SLEMAN ACADEMY PROGRAMAÇÃO MICROCONTROLADORES ARM

Sumário

Int	trodução5Para quem é essa apostila5Sobre o autor5Agradecimentos5Materiais necessários5					
1	Introdução aos Microcontroladores51.1 O que são Microcontroladores?51.2 Aplicações de Microcontroladores51.3 Diferença entre Microcontroladores e Microprocessadores5					
2	Arquitetura do STM32F103 5 2.1 Estrutura Geral do STM32F103 5 2.1.1 Núcleo ARM Cortex-M3 5 2.1.2 Memória (Flash, RAM, EEPROM) 5 2.2 Periféricos do STM32F103 5 2.2.1 GPIO (General Purpose Input/Output) 5 2.2.2 ADC (Conversor Analógico-Digital) 5 2.2.3 Timers 5 2.2.4 USART (Universal Asynchronous Receiver/Transmitter) 5 2.3 Registradores 5 2.3.1 O que são registradores? 5 2.3.2 Estrutura e função dos registradores no STM32F103 6					
3	Programação Bare Metal vs. Uso de Frameworks63.1 O que é Programação Bare Metal?63.2 Vantagens e Desvantagens da Programação Bare Metal63.3 Introdução a Frameworks (Arduino, HAL)63.4 Comparação: Bare Metal x Framework6					
4	Configuração do Ambiente de Desenvolvimento64.1 Ferramentas Necessárias64.1.1 Hardware (placa STM32F103, gravador/programador)64.1.2 Software (IDE, compilador)64.2 Ideias de IDEs64.2.1 STM32CubeIDE64.2.2 Keil uVision64.2.3 IAR Embedded Workbench64.3 Uso de Gravadores e Debuggers64.3.1 Métodos de programação (JTAG, SWD)64.3.2 Configuração do gravador6					
5	Fundamentos da Programação em C65.1 Estruturas Básicas da Linguagem C65.1.1 Sintaxe e semântica65.1.2 Tipos de dados6					

	5.2	5.2.1	cões Bitwise7O que são operações bitwise?7Exemplos práticos de operações7							
6	Prim	neiros P	assos com o STM32F103 7							
	6.1		uração Inicial do Projeto							
			Estrutura de um projeto bare metal							
			Macro de um registrador							
			Configurando o compilador e includes							
	6.2		um LED: O Primeiro Programa							
			Configuração de GPIO							
			Implementação do delay com loop for							
	6.3		ção a Delay e Timers							
			Implementação de delay usando timers - Systick							
7	PWN	/I e Gera	ação de Tempo 7							
	7.1	O que	é PWM (Modulação por Largura de Pulso)?							
			Conceito e aplicações							
	7.2		uração do PWM no STM32F103							
	7.3		los Práticos de PWM							
			Controle de brilho de LED							
		7.3.2	Controle de velocidade de motores							
8	Estr		(Structs) e Representação de Registradores 7							
	8.1		ção às Estruturas em C							
			O que são estruturas?							
			Sintaxe e uso básico de structs							
			Diferença entre structs e outras estruturas de dados							
	8.2	_	ens de Usar Estruturas							
			Organização do código							
			Legibilidade e manutenção							
	0.0		Agrupamento lógico de dados							
	8.3		sentando Registradores com Estruturas							
			Definindo um struct para GPIO							
			Inicialização de um registrador usando a struct							
		8.3.3	Utilização da struct para melhor legibilidade do código							
9			Sinais Analógicos 8							
	9.1		ção ao ADC (Conversor Analógico-Digital)							
	0.0		Funcionamento do ADC							
	9.2		uração do ADC no STM32F103							
	0.0		Passos para configurar o ADC							
	9.3		de um potenciometro							
		9.3.1	Utilizando o debug para ver o resultado da conversão							
10	Interrupções									
	10.1		são Interrupções?							
	10.0		Funcionamento e utilização de interrupções							
	10.2	Johns	mayao ao mionapyoos an io							

10.2.1 Exemplos de configuração de interrupções por botõe 10.3 Interrupções de Timer		 		8 8 9
10.6 Cronômetro básico				9
11 Projetos Finais 11.1 Projeto 1: Controle de LED com Botão e PWM 11.2 Projeto 3: Cronômetro com Display LED e Botões 11.3 Projeto 3: Blinky em outro microcontrolador ARM				9 0
Conclusão Considerações finais				9

Introdução

Aqui está o conteúdo da sua introdução.

Para quem é essa apostila

Sobre o autor

Agradecimentos

Materiais necessários

dasd

1 Introdução aos Microcontroladores

- 1.1 O que são Microcontroladores?
- 1.2 Aplicações de Microcontroladores
- 1.3 Diferença entre Microcontroladores e Microprocessadores
- 2 Arquitetura do STM32F103
- 2.1 Estrutura Geral do STM32F103
- 2.1.1 Núcleo ARM Cortex-M3
- 2.1.2 Memória (Flash, RAM, EEPROM)
- 2.2 Periféricos do STM32F103
- 2.2.1 GPIO (General Purpose Input/Output)
- 2.2.2 ADC (Conversor Analógico-Digital)
- **2.2.3 Timers**
- 2.2.4 USART (Universal Asynchronous Receiver/Transmitter)
- 2.3 Registradores
- 2.3.1 O que são registradores?

- 2.3.2 Estrutura e função dos registradores no STM32F103
- 3 Programação Bare Metal vs. Uso de Frameworks
- 3.1 O que é Programação Bare Metal?
- 3.2 Vantagens e Desvantagens da Programação Bare Metal
- 3.3 Introdução a Frameworks (Arduino, HAL)
- 3.4 Comparação: Bare Metal x Framework
- 4 Configuração do Ambiente de Desenvolvimento
- 4.1 Ferramentas Necessárias
- 4.1.1 Hardware (placa STM32F103, gravador/programador)
- 4.1.2 Software (IDE, compilador)
- 4.2 Ideias de IDEs
- 4.2.1 STM32CubeIDE
- 4.2.2 Keil uVision
- 4.2.3 IAR Embedded Workbench
- 4.3 Uso de Gravadores e Debuggers
- 4.3.1 Métodos de programação (JTAG, SWD)
- 4.3.2 Configuração do gravador
- 5 Fundamentos da Programação em C
- 5.1 Estruturas Básicas da Linguagem C
- 5.1.1 Sintaxe e semântica
- 5.1.2 Tipos de dados

5 2	\bigcirc	perações	Ditwico
5.2	U	vei açues	DILWISE

- 5.2.1 O que são operações bitwise?
- 5.2.2 Exemplos práticos de operações

6 Primeiros Passos com o STM32F103

- 6.1 Configuração Inicial do Projeto
- 6.1.1 Estrutura de um projeto bare metal
- 6.1.2 Macro de um registrador
- 6.1.3 Configurando o compilador e includes
- 6.2 Piscar um LED: O Primeiro Programa
- 6.2.1 Configuração de GPIO
- 6.2.2 Implementação do delay com loop for
- 6.3 Introdução a Delay e Timers
- 6.3.1 Implementação de delay usando timers Systick

7 PWM e Geração de Tempo

- 7.1 O que é PWM (Modulação por Largura de Pulso)?
- 7.1.1 Conceito e aplicações
- 7.2 Configuração do PWM no STM32F103
- 7.3 Exemplos Práticos de PWM
- 7.3.1 Controle de brilho de LED
- 7.3.2 Controle de velocidade de motores
- 8 Estruturas (Structs) e Representação de Registradores
- 8.1 Introdução às Estruturas em C
- 8.1.1 O que são estruturas?

8.1.2 Sintaxe e uso basico de structs
8.1.3 Diferença entre structs e outras estruturas de dados
8.2 Vantagens de Usar Estruturas
8.2.1 Organização do código
8.2.2 Legibilidade e manutenção
8.2.3 Agrupamento lógico de dados
8.3 Representando Registradores com Estruturas
8.3.1 Definindo um struct para GPIO
8.3.2 Inicialização de um registrador usando a struct
8.3.3 Utilização da struct para melhor legibilidade do código
9 Leitura de Sinais Analógicos
9.1 Introdução ao ADC (Conversor Analógico-Digital)
9.1.1 Funcionamento do ADC
9.2 Configuração do ADC no STM32F103
9.2.1 Passos para configurar o ADC
9.3 Leitura de um potenciometro
9.3.1 Utilizando o debug para ver o resultado da conversão
10 Interrupções
10.1 O que são Interrupções?
10.1.1 Funcionamento e utilização de interrupções
10.2 Configuração de Interrupções GPIO
10.2.1 Exemplos de configuração de interrupções por botões
10.3 Interrupções de Timer
10.3.1 Utilização de interrupções para temporização

- 10.4 Exemplos Práticos
- 10.5 Toggle de GPIO por interrupção de tempo
- 10.6 Cronômetro básico
- 11 Projetos Finais
- 11.1 Projeto 1: Controle de LED com Botão e PWM
- 11.2 Projeto 3: Cronômetro com Display LED e Botões
- 11.3 Projeto 3: Blinky em outro microcontrolador ARM

Conclusão

Considerações finais