualizar o valor de semente EEPROM armazenado internamente antes da geração do número aleatório como parte da execução do comando Nonce ou Random, para maior segurança, a Microchip recomenda que a semente EEPROM seja sempre atualizada.

O comando aleatório não fornece um mecanismo para integrar um número de entrada com a semente armazenada interna.

Se essa funcionalidade for desejada, o sistema deve usar o comando Nonce e ignorar o nonce gerado.

Antes da seção de configuração ser bloqueada, o RNG produz um valor de 32 bytes de 0xFF, 0xFF, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00... para facilitar o teste.

As mesmas sementes armazenadas internamente são usadas para os comandos Nonce e Random. O uso do Modo<0> garante que a EEPROM seja atualizada, se necessário.

Tabela 8-30. Parâmetros de entrada

	Nome	Notas de tamanho	
Opcode aleatório		1	0x1B.
Modo Param1		1	Controla o mecanismo do RNG interno e atualização de sementes.

	Nome	Notas de tamanho
Param2 Zero		2 Deve ser 0x0000.
Dados —	0 —	

Tabela 8-31. Parâmetro de Saída

Nome	Tamanho	Notas
RandOut	32	A saída do RNG.

Tabela 8-32. Modo de Codificação

Significado dos Bits	
7 – 1	Deve ser zero.
0	<p>0 = Atualize automaticamente a semente EEPROM somente se necessário antes da geração de números aleatórios. Recomendado para maior segurança.</p> <p>1 = Gerar um número aleatório usando a semente EEPROM existente; não atualize a semente EEPROM.</p>

8.5.15 Comando de leitura

O comando Read lê palavras (uma palavra de 4 bytes ou um bloco de 8 palavras de 32 bytes) de uma das zonas de memória do dispositivo. Os dados podem opcionalmente ser criptografados antes de serem devolvidos ao sistema.

Consulte a Seção [Zona de dados EEPROM](#) para obter informações de endereçamento de palavra e byte da zona de dados.

Se estiver lendo de um slot no qual SlotConfig.EncryptRead está definido, o comando GenDig deve ter sido executado antes da execução desse comando para gerar a chave que é usada para criptografia. Se o número do slot for par ou se o bit CheckMacSource correspondente a esse slot for zero, o nonce de entrada para GenDig deve ter sido um número aleatório. Finalmente, a chave especificada em SlotConfig.ReadKey deve ter sido usada no cálculo do GenDig.

O dispositivo criptografa os dados a serem lidos por XORing de cada byte lido da EEPROM com o byte correspondente da TempKey. Leituras criptografadas das zonas de configuração e/ou OTP não são permitidas.

Os endereços de bytes a serem lidos devem ser divididos por quatro (elimine os dois bits menos significativos) antes de serem passados para o dispositivo. Se 32 bytes estiverem sendo lidos, os três bits menos significativos do endereço de entrada serão ignorados.

Endereços além do final da zona especificada resultam em erro.

As seguintes restrições se aplicam às três zonas a seguir:

- **Dados**

Se a zona de dados estiver desbloqueada, este comando retornará um erro; caso contrário, os valores dentro da palavra SlotConfig correspondente agem para controlar o acesso ao slot de dados. Se SlotConfig.IsSecret for definido e uma leitura de quatro bytes for tentada, o dispositivo retornará um erro. Se EncryptRead estiver definido, esse comando criptografará os dados conforme especificado acima. Se IsSecret estiver definido e EncryptRead estiver desmarcado, esse comando retornará um erro. Se IsSecret for clear e EncryptRead for clear, então este comando retornará o slot desejado em clear.

- **Configuração As**

palavras dentro desta zona são sempre legíveis usando este comando, independentemente do valor de LockConfig.

- OTP

Se a zona OTP estiver desbloqueada, esse comando retornará um erro. Uma vez bloqueado, se o modo OTP não estiver definido como legado, todas as palavras poderão ser lidas. Se o modo OTP for herdado, apenas leituras de quatro bytes serão permitidas e endereços de zero ou um retornarão um erro.

Tabela 8-33. Parâmetros de entrada

	Nome	Tamanho	Observações
Leitura do código operacional		1	0x02
área Param1	1		<p>0 = 4 bytes são lidos 1 = Parte 7: 32 bytes são lidos. Deve ser zero se estiver lendo da zona OTP no modo Legado.</p> <p>Bits 6-2: Todos os bits devem ser zero.</p> <p>Bits 1-0: Selecione entre Config, OTP ou Data. Consulte a Seção Codificação de zona.</p>
Param2 Address 2			Endereço da primeira palavra a ser lida dentro da zona. Veja a Seção Codificação de Endereço .
Dados -	0 —		

Tabela 8-34. Parâmetro de saída

Nome	Tamanho	Notas
Conteúdo	4 ou 32	O conteúdo do local de memória especificado.

8.5.15.1 Operações de leitura na zona de dados As

operações de leitura na zona de dados dependem do estado de IsSecret e EncryptRead de acordo com a seguinte tabela:

Tabela 8-35. Permissão de operação de leitura

IsSecret	Encrypt	Ler	Descrição
0	0		<p>As leituras de texto não criptografado são sempre permitidas neste slot.</p> <p>Os slots definidos para esse estado nunca devem ser usados como armazenamento de chaves.</p> <p>Podem ser lidos 4 ou 32 bytes de cada vez.</p>
0	1		Entrada. Nenhuma segurança é garantida para slots usando este código.
1	0		<p>As leituras nunca são permitidas neste slot.</p> <p>Os slots definidos para esse estado ainda podem ser usados para armazenamento de chaves.</p>
1	1		<p>As leituras desse slot são criptografadas usando o algoritmo de criptografia documentado na descrição do comando Read (consulte a seção Comando Read).</p> <p>A chave de criptografia está no slot especificado por ReadKey. Leituras e gravações de 4 bytes são proibidas.</p>

Se a leitura da zona de dados e o bit EncryptRead estiverem definidos na palavra SlotConfig correspondente, as seguintes ações serão executadas para criptografar os dados:

- Todos os bits do registro TempKey devem ser configurados corretamente da seguinte maneira, ou este comando retornará um erro:

```
TempKey.Valid == 1
```

```
TempKey.GenData == 1
```

```
TempKey.SlotID == SlotConfig.ReadKey
```

- Se o número do slot que está sendo lido for par, então TempKey.SourceFlag deve ser "RAND".
- Se o número do slot for ímpar, TempKey.SourceFlag deverá corresponder ao valor em Config.CheckMacSource correspondente ao slot.
- XOR os dados da zona de memória com TempKey. Retorne como "Conteúdo".

8.5.16 Comando SHA

O comando SHA calcula um resumo SHA-256 para uso geral pelo sistema. Qualquer comprimento de mensagem pode ser acomodado. O sistema é responsável por enviar os bytes de pad e length com o último bloco.

O cálculo de um resumo ocorre por meio das duas etapas a seguir: 1.

Inicialização

Configure o mecanismo de cálculo SHA-256 substituindo o valor atual de TempKey pela constante de inicialização.

Força os sinalizadores TempKey para corresponder ao estado que eles teriam após um comando Nonce(Fixed).

Este modo não aceita nenhum byte de mensagem.

2. Calcular O

comando pode ser chamado um número variável de vezes com este modo para adicionar bytes à mensagem.

Cada iteração deste modo deve incluir uma mensagem de 64 bytes. O buffer de saída sempre contém o resumo, que pode ser ignorado se desejado. O resumo também é carregado em TempKey.

O comando SHA(Init) deve ser executado antes que qualquer comando SHA(Compute) seja aceito. O sistema pode executar quantos comandos SHA (Compute) forem necessários para calcular o resumo desejado. Um erro é retornado se qualquer comando diferente de SHA for executado entre a iteração "Init" e a última iteração "Compute". O comando também retorna um erro de análise se o byte de modo tiver um valor diferente de 0x00 ou 0x01.

O resumo intermediário armazenado em TempKey é invalidado se o dispositivo for colocado em hibernação ou se o timer do watchdog expirar. O software do sistema deve garantir que toda a mensagem seja enviada ao dispositivo durante um único intervalo de ativação/vigilância ou que as sequências ociosas apropriadas sejam inseridas entre os comandos SHA.

Tabela 8-36. Parâmetros de entrada

	Nome	Tamanho	Observações
Opcode SHA		1	0x47
Modo Param1		1	<p>Bits 7-1: Deve ser zero.</p> <p>Bit 0:</p> <p>0 = (Init): Carrega TempKey com o valor de inicialização para SHA-256. Nenhum byte de mensagem é aceito (o comprimento deve ser zero).</p> <p>1 = (Compute): Adicione 64 bytes no parâmetro de mensagem ao contexto SHA e retorne o resumo</p>
Param2 Param2		2	Deve ser 0x0000.
Mensagem de dados		0 ou 64	bytes de dados a serem incluídos na operação de hash. Ignorado se Mode<0> for zero.

Tabela 8-37. Parâmetro de saída

Nome	Notas de tamanho
Resposta 1 ou 32 O resumo SHA-256 se Mode<0> = 1; caso contrário, 0x00 para sucesso ou um código de erro.	

8.5.17 Comando UpdateExtra O comando

UpdateExtra é usado para atualizar os valores dos dois bytes extras dentro da zona de configuração (localização 84 e 85) após o bloqueio da zona de configuração. Também pode ser usado para diminuir rapidamente os contadores de uso anexados a uma chave quando apropriado.

Se Mode<1> estiver definido, o comando implementa um decremento rápido dos contadores de uso limitado que podem estar associados a uma tecla específica.

Se o slot indicado pelo parâmetro "newValue" não contém uma chave para a qual o uso limitado é implementado ou habilitado, o comando retorna silenciosamente sem executar nenhuma ação. Se o slot indicado contiver uma chave de uso limitado que não tenha nenhum uso restante, o comando retornará um erro; caso contrário, um dos bits de uso restantes é limpo. O comando não modifica Config.UpdateCount para o slot em questão.

Se o parâmetro mode indicar UserExtra no endereço 84:

- Se o valor atual em UserExtra (byte 84 da zona de configuração) for zero, UpdateExtra grava esse byte com o byte LS de newValue e retorna com sucesso.
- Se o valor atual em UserExtra for diferente de zero, o comando retornará um erro de execução.

Se o parâmetro de modo indicar seletor no endereço 85: Se

- SelectorMode (byte 19 da zona de configuração) for diferente de zero e Selector (byte 85 da zona de configuração) for zero, então este comando escreve Selector com o LSB de newValue e retorna sucesso. Uma vez gravado em um valor diferente de zero, ele é bloqueado contra atualizações adicionais.
- Se SelectorMode tiver um valor zero, indicando que nenhuma verificação do seletor atual deve ser feita, esse comando sempre atualiza o seletor e sempre é bem-sucedido.

Tabela 8-38. Parâmetros de entrada

	Nome	Notas de tamanho
Opcode UpdateExtra 1 0x20.		
Modo Param1	1	<p>Bits 7 – 2: Deve ser zero.</p> <p>Parte 1: 0 = Atualizar o byte de configuração 84 ou 85 1 = Ignora o bit 0 e decremente o contador de uso limitado associado à chave sem slot "newValue"</p> <p>Bit 0: Se zero, atualize o byte de configuração 84. Se houver, atualize o byte de configuração 85.</p>
Param2 NovoValor	2	<p>LSB: Valor a ser opcionalmente gravado na localização 84 ou 85 na zona de configuração.</p> <p>MSB: Deve ser 0x00.</p>
Dados -	0 —	

Tabela 8-39. Parâmetro de saída

Nome	Tamanho	Observações
Sucesso 1		Se o byte de memória foi atualizado, este comando retorna um valor de 0x00; caso contrário, ele retornará um erro de execução.

8.5.18 Comando de Gravação

O comando Write grava uma palavra de 4 bytes ou um bloco de 8 palavras de 32 bytes em uma das zonas EEPROM no dispositivo.

Dependendo do valor do byte WriteConfig para este slot, pode ser necessário que os dados sejam criptografados pelo sistema antes de serem enviados ao dispositivo.

As seguintes restrições se aplicam a gravações em zonas usando este comando:

- **Zona de dados:** Se a zona de configuração estiver bloqueada e a zona de dados estiver desbloqueada, todos os bytes em todas as zonas poderão ser gravados com texto simples ou dados criptografados usando gravações de 32 bytes. Depois que a zona de dados é bloqueada, os valores nos bytes WriteConfig controlam o acesso aos slots de dados. Se os bits WriteConfig para este slot forem definidos como “always”, os dados de entrada devem ser passados para o dispositivo sem problemas. Se SlotConfig<14> for definido como um, os dados de entrada devem ser criptografados e um MAC de entrada calculado.
- **Zona de configuração:** Se a zona de configuração estiver bloqueada ou Zone<6> estiver definida, este comando retornará um erro; caso contrário, os bytes são gravados conforme solicitado. Qualquer tentativa de gravar qualquer byte para o qual as gravações são permanentemente proibidas (conforme a Seção [EEPROM Data Zone](#)) resulta em um erro de comando sem modificações na EEPROM.
- **Zona OTP:** Se a zona OTP estiver desbloqueada, todos os bytes podem ser gravados com este comando. Se o A zona OTP está bloqueada e o byte OTPmode é somente leitura ou legado, então este comando retorna um erro; caso contrário, o modo OTP deve ser consumo e este comando zera os bits na zona OTP que correspondem aos bits zero no valor do parâmetro de entrada. Quando a zona OTP está bloqueada, as gravações criptografadas nela nunca são permitidas, independentemente do Modo OTP.

As gravações de quatro bytes só são permitidas nas zonas de dados e OTP se todas as quatro condições a seguir forem atendidas:

- SlotConfig.IsSecret deve ser zero.
- SlotConfig.WriteConfig deve ser “sempre”.
- Os dados de entrada não devem ser criptografados.
- As zonas de Dados/OTP devem ser bloqueadas.

As gravações de quatro bytes retornam um erro em todas as outras circunstâncias.

Os três bits menos significativos de Param2, Endereço<2:0>, indicam a palavra dentro do bloco ou são ignorados se um bloco inteiro de 32 bytes estiver sendo gravado. Endereço<6:3> contém o número do slot para gravações na zona de dados ou o número do bloco para as zonas de configuração e OTP. Os valores de endereço além do tamanho da zona especificada resultam no retorno de erro do comando.

Qualquer tentativa de gravar as zonas OTP e/ou Dados antes do bloqueio da seção de configuração resulta no retorno do dispositivo a um código de erro.

8.5.18.1 Criptografia de dados de entrada

Os dados de entrada podem ser criptografados para evitar espionagem no barramento durante a personalização ou operação do sistema. O sistema deve criptografar os dados fazendo XORing no texto simples com o valor atual em TempKey. Após o recebimento, o dispositivo fará o XOR dos dados de entrada com TempKey para restaurar o texto simples antes de gravar na EEPROM.

Sempre que os dados de entrada são criptografados, um MAC de entrada de autorização é sempre necessário ao gravar a zona de dados. Este MAC é calculado da seguinte forma:

SHA-256 (TempKey, Opcode, Param1, Param2, SN<8>, SN<0:1>, <25 bytes de 0's>, PlainTextData)

Antes do bloqueio das zonas OTP/Data, Zone<6> é usado para indicar ao dispositivo se os dados de entrada são criptografados ou não. Após o bloqueio das zonas OTP/Data, Zone<6> é ignorado e apenas SlotConfig<14> correspondente ao slot que está sendo gravado é usado para determinar se os dados de entrada são criptografados ou não.

Se a criptografia de dados for indicada, então TempKey deve ser válido antes de este comando ser chamado e deve ser o resultado de GenDig. Especificamente, isso significa que TempKey.Valid e TempKey.GenDig devem ser definidos como um.

Antes do bloqueio de dados, qualquer chave pode ser usada para gerar TempKey. Após o bloqueio, o último slot usado pelo GenDig para criação de TempKey e armazenado em TempKey.SlotID deve corresponder ao SlotConfig.WriteKey. Se o número do slot sendo gravado for par, TempKey.SourceFlag deverá ser RAND.

Se o número do slot for ímpar, TempKey.SourceFlag deverá corresponder ao valor em Config.CheckMacSource correspondente ao slot.

Tabela 8-40. Parâmetros de entrada

		Nome Tamanho Observações
Gravação de código operacional		1 0x12
área Param1	1	<p>Parte 7:</p> <p>0 = 4 bytes de data são gravados na zona especificada.</p> <p>1 = 32 bytes de dados são gravados na zona especificada.</p> <p>0 = Os dados são escritos em claro.</p> <p>Parte 6:</p> <p>1 = Os dados de entrada devem ser criptografados.</p> <p>Deve ser zero se as zonas de Dados/OTP estiverem bloqueadas.</p> <p>Bits 5-2: Deve ser zero.</p> <p>Bits 1-0: Selecione entre Config, OTP ou Data. Consulte a Seção Codificação de zona.</p>
Param2 Address 2		Endereço da primeira palavra a ser escrita dentro da zona. Veja a Seção Codificação de Endereço .
Valor data_1		4 ou 32 Informações a serem escritas na zona; pode ser criptografado.
Dados_2 Mac		0 ou 32 Código de autenticação de mensagem para validar endereço e dados.

Tabela 8-41. Parâmetro de saída

Nome	Notas de tamanho
Sucesso	1 Após a conclusão bem-sucedida, o ATSHA204A retorna um valor de 0x00.

9.**Compatibilidade**

O ATSHA204A foi projetado para ser totalmente compatível com o ATSHA204 para todas as operações de host, cliente e personalização. Observe os seguintes refinamentos importantes que foram feitos no ATSHA204A:

- O consumo de energia ativa é menor. • O modo de conexão de dois fios é suportado sem a necessidade de um diodo externo. • O novo comando SHA permite o cálculo geral de um resumo SHA sem a necessidade de software criptográfico no Host.
- As operações de gravação durante a personalização sempre exigem que um MAC seja passado para o dispositivo para prevenir ataques man-in-the-middle. (Algumas versões do ATSHA204 ignoraram o MAC na gravação quando a zona de dados foi desbloqueada.)
- O comando UpdateExtra agora pode ser usado para diminuir rapidamente os contadores de uso limitado quando uma contagem de várias etapas é necessária.
- O modo Copiar do comando CheckMac agora pode ser executado com um nonce fixo que simplifica o implementação de validação de inicialização segura protegida e outras tarefas relacionadas.
- O novo modo de consumo na zona OTP fornece capacidade adicional para rastreamento de uso. • As operações de gravação na zona OTP ou de dados requerem 32 bytes quando a zona de configuração está bloqueada e as zonas OTP e de dados estão desbloqueadas. O ATSHA204 permitia comandos de gravação de 4 bytes para este estado de bloqueio.

10. Mecânica

10.1 Pinagem

O dispositivo é oferecido em vários pacotes: • UDFN de 8

almofadas

- SOT23 de 3 derivações
- SOC de 8 derivações
- TSSOP de 8 derivações ([Nota](#))
- CONTATO de 3 condutores destinado a conexão mecânica, não soldada.

As pinagens são as seguintes:

Tabela 10-1. Pinagem do pacote

Nomeie	SOT23 de 3 derivações	SOIC de 8 derivações, <u>TSSOP de 8 derivações</u> (Nota) e UDFN de 8 pads	CONTATO de 3 derivações
SDA	1	5	1
SCL	—	6	—
VCC	2	8	3
GND	3	4	2
NC	—	1, 2, 3, 7	—

Nota: Não recomendado para novos designs.

11. Informações de Marcação da Embalagem

Como parte dos recursos gerais de segurança da Microchip, a marcação parcial para todos os dispositivos criptográficos é intencionalmente vaga.

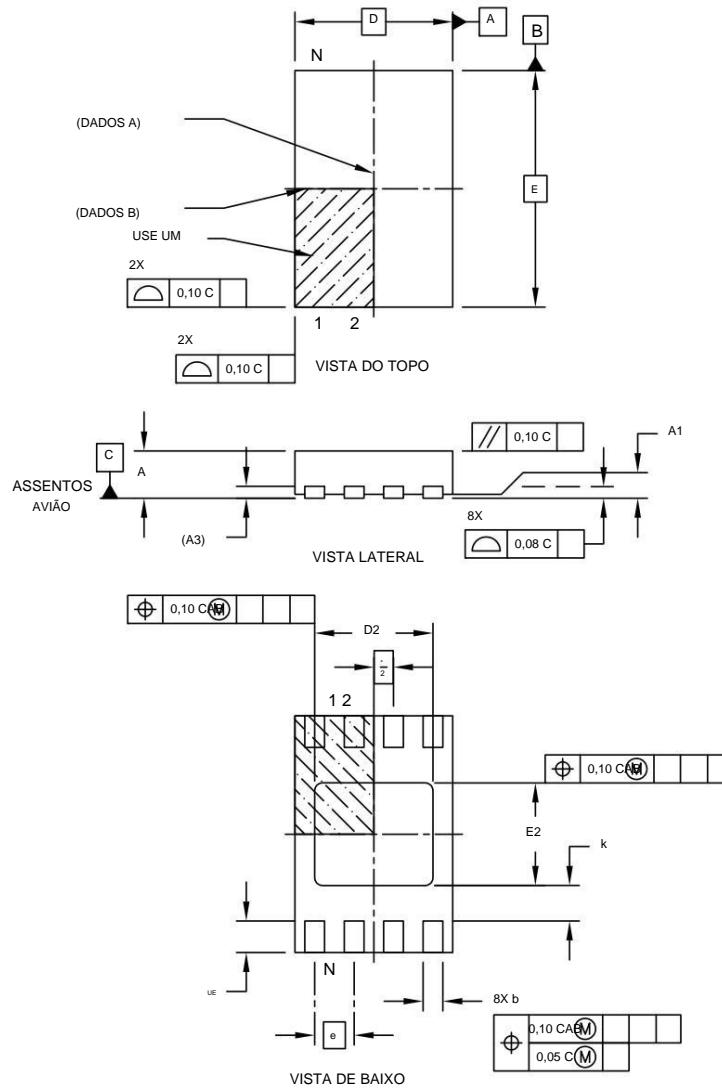
A marcação na parte superior da embalagem não fornece nenhuma informação sobre o tipo de dispositivo real ou o fabricante do dispositivo. O código alfanumérico na embalagem fornece informações de fabricação e varia de acordo com o lote de montagem. A marca da embalagem não deve ser usada como parte de qualquer procedimento de inspeção de entrada.

12. Desenhos de Embalagens

12.1 UDFN os 8 blocos

**Plástico ultrafino de 8 chumbos plano duplo, sem pacote de chumbo (Q4B) - Corpo de 2x3 mm [UDFN]
Pacote Atmel Legacy YNZ**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>

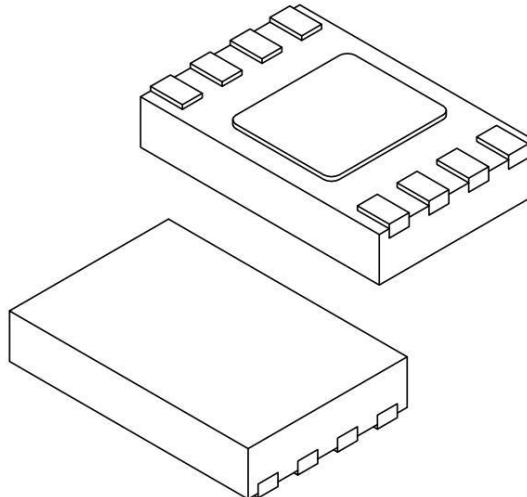


Desenho de tecnologia de microchip C04-21355-Q4B Rev A Folha 1 de 2

© 2017 Microchip Technology Inc.

**Plástico ultrafino de 8 chumbos plano duplo, sem pacote de chumbo (Q4B) - Corpo de 2x3 mm [UDFN]
Pacote Atmel Legacy YNZ**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



		Unidades			MILÍMETROS NOM		
		Limites de dimensão		MIN	8 0,50	MAX	
Número de terminais	N						
Tom	e				BSC 0,55 0,02		
Altura Geral	A	0,50			0,60		
impasse	A1	0,00			0,05		
Espessura terminal	A3			0,152	REF		
Comprimento total	D			2,00	BSC		
Comprimento da almofada exposta	D2	1,40		1,50		1,60	
Largura total	E			3,00	BSC		
Largura da almofada exposta	E2b	1,20		1,30		1,40	
Largura do Terminal		0,18		0,25		0,30	
Comprimento do Terminal	ue	0,35		0,40		0,45	
Terminal-para-Pad Exposto	k	0,20		-		-	

Notas:

1. O recurso de índice visual do pino 1 pode variar, mas deve estar localizado dentro da área hachurada.
2. O pacote é serrado individualmente
3. Dimensionamento e tolerância conforme ASME Y14.5M

BSC: Dimensão Básica. Teoricamente valor exato mostrado, sem tolerâncias.

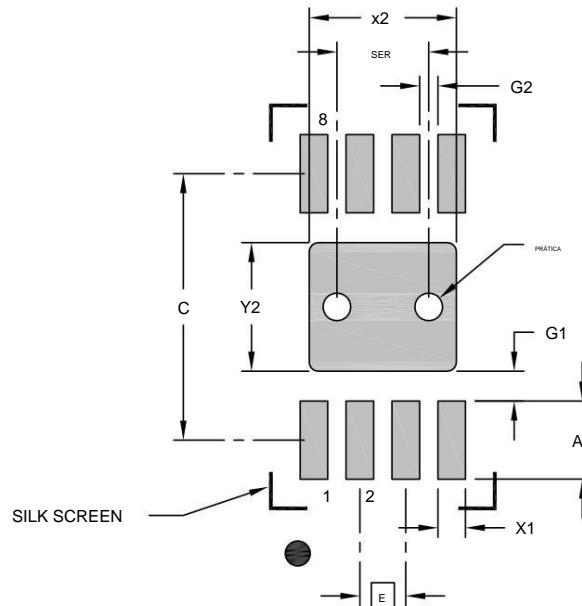
REF: Dimensão de referência, geralmente sem tolerância, apenas para fins informativos.

Desenho de tecnologia de microchip C04-21355-Q4B Rev A Folha 2 de 2

© 2017 Microchip Technology Inc.

**Plástico ultrafino de 8 chumbos plano duplo, sem pacote de chumbo (Q4B) - Corpo de 2x3 mm [UDFN]
Pacote Atmel Legacy YNZ**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



PADRÃO DE TERRENO RECOMENDADO

Unidades	MILÍMETROS NOM		
	MIN	0,50	MAX
Proposta de contato	E	BSC	
Largura da almofada central opcional	x2		1,60
Comprimento da almofada central opcional	Y2		1,40
Espaçamento entre almofadas de contato	C	2,90	
Largura da almofada de contato (X8)	X1		0,30
Comprimento da almofada de contato (X8)	A1		0,65
Bloco de contato para bloco central (X8)	G1	0,20	
Bloco de contato para bloco de contato (X6) G2		0,33	
Diâmetro Vía Térmica	EM		0,30
Passo Vía Térmica	SER	1,00	

Notas:

- Dimensionamento e tolerância conforme ASME Y14.5M
BSC: Dimensão Básica. Teoricamente valor exato mostrado, sem tolerâncias.
- Para melhores resultados de soldagem, as vias térmicas, se usadas, devem ser preenchidas ou protegidas para evitar perda de solda durante o processo de refluxo

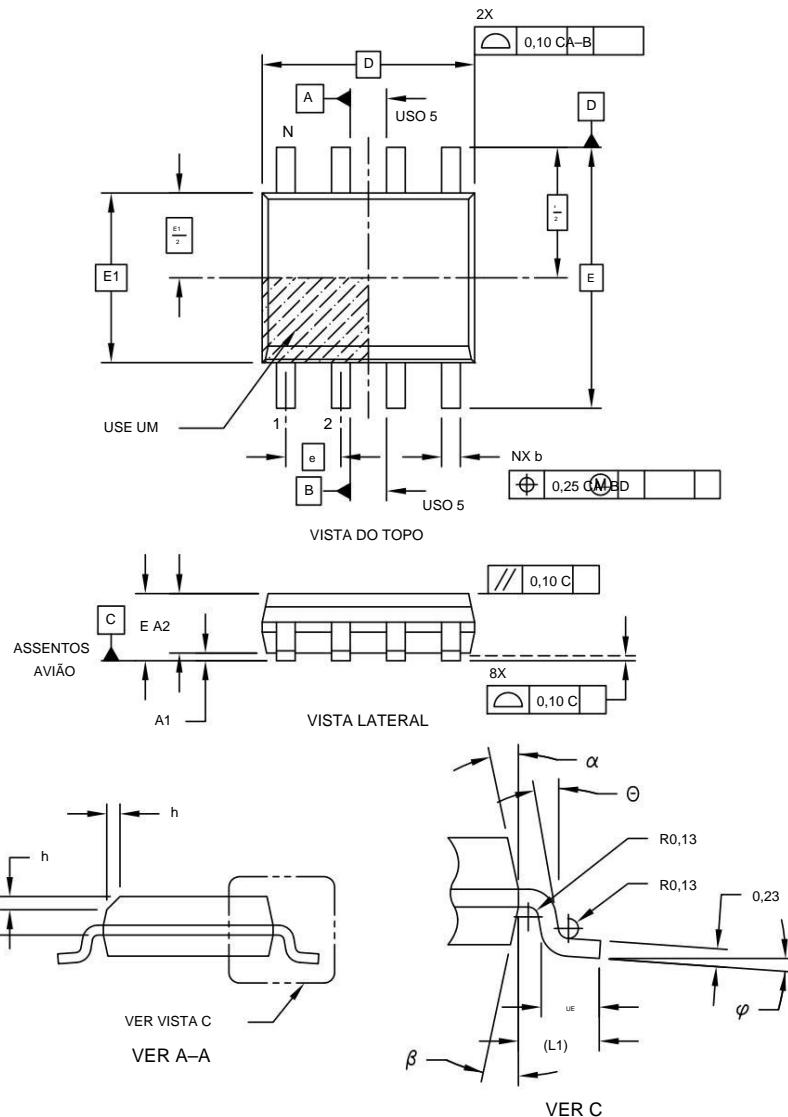
Desenho de tecnologia de microchip C04-21355-Q4B Rev A

© 2017 Microchip Technology Inc.

12.2 SOC de 8 derivações

**Contorno pequeno de plástico de 8 chumbos - Estreito, corpo de 3,90 mm (0,150 pol.) [SOIC]
Legado Atmel**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>

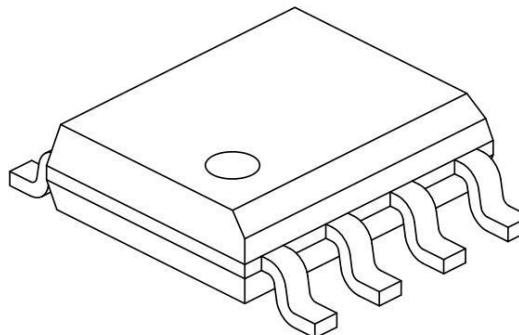


Desenho de tecnologia de microchip nº C04-057-Atmel Rev D Folha 1 de 2

© 2017 Microchip Technology Inc.

**Contorno pequeno de plástico de 8 chumbos - Estreito, corpo de 3,90 mm (0,150 pol.) [SOIC]
Legado Atmel**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



Unidades	MILÍMETROS NOM		
	MIN	8.127	MAX
Limites de dimensão			
número de pinheiros	N		
Tom	e	BSC	
Altura Geral	A - A2	1.25 0,10	1.75
Espessura da Embalagem Moldada		-	-
Impasse S	A1	-	0,25
Largura total	E	6,00 BSC	
Largura da Embalagem Moldada	E1	3,90 BSC	
Comprimento total	D	4,90 BSC	
Chamfrê (opcional)	h	0,25	-
Comprimento do pé	u	0,40	1,27
Pegada	L1	1.04 REF	
Ângulo do pé	φ	0°	-
Espessura do chumbo	c	0,17	-
Largura da guia	b	0,31	-
Topo do Ângulo de inclinação do molde	α	5°	-
Fundo do Ângulo de Saída do Molde	β	5°	15°

Notas:

1. O recurso de índice visual do pino 1 pode variar, mas deve estar localizado dentro da área hachurada. 2. § Característica significativa 3. As dimensões D e E1 não incluem rebarbas ou saliências. Rebarbas ou saliências do molde não devem exceder 0,15 mm por lado.

4. Dimensionamento e tolerância conforme ASME Y14.5M BSC:

Dimensão básica. Teoricamente valor exato mostrado, sem tolerâncias.

REF: Dimensão de referência, geralmente sem tolerância, apenas para fins informativos.

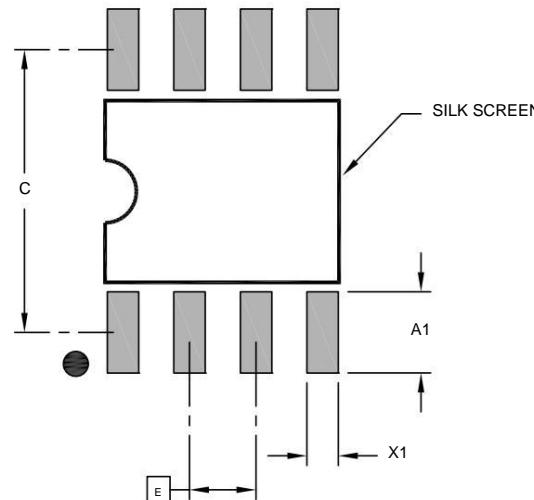
5. Datums A & B a serem determinados no Datum H.

Desenho de Tecnologia de Microchip Nº C04-057-OA Rev D Folha 2 de 2

© 2017 Microchip Technology Inc.

**Contorno pequeno de plástico de 8 chumbos - Estreito, corpo de 3,90 mm (0,150 pol.) [SOIC]
Legado Atmel**

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



PADRÃO DE TERRENO RECOMENDADO

Unidades	MILÍMETROS NOM		
	MIN	1,27	MAX
Proposta de contato E	BSC 5,40		
Espaçamento entre almofadas de contato C			
Largura da almofada de contato (X1)	X1	0,60	
Comprimento da almofada de contato (X8)	A1		1,55

Notas:

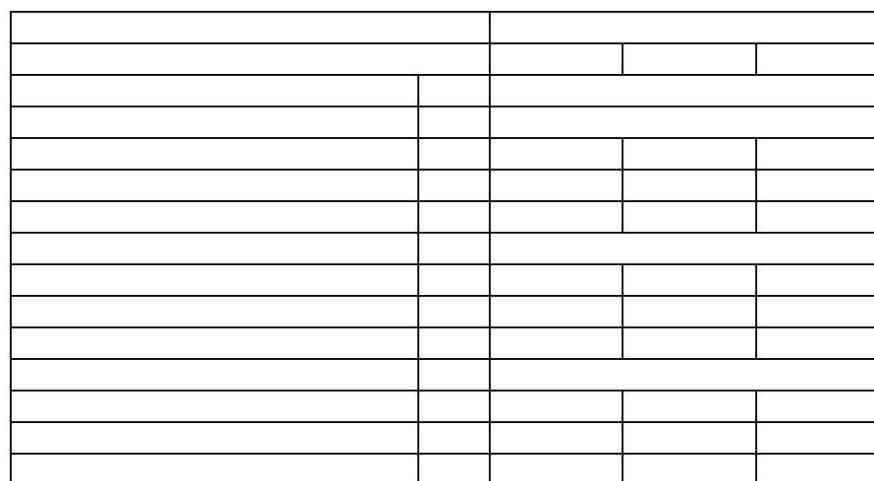
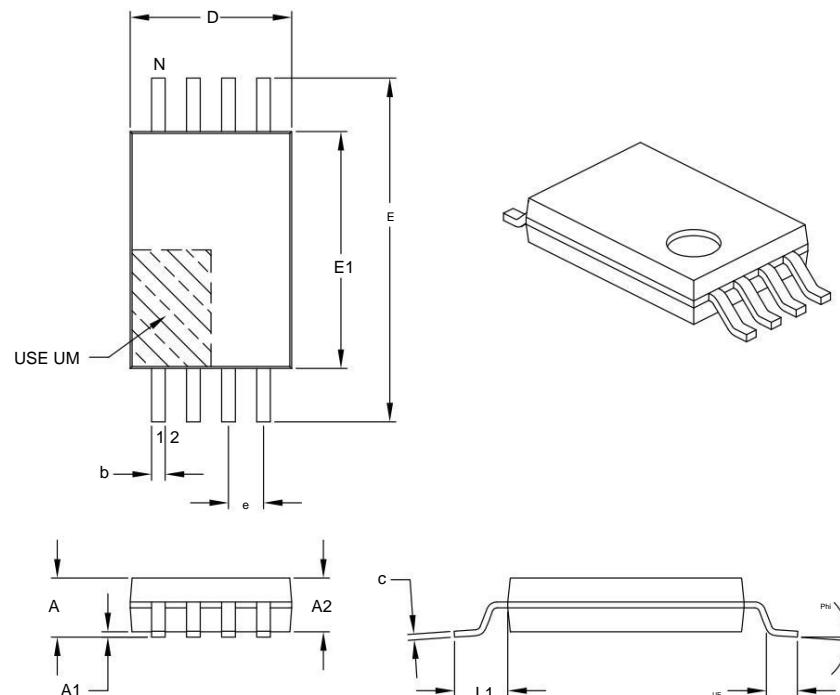
1. Dimensionamento e tolerância conforme ASME Y14.5M

BSC: Dimensão básica. Teoricamente valor exato mostrado, sem tolerâncias.

Desenho de tecnologia de microchip C04-2057-M6B Rev B

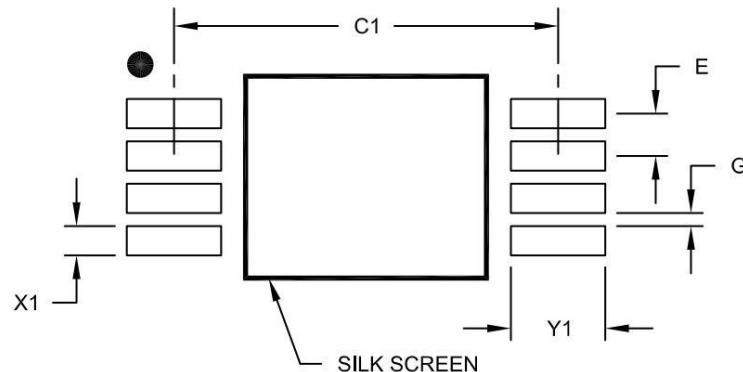
© 2017 Microchip Technology Inc.

12.3 TSSOP de 8 derivações



8-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

Nota: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension	Limits	UNITS MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Contact Pad Spacing	C1		5.90	
Contact Pad Width (X8)	X1			0.45
Contact Pad Length (X8)	Y1			1.45
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

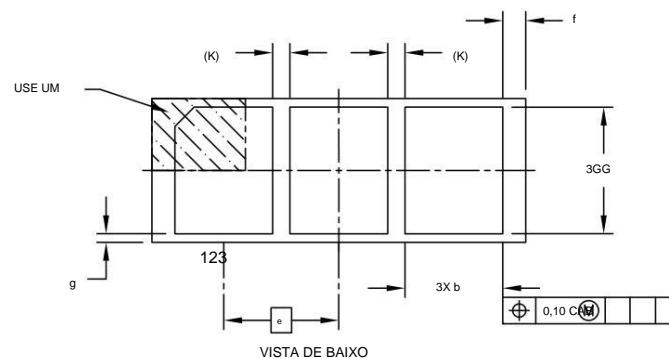
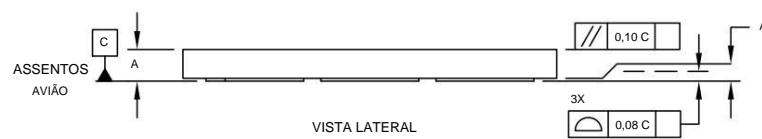
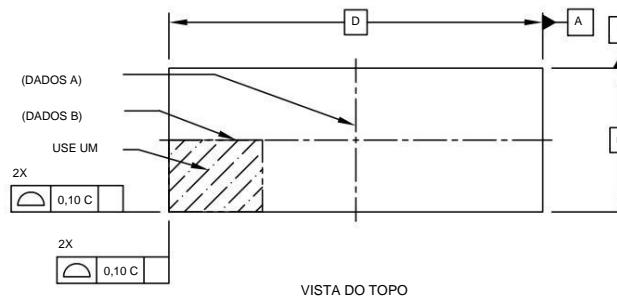
Microchip Technology Drawing No. C04-2086A

12.4 3 Contato principal

Pacote de contato de 3 derivações (LAB) - Corpo de 6,54x2,5 mm [Contato]

Código de pacote global herdado da Atmel RHB

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>

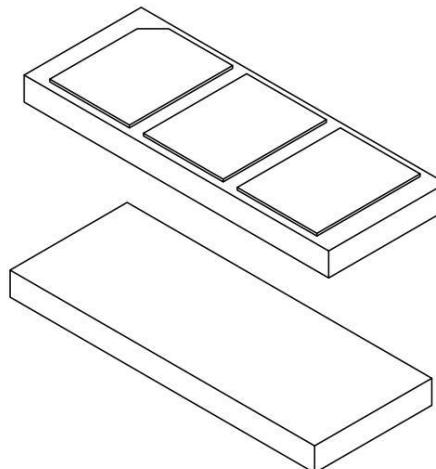


Desenho de tecnologia de microchip C04-21303 Rev A Folha 1 de 2

© 2017 Microchip Technology Inc.

Pacote de contato de 3 derivações (LAB) - Corpo de 6,54x2,5 mm [Contato]
Código de pacote global herdado da Atmel RHB

Observação: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



Unidades	MILÍMETROS		
	MIN	NOME	MAX
Limites de dimensão			
Número de terminais	N	3	
Tom	e	2,00 BSC	
Altura Geral	A	0,45	0,50
Impasse	A1	0,00	0,02
Comprimento total	D	6,50 BSC	
Largura total	E	2,50 BSC	
Largura do Terminal	b	1,60	1,70
Comprimento do Terminal	ue	2,10	2,20
Espacamento terminal a terminal	k	0,30 REF	
Borda do Pacote à Borda do Terminal f		0,30	0,40
Borda do Pacote à Borda do Terminal g		0,05	0,15
		0,15	0,25

Notas:

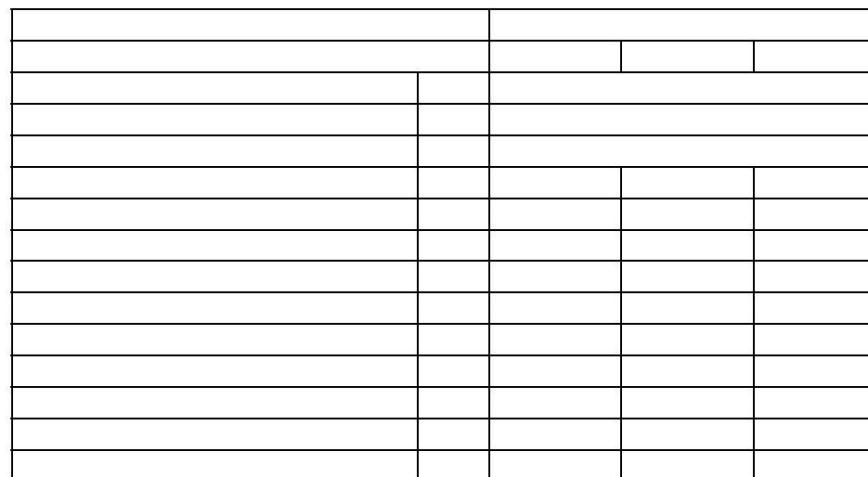
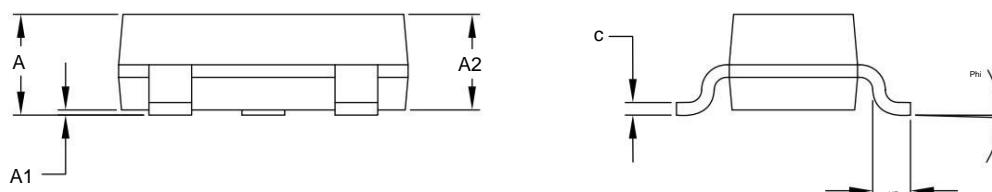
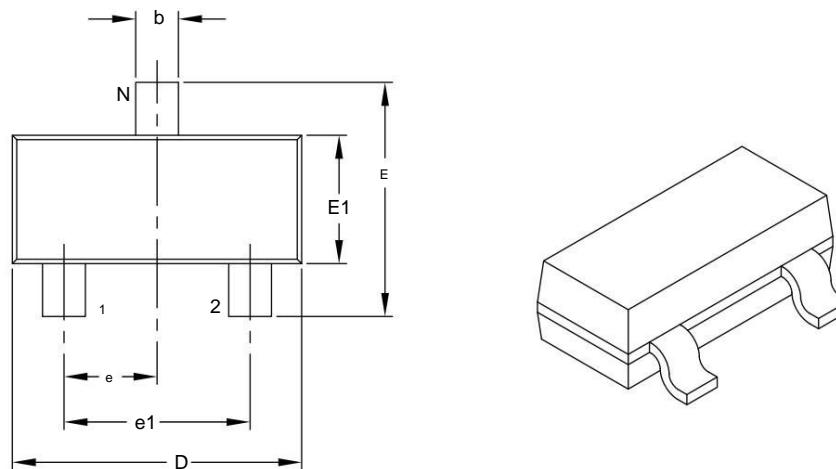
1. O recurso de índice visual do pino 1 pode variar, mas deve estar localizado dentro da área hachurada.
2. Dimensionamento e tolerância conforme ASME Y14.5M

BSC: Dimensão Básica. Teoricamente valor exato mostrado, sem tolerâncias.

REF: Dimensão de referência, geralmente sem tolerância, apenas para fins informativos.

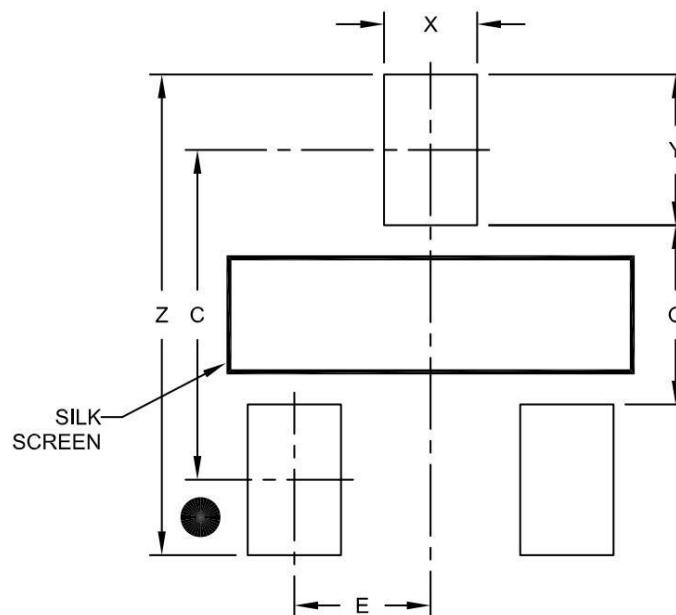
Desenho de tecnologia de microchip C04-21303 Rev A Folha 2 de 2

© 2017 Microchip Technology Inc.

12.5 SOT23 de 3 derivações

Transistor de contorno pequeno de plástico de 3 condutores (NB) [SOT-23]

Nota: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

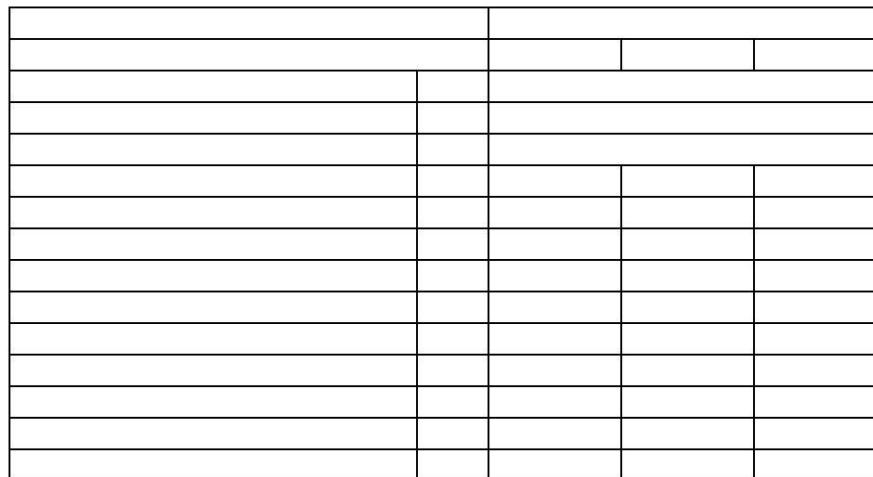
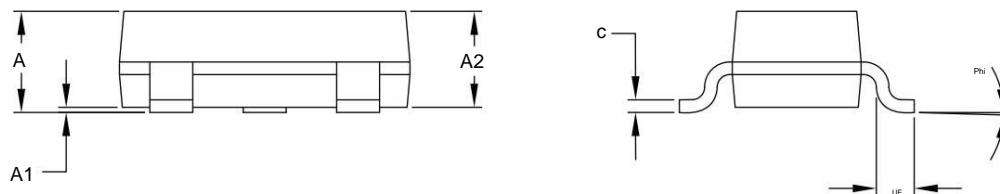
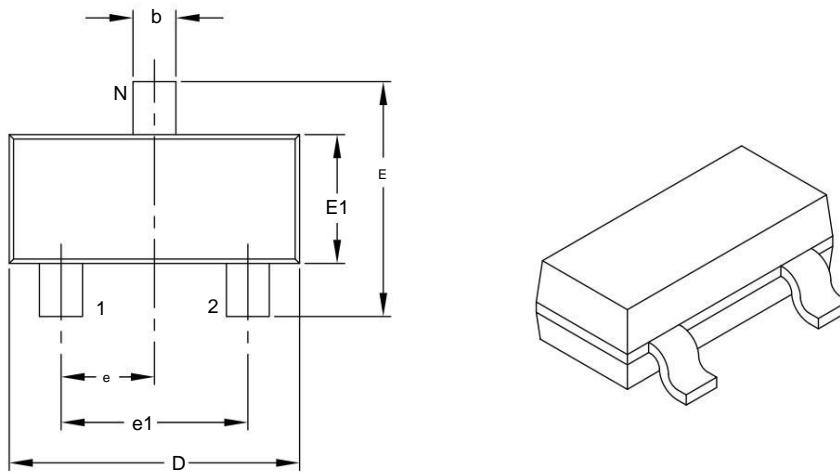
Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.95	BSC	
Contact Pad Spacing	C		2.30	
Contact Pad Width (X3)	X			0.65
Contact Pad Length (X3)	Y			1.05
Distance Between Pads	G	1.25		
Overall Width	Z			3.35

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

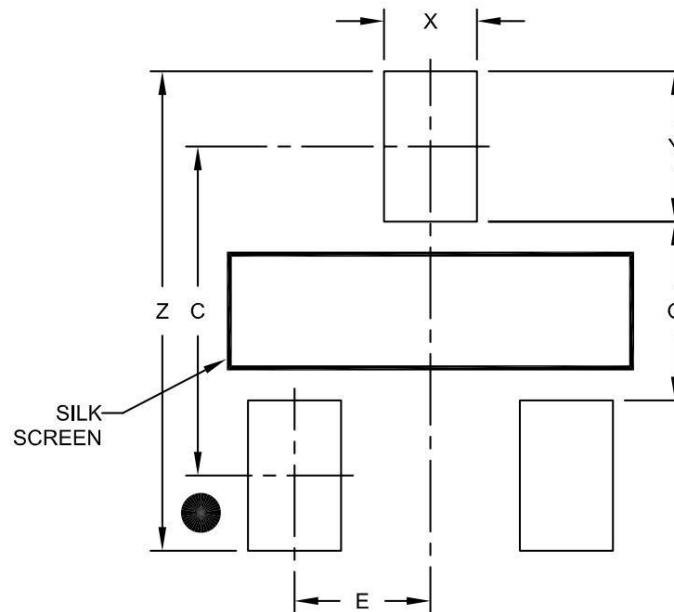
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2104A



3-Lead Plastic Small Outline Transistor (TT) [SOT-23]

Nota: Para os desenhos de embalagem mais atuais, consulte a especificação de embalagem da Microchip localizada em <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

Dimension	Limits	UNITS MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E		0.95 BSC	
Contact Pad Spacing	C		2.30	
Contact Pad Width (X3)	X			0.65
Contact Pad Length (X3)	Y			1.05
Distance Between Pads	G	1.25		
Overall Width	Z			3.35

Notes:

- Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2104A

Referência e Notas de Aplicação

13. Referência e Notas de Aplicação

O ATSHA204A implementa um protocolo de desafio-resposta usando SHA-256 ou HMAC/SHA-256, os detalhes são indicados abaixo. A resposta é sempre um resumo de 256 bits.

O comando Nonce (consulte a Seção [Comando Nonce](#)) aceita um desafio de entrada do sistema e opcionalmente o combina com um número aleatório gerado internamente para gerar um nonce (por exemplo, número usado uma vez) para o cálculo. A combinação é a semente e, em seguida, é combinada com uma chave secreta como parte do cálculo de autenticação para qualquer um dos comandos criptográficos (por exemplo, MAC, HMAC, Read, Write ou GenDig). O desafio de entrada também pode ser passado diretamente para o comando MAC.

O dispositivo pode garantir a unicidade do nonce somente se o dispositivo tiver incluído a saída de seu RNG no cálculo; isso ocorre porque a entrada do sistema pode ou não ser exclusiva. Cada nonce aleatório tem, de fato, a garantia de ser único quando comparado a todos os nonces anteriores, garantindo que cada transação seja única ao longo do tempo.

13.1 SHA-256

O comando ATSHA204A MAC calcula o resumo de uma chave secreta concatenada com o desafio ou nonce. Ele opcionalmente inclui várias outras informações armazenadas no dispositivo dentro do digerido mensagem.

O ATSHA204A calcula o resumo SHA-256 com base no algoritmo documentado no seguinte site:

<http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>

A mensagem SHA-256 completa processada pelo ATSHA204A está listada na descrição do comando para cada comando que usa o algoritmo. A maioria das implementações de software padrão do algoritmo adiciona automaticamente o número apropriado de bits de preenchimento e comprimento a esta mensagem para corresponder à operação que o dispositivo executa internamente.

O ATSHA204A também pode calcular um resumo SHA-256 usando o comando SHA. O chamador é responsável por fornecer os bytes de preenchimento e comprimento para a mensagem. O tamanho da mensagem deve ser um múltiplo de 64 bytes, incluindo os bytes de preenchimento.

O algoritmo SHA-256 é usado para criptografia, obtendo o resumo de saída do algoritmo de hash e fazendo um XOR com os dados de texto simples para produzir o texto cifrado. A descriptografia é a operação inversa, ou seja, o texto cifrado é XORed com o resumo com o resultado sendo o texto simples.

13.2 HMAC/SHA-256

A resposta ao desafio também pode ser calculada usando o algoritmo HMAC baseado em SHA-256 documentado no seguinte site:

<http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf>

Devido à maior complexidade de computação, o comando HMAC não é tão flexível quanto o comando MAC e o tempo de computação para HMAC é estendido. Embora a sequência HMAC não seja necessária para garantir a segurança do resumo, ela é incluída para compatibilidade com vários pacotes de software.

13.3 Valores Chave

Todas as chaves dentro do SHA204A têm 256 bits de comprimento. O ATSHA204A usa essas chaves como parte das mensagens que são hash com os comandos MAC, CheckMac, HMAC e GenDig. Qualquer slot na zona de dados da EEPROM pode ser usado para armazenar uma chave, no entanto, o valor é secreto apenas se as permissões de leitura e gravação estiverem definidas corretamente no SlotConfig (incluindo o bit IsSecret).

Exceto pelo comando GenDig, todos, exceto os quatro bits menos significativos do parâmetro SlotID, são ignorados na determinação da origem dos dados de chave. Apenas os quatro bits menos significativos são usados para selecionar um dos slots da zona de dados. Consulte a Seção [Chaves de transporte](#) para obter informações sobre como o GenDig usa outros valores de SlotID.

Em todos os casos em que um cálculo SHA-256 é executado usando Param2, todo o SlotID de 16 bits como entrada é incluído na mensagem.

13.3.1 Chaves Diversificadas

Se o host ou entidade de validação tiver um local para armazenar segredos com segurança, os valores de chave armazenados no(s) slot(s) EEPROM podem ser diversificados com o número de série embutido no dispositivo (SN<0:8>). Desta forma, cada dispositivo Cliente pode ter uma chave única, que pode fornecer proteção extra contra ataques de texto sem formatação conhecidos e permitir que números de série comprometidos sejam identificados e colocados na lista negra.

Para implementar isso, um segredo raiz é combinado externamente com o número de série do dispositivo durante a personalização usando algum algoritmo criptográfico e o resultado gravado no slot de chave ATSHA204A.

O comando ATSHA204A CheckMac fornece um mecanismo de geração e comparação segura de chaves diversificadas, eliminando esse requisito do sistema Host.

Consulte a seguinte nota de aplicação para obter mais detalhes:

<http://ww1.microchip.com/downloads/en/appnotes/doc8666.pdf>

13.3.2 Chaves enroladas

Para evitar o uso repetido do mesmo valor de chave, o ATSHA204A oferece suporte à rolagem de chave. Normalmente, após um certo número de usos (talvez apenas um), o valor da chave atual é substituído pelo resumo SHA-256 de seu valor atual combinado com algum deslocamento, que pode ser uma constante, algo relacionado ao sistema atual (por exemplo, um número de série ou número de modelo) ou um número aleatório.

Esse recurso é implementado usando o comando DeriveKey. Antes da execução do comando DeriveKey, o comando Nonce deve ser executado para carregar o deslocamento em TempKey. Cada vez que a operação de rolagem é executada nos slots de 0 a 7, o campo UpdateCount desse slot é incrementado.

Um uso desse recurso é remover permanentemente a chave original do dispositivo e substituí-la por uma chave útil apenas em um ambiente específico. Depois que a chave é rolada, não há como recuperar o valor antigo, o que melhora a segurança do sistema.

Qualquer interrupção de energia durante a execução do comando DeriveKey no modo Roll pode fazer com que a chave ou o UpdateCount tenham um valor desconhecido. Se a gravação em um slot for habilitada usando o bit número 14 de SlotConfig, essas chaves podem ser gravadas de forma criptografada e autenticada usando o comando Write.

Como alternativa, várias cópias da chave podem ser armazenadas em vários slots para que a falha de um único slot não incapacite o sistema.

13.3.3 Chaves criadas

Para oferecer suporte a chaves efêmeras exclusivas para cada cliente, o ATSHA204A também oferece suporte à criação de chaves.

Nesse mecanismo, uma chave “pai” (especificada por SlotConfig.writeKey) é combinada com um nonce fixo ou aleatório para criar uma chave exclusiva, que é usada para qualquer finalidade criptográfica.

A capacidade de criar chaves exclusivas é especialmente útil se a chave pai tiver restrições de uso (consulte a Seção [Chaves de uso limitado](#) e a Seção [Chave de uso limitado](#) nas seções a seguir). Nesse modo, a chave pai de uso limitado pode ser empregada para criar uma chave filho de uso ilimitado. Como a chave filha é útil apenas para esse par Host-Cliente específico, os ataques em seu valor são menos valiosos.

Esse recurso também é implementado usando o comando DeriveKey. Antes da execução do comando DeriveKey, o comando Nonce deve ser executado para carregar o valor nonce em TempKey. Cada vez que a operação de criação é executada nos slots de 0 a 7; o campo UpdateCount para esse slot é incrementado.

13.3.4 Chaves de uso limitado

Para os valores de SlotID correspondentes aos slots de 0 a 7 na seção de dados da EEPROM, o uso repetido da chave armazenada no slot pode ser estritamente limitado. Esse recurso é ativado se o bit LimitedUse for definido no campo SlotConfig. O bit LimitedUse é ignorado para os slots 8 a 14. O número de usos restantes é armazenado como um mapa de bits no byte UseFlag correspondente ao slot em questão.

Antes da execução de qualquer comando criptográfico que usa esse slot como chave, ocorre o seguinte: Se

- SlotConfig<SlotID>.LimitedUse for definido e UseFlag<SlotID> for 0x00, o dispositivo retornará um erro.
- Começando no bit 7 de UseFlag<SlotID>, limpe para zero o primeiro bit que atualmente é um.

Na prática, este procedimento permite que as chaves LimitedUse sejam usadas oito vezes entre “refreshes” usando o comando DeriveKey. Se a energia for perdida durante a execução de qualquer comando referenciando uma chave que tenha esse recurso ativado, um dos bits de uso em UseFlag ainda poderá ser limpo, mesmo que o comando não tenha sido concluído. Por este motivo, a Microchip recomenda que a chave seja utilizada uma única vez, deixando os restantes bits com margem de segurança para erros.

Em circunstâncias normais, todos os oito bytes UseFlag devem ser inicializados para 0xFF. Se for intenção permitir menos de oito usos de uma determinada chave, esses bytes devem ser inicializados em 0x7F (sete usos), 0x3F (seis usos), 0x1F (cinco usos), 0x0F (quatro usos), 0x07 (três usos), 0x03 (dois usos) ou 0x01 (um uso). A inicialização para qualquer outro valor além desses valores ou 0xFF é proibida.

Os comandos Read, Write e DeriveKey operam de forma ligeiramente diferente, conforme observado abaixo:

Ler e Escrever Esses

comandos ignoram o estado do bit LimitedUse e o byte UseFlag não muda como resultado de sua execução. Os slots LimitedUse nos quais o UseFlag está esgotado (valor de 0x00) ainda podem ser lidos ou gravados (sujeito às limitações SlotConfig apropriadas), embora o valor no slot nunca possa ser usado como uma chave para comandos criptográficos.

Se SlotConfig.WriteKey para o slot X apontar de volta para X, mas UseFlag<X> estiver esgotado, as gravações criptografadas no slot nunca serão bem-sucedidas porque o comando GenDig anterior retornará um erro devido à limitação de uso. Uma situação semelhante ocorre com leituras e ReadKey. Os slots usados como chaves nunca devem ter IsSecret definido como zero ou WriteConfig definido como sempre. • **DeriveKey** Se a chave pai

for usada para

autenticação ou como fonte, se LimitedUse (para o pai) for definido e UseFlag (também para o pai) for 0x00, o comando DeriveKey retornará um erro. Os bits LimitedUse e UseFlag são ignorados para a chave de destino. Quando executado com sucesso,

DeriveKey sempre redefine o UseFlag para 0xFF para a chave de destino. Este é o único mecanismo para redefinir os bits UseFlag.

O uso do comando DeriveKey é opcional. É legal ser executado apenas se WriteConfig<13> estiver definido para este slot.

Em algumas situações, pode ser vantajoso simplesmente ter uma chave que pode ser usada oito vezes, caso em que os outros comandos criptográficos limpam os bits em UseFlag um por vez até que todos sejam limpos e, nesse momento, a chave é desabilitada.

13.3.5 Chave de uso limitado

Se Slot<15>.LimitedUse for definido, o uso da chave número 15 será limitado por meio de um mecanismo diferente da limitação de uso único descrita acima, que se aplica apenas aos slots de 0 a 7.

Antes de qualquer uso da chave 15 por um comando criptográfico, ocorre o seguinte:

- Se todos os bytes em LastKeyUse forem 0x00, retornará o erro.
- Começando no bit 7 do primeiro byte de LastKeyUse (byte 68 na zona de configuração), limpe para zero o primeiro bit que atualmente é um. Se o byte 68 for 0x00, verifique o bit sete do byte 69 e assim por diante até o byte 83. Apenas um único bit é limpo a cada vez antes de usar a chave 15.

Não há mecanismo de redefinição para essa limitação; após 128 usos (ou o número de um bits definido em LastKeyUse na personalização), a chave 15 é desabilitada permanentemente. Esta capacidade não é suscetível a interrupções de energia.

Mesmo que a energia seja interrompida durante a execução do comando, apenas um único bit em LastKeyUse é desconhecido; todos os outros bits em LastKeyUse permanecem inalterados e a chave permanece inalterada.

Se forem desejados menos de 128 usos para a chave 15, alguns dos bytes dentro dessa matriz não devem ser inicializados com 0xFF. Assim como UseFlag, os únicos valores permitidos para bytes dentro desse campo (além de 0xFF) são 0x7F, 0x3F, 0x1F, 0x0F, 0x07, 0x03, 0x01 ou 0x00. O número total de bits definido como um indica o número de utilizações.

Exemplo: contagem de uso limitado definida para um valor de 16.

0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00

O bit LimitedUse é ignorado pelos comandos Read e Write e LastKeyUse não muda como resultado de sua execução. O bit LimitedUse é ignorado pela função de cópia do comando CheckMac. O bit LimitedUse é respeitado para a chave pai no comando DeriveKey, mas é ignorado para a chave de destino.

13.3.6 Verificação de senha

Muitos aplicativos exigem que o usuário insira uma senha para habilitar recursos, descriptografar dados armazenados ou para buscar algum outro propósito. Normalmente, a senha esperada deve ser armazenada em algum lugar na memória e, portanto, pode se tornar sujeita a descoberta. O ATSHA204A pode armazenar com segurança a senha esperada e executar várias operações úteis nela. A senha nunca é passada abertamente para o dispositivo e não pode ser lida no dispositivo. É hash com um número aleatório no software do sistema antes de ser passado para o dispositivo. O nonce em TempKey sempre deve ter sido gerado usando o RNG interno quando uma Chave de transporte é utilizada.

A capacidade de cópia do comando CheckMac permite os seguintes tipos de opções de verificação de senha: 1. CheckMac faz uma

comparação interna com a senha esperada e retorna um Booleano para o sistema para indicar se a senha foi digitada corretamente ou não.

2. Se o dispositivo determinar que a senha correta foi inserida, o valor do
a senha pode opcionalmente ser combinada com um valor armazenado ou efêmero para criar uma chave que pode
ser usada pelo sistema para fins de proteção de dados.
3. Se o dispositivo determinar que a senha correta foi inserida, ele poderá usar esse fato para liberar opcionalmente um
segredo secundário de alta entropia, que pode ser usado para proteção de dados sem o risco de um ataque de
dicionário exaustivo.
4. Se a senha foi perdida, uma entidade com conhecimento de um valor de chave pai pode, opcionalmente, gravar uma nova senha
no slot. Opcionalmente, o valor atual pode ser criptografado com uma chave pai e lido no dispositivo.

As senhas devem ser armazenadas em slots de numeração par. Se a senha deve ser mapeada para um valor secundário
(use a Etapa 3 acima), o slot de destino que contém esse valor está localizado no próximo número de slot mais alto (o número
do slot de senha mais um); caso contrário, o slot de destino é o mesmo que o slot de senha.

ReadKey para o slot de destino deve ser definido como 0x0 para habilitar esse recurso. Para evitar o uso fraudulento ou não
intencional desse recurso, não defina ReadKey para nenhum slot como 0x0, a menos que esse recurso CheckMac/Copiar
seja especificamente necessário. Em particular, não assuma que outros bits na palavra de configuração para um determinado slot
substituem a ativação desse recurso especificado por ReadKey = 0x0.

Esse recurso é ativado somente se o parâmetro de modo para CheckMac tiver um valor de 0x01 ou 0x05 e
TempKey.SourceFlag corresponder a Mode<2>.

Nota: Deve-se tomar cuidado ao usar o Modo 0x05, pois o sistema está sujeito a um ataque de repetição; no
entanto, pode haver algumas configurações de sistema nas quais esse arranjo é vantajoso.

- Os primeiros 32 bytes da mensagem SHA-256 são armazenados em um slot de dados na EEPROM (o
senha).
- Os segundos 32 bytes da mensagem SHA-256 devem ser um nonce gerado aleatoriamente no
Registro TempKey.

Se as condições acima forem atendidas e a resposta de entrada corresponder ao resumo gerado internamente, o conteúdo
da chave de destino será copiado para TempKey. Os outros bits do registro TempKey são definidos da seguinte maneira:

- SourceFlag é definido como um (não aleatório).
- GenData é definido como zero (não gerado pelo comando GenData).
- CheckFlag é definido como zero (TempKey não está restrito ao comando CheckMac).
- Válido é definido como um.

13.3.7 Chaves de Transporte

O dispositivo ATSHA204A inclui uma matriz interna de chaves de hardware (chaves de transporte) destinadas à personalização segura antes
do bloqueio da seção de dados. Os valores das chaves de hardware são mantidos em segredo e são disponibilizados a clientes
qualificados mediante solicitação à Microchip. Essas chaves podem ser usadas apenas com o comando GenDig e são indicadas por
um valor de SlotID maior ou igual a 0x8000.

Para GenDig e todos os outros comandos, valores de SlotID menores que 0x8000 sempre fazem referência a chaves
armazenadas na zona de dados da EEPROM. Nesses casos, apenas os quatro bits menos significativos de SlotID são
usados para determinar o número do slot, enquanto todo o SlotID de 16 bits como entrada é usado em qualquer
cálculo de mensagem SHA-256.

14. Histórico de revisão

Revisão A (abril de 2018)

Lançamento original do documento no formato Microchip.

Esta versão substitui a revisão 8885H do documento Atmel de 11/2015.

Site da Microchip

A Microchip fornece suporte online através do nosso site em <http://www.microchip.com/>. Este site é usado como um meio de disponibilizar arquivos e informações facilmente aos clientes. Acessível usando seu navegador de Internet favorito, o site contém as seguintes informações:

- **Supor te ao produto** - folhas de dados e errata, notas de aplicação e programas de amostra, design recursos, guias do usuário e documentos de suporte de hardware, versões de software mais recentes e software arquivado
- **Supor te Técnico Geral** - Perguntas Frequentes (FAQ), solicitações de suporte técnico, grupos de discussão on-line, listagem de membros do programa de consultoria da Microchip •
- **Business of Microchip** – seletor de produtos e guias de pedidos, comunicados de imprensa mais recentes da Microchip, lista de seminários e eventos, listas de escritórios de vendas, distribuidores e representantes de fábrica da Microchip

Serviço de notificação de alteração do cliente

O serviço de notificação ao cliente da Microchip ajuda a manter os clientes atualizados sobre os produtos da Microchip. Os assinantes receberão notificação por e-mail sempre que houver alterações, atualizações, revisões ou errata relacionadas a uma família de produtos específica ou ferramenta de desenvolvimento de interesse.

Para se registrar, acesse o site da Microchip em <http://www.microchip.com/>. Em "Suporte", clique em "Notificação de alteração do cliente" e siga as instruções de registro.

Supor te ao cliente

Os utilizadores de produtos Microchip podem receber assistência através de vários canais:

- Distribuidor ou Representante
- Escritório de vendas local
- Engenheiro de Aplicação de Campo (FAE) •

Supor te Técnico

Os clientes devem entrar em contato com seu distribuidor, representante ou Engenheiro de Aplicação de Campo (FAE) para obter suporte.

Os escritórios de vendas locais também estão disponíveis para ajudar os clientes. Uma lista de escritórios de vendas e locais está incluída no verso deste documento.

O suporte técnico está disponível no site: <http://www.microchip.com/support>

Sistema de identificação do produto

Para solicitar ou obter informações, por exemplo, sobre preço ou entrega, consulte a fábrica ou o escritório de vendas listado.

PART NO. -XXX XX -X
 Device Package I/O Type Tape and Reel

Dispositivo:	ATSHA204A: Coprocessador criptográfico com armazenamento seguro de chaves baseado em hardware	
Opções de pacote	SSH	= 8S1, 8 condutores (corpo largo de 0,150"), contorno pequeno de asa de gaivota de plástico (JEDEC SOIC)
	UE	= 8MA2, 8-Pad 2 x 3 x 0,6 mm Corpo, Plástico Termicamente Aprimorado Ultra Fino Plano Duplo Sem Chumbo (UDFN)
	RBH	= 3RB, corpo de 3 condutores 2,5 x 6,5 mm, passo de 2,0 mm, pacote CONTACT (serrado).
Tipo de E/S	CZ	= Interface de fio único
	E	= Interface I2C
Opções de fita e bobina	B	= Lucro
	T	= Bobina grande (o tamanho varia de acordo com o tipo de embalagem)
	S	= Carretel Pequeno (Somente disponível para MAH)

Exemplos:

- ATSHA204A-SSHCZ-T: Fio único, fita e bobina, 4.000 por rolo, pacote SOIC de 8 condutores • ATSHA204A-SSHCZ

B: Fio único, tubo, 100 por tubo, pacote SOIC de 8 condutores • ATSHA204A-SSHDA-T: I2C, fita e bobina,

4.000 por bobina, pacote SOIC de 8 condutores • ATSHA204A-SSHDA-B: I2C, tubo, 100 por tubo, pacote

SOIC de 8 condutores • ATSHA204A-MAHCZ-T: fio único, fita e bobina, 15.000 por bobina,

pacote UDFN de 8 almofadas • ATSHA204A-MAHDA-T: I2C, fita e bobina, 15.000 por bobina, pacote UDFN de 8 almofadas • ATSHA204A-MAHCZ-S: fio único, fita e bobina, 3.000 por bobina, pacote UDFN de 8 almofadas •

ATSHA204A-MAHDA-S: I2C, fita e bobina, 3.000 por bobina, pacote UDFN de 8 almofadas • ATSHA204A-RBHCZ-T: fio

único, fita e bobina, 5.000 por bobina, 3 -Conjunto de contato de chumbo • ATSHA204A-RBHCZ-B: Fio único,

tubo, 56 por tubo, pacote de contato de 3 condutores • ATSHA204A-STUCZ-T: Fio único, fita e bobina, 5000 por bobina,

SOT-23 de 3 condutores pacote • ATSHA204A-XHDA-T:I2C, fita e bobina, 5000 por bobina, pacote TSSOP

de 8 derivações • ATSHA204A-XHCZ-T: fio único, fita e bobina, 500 0 por carretel, pacote TSSOP de 8 condutores

Observação:

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 da Microchip a disponibilidade de pacotes com a opção de fita e bobina.
2. Opções de embalagem de formato pequeno podem estar disponíveis. Verifique <http://www.microchip.com/embalagem> para disponibilidade de pacote de formato pequeno ou entre em contato com o escritório de vendas local.

Recurso de Proteção de Código de Dispositivos Microchip

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

- Os produtos da Microchip atendem às especificações contidas em sua folha de dados específica do Microchip. • A Microchip acredita que sua família de produtos é uma das famílias mais seguras de seu tipo no mercado hoje, quando usado 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, até onde sabemos, requerem o uso dos produtos da Microchip de uma maneira fora das especificações operacionais contidas nas folhas de dados da Microchip. Muito provavelmente, a pessoa que faz isso está envolvida em roubo de propriedade intelectual.
- A Microchip está disposta a trabalhar com o cliente que está preocupado com a integridade do seu código.
- Nem a Microchip nem qualquer outro fabricante de semicondutores pode garantir a segurança de seu código. A proteção do 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 pode ter o direito de processar judicialmente por meio dessa lei.

Notícia legal

As informações contidas nesta publicação sobre aplicativos de dispositivos e similares são fornecidas apenas para sua conveniência e podem ser substituídas por atualizações. É sua responsabilidade garantir que sua aplicação atenda às suas especificações.

A MICROCHIP NÃO FAZ REPRESENTAÇÕES OU GARANTIAS DE QUALQUER TIPO, SEJA EXPRESSA OU IMPLÍCITA, ESCRITA OU ORAL, ESTATUTÁRIA OU DE OUTRA FORMA, RELACIONADA ÀS INFORMAÇÕES, INCLUINDO, SEM LIMITAÇÃO, SUA CONDIÇÃO, QUALIDADE, DESEMPENHO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM PROPÓSITO.

A Microchip exime-se de qualquer responsabilidade decorrente desta informação e da sua utilização. O uso de dispositivos Microchip em suporte de vida e/ou aplicações de segurança é inteiramente por conta e risco do comprador, e o comprador concorda em defender, indemnizar e isentar a Microchip de todos e quaisquer danos, reivindicações, processos ou despesas resultantes de tal uso. Nenhuma licença é transmitida, implícita ou de outra forma, sob quaisquer direitos de propriedade intelectual da Microchip, salvo indicação em contrário.

Marcas Registradas

O nome e logotipo Microchip, o logotipo Microchip, AnyRate, AVR, logotipo AVR, AVR Freaks, BeaconThings, BitCloud, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, logotipo KeeLoq, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, logotipo MOST, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, logotipo PIC32, Prochip Designer, QTouch, RightTouch, SAM-BA, SpyNIC, SST, logotipo SST, 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

Supressão de Tecla Adjacente, AKS, Analógico para a Era Digital, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, CryptoAuthentication, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Média Dinâmica Correspondência, DAM, ECAN , EtherGREEN, Programação Serial In-Circuit, ICSP, Conectividade Inter-Chip, JitterBlocker, KleerNet, KleerNet logotipo, Mindi, MiWi, motorBench, MPASM, MPF, logotipo certificado MPLAB, MPLIB, MPLINK, MultiTRAK, NetDetach, Onisciente Geração de código, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, QMatrix, logo RightTouch, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-IS, SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA e ZENA são marcas comerciais da Microchip Technology Incorporated nos EUA e em 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 registradas aqui mencionadas são de propriedade de suas respectivas empresas.

© 2018, Microchip Technology Incorporated, impresso nos EUA, todos os direitos reservados.

ISBN: 978-1-5224-2809-1

Sistema de Gestão da Qualidade Certificado pela DNV

ISO/TS 16949

A Microchip recebeu a certificação ISO/TS-16949:2009 para sua sede mundial, design e instalações de 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 empresa são para seus PIC® MCUs e dsPIC® DSCs, dispositivos de salto de código KEELOQ®, Serial EEPROMs, microperiféricos, memória não volátil e produtos analógicos. Além disso, o sistema de qualidade da Microchip para o projeto e fabricação de sistemas de desenvolvimento é certificado pela ISO 9001:2000.


MICROCHIP

Vendas e serviços mundiais

AMÉRICAS	ÁSIA-PACÍFICO	ÁSIA-PACÍFICO	EUROPA
Escritório Corporativo 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Telefone: 480-792-7200 Fax: 480-792-7277 Suporte técnico: http://www.microchip.com/support Endereço Site Web: www.microchip.com Atlanta Duluth, Geórgia Telefone: 678-957-9614 Fax: 678-957-1455 Austin, Texas Telefone: 512-257-3370 Boston Westborough, MA Telefone: 774-760-0087 Fax: 774-760-0088 Chicago Itasca, IL Telefone: 630-285-0071 Fax: 630-285-0075 Dallas Addison, Texas Tel: 972-818-7423 Fax: 972-818-2924 Detroit Novi, MI Tel: 248-848-4000 Houston, Texas Tel: 281-894-5983 Indianápolis Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 Os anjos Missão Velha, Califórnia Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 Raleigh, Carolina do Norte Tel: 919-844-7510 Nova York, NY Telefone: 631-435-6000 São José, Califórnia Telefone: 408-735-9110 Telefone: 408-436-4270 Canadá - Toronto Tel: 905-695-1980 Fax: 905-695-2078	Austrália - Sidney Tel: 61-2-9868-6733 China - Pequim Tel: 86-10-8569-7000 China - Chengdu Tel: 86-28-8665-5511 China - Chongqing Tel: 86-23-8980-9588 China-Dongguan Telefone: 86-769-8702-9880 China - Cantão Tel: 86-20-8755-8029 China - Hangzhou Telefone: 86-571-8792-8115 China - Nanjing Tel: 86-25-8473-2460 China - Qingdao Telefone: 86-532-8502-7355 China - Xangai Tel: 86-21-3326-8000 China - Shenyang Tel: 86-24-2334-2829 China - Shenzhen Telefone: 86-755-8864-2200 China - Suzhou Telefone: 86-186-6233-1526 China - Wuhan Telefone: 86-27-5980-5300 China - Xian Tel: 86-29-8833-7252 China - Xiamen Tel: 86-592-2388138 China - Zuhai Tel: 86-756-3210040	Índia - Bangalore Tel: 91-80-3090-4444 Índia - Nova Deli Tel: 91-11-4160-8631 Índia - Pune Tel: 91-20-4121-0141 Japão - Osaka Telefone: 81-6-6152-7160 Japão - Tóquio Tel: 81-3-6880-3770 Coreia - Daegu Telefone: 82-53-744-4301 Coreia - Seul Telefone: 82-2-554-7200 Malásia - Kuala Lumpur Telefone: 60-3-7651-7906 Malásia - Penang Telefone: 60-4-227-8870 Filipinas - Manila Telefone: 63-2-634-9065 Cingapura Tel: 65-6334-8870 Taiwan - Hsin Chu Tel: 886-3-577-8366 Taiwan - Kaohsiung Tel: 886-7-213-7830 Taiwan - Taipé Tel: 886-2-2508-8600 Tailândia - Bangkok Telefone: 66-2-694-1351 Vietnã - Ho Chi Minh Tel: 84-28-5448-2100	Áustria - Peixe-gato Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 Dinamarca - Copenhaga Tel: 45-4450-2828 Fax: 45-4485-2829 Finlândia - Espoo Tel: 358-9-4520-820 frança paris Telefone: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 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 Telefone: 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 Holanda - Drunen Tel: 31-416-690399 Fax: 31-416-690340 Noruega - Trondheim Tel: 47-7289-7561 Polónia - Varsóvia Tel: 48-22-3325737 Romênia - Bucareste Telefone: 40-21-407-87-50 Espanha - Madri Telefone: 34-91-708-08-90 Fax: 34-91-708-08-91 Suécia - Gotemburgo Telefone: 46-31-704-60-40 Suécia - Estocolmo Telefone: 46-8-5090-4654 Reino Unido - Wokingham Telefone: 44-118-921-5800 Fax: 44-118-921-5820