

# EEN251-Microcontroladores e Sistemas Embarcados

## Pesquisa 3

### C para embarcados e kit de desenvolvimento

Bruna Tavares, Bruno Campos, Keneth Yamada

16 de março de 2016

## 1 C

### 1.1 Otimizações

É possível gerar códigos otimizados, seja alterando o algoritmo apenas, seja usando o GCC para tornar o código menor ou ainda mais rápido. Ao usar o GCC, os níveis de otimização disponíveis são:

- O0: nível padrão, no qual não há qualquer tentativa de otimização;
- O1: o GCC tenta reduzir o tempo de compilação e o tamanho do executável, possibilitando ainda o uso de depuradores;
- O2: melhor do que o nível O1, não causa aumento do arquivo por causa disso. Também é o melhor, mais seguro e o mais usado nos dias atuais na distribuição de softwares em Linux;
- O3: opção com melhor nível de otimização; entretanto, é a que mais causa efeitos colaterais: arquivos maiores, maior uso de memória RAM e poucas chances de fazer uso de um depurador sobre o programa executável;

## 1.2 volatile/const/static

Uma variável volatile indica ao compilador que a variável pode ser modificada sem o conhecimento do programa principal. Dessa forma, o compilador não pode prever com segurança se pode otimizar trechos de programa onde esta variável se encontra.

O modificador const faz com que a variável não possa ser modificada no programa. Seu uso mais comum é evitar que um parâmetro de uma função seja alterado pela função.

O funcionamento das variáveis declaradas como static depende se estas são globais ou locais. Variáveis globais static funcionam como variáveis globais dentro de um módulo, ou seja, são variáveis globais que não são conhecidas em outros módulos. Isto é útil se quisermos isolar pedaços de um programa para evitar mudanças acidentais em variáveis globais. Variáveis locais static são variáveis cujo valor é mantido de uma chamada da função para a outra.

## 1.3 MakeFile

O objetivo de Makefile é definir regras de compilação para projetos de software. Tais regras são definidas em arquivo chamado Makefile. O programa make interpreta o conteúdo do Makefile e executa as regras lá definidas.

O texto contido em um Makefile é usado para a compilação, ligação (linking), montagem de arquivos de projeto entre outras tarefas como limpeza de arquivos temporários, execução de comandos, etc.

Vantagens do uso do Makefile:

- Evita a compilação de arquivos desnecessários. Por exemplo, se um programa utiliza 120 bibliotecas e você altera apenas uma, o make descobre qual arquivo foi alterado e compila apenas a biblioteca necessária.
- Automatiza tarefas rotineiras como limpeza de vários arquivos criados temporariamente na compilação.
- Pode ser usado como linguagem geral de script embora seja mais usado para compilação.

## 1.4 ASCII

ASCII é um código binário que codifica um conjunto de 128 sinais: 95 sinais gráficos e 33 sinais de controle. Cada código binário possui 8 bits, sendo 7 bits para o propósito de codificação e 1 bit de paridade (detecção de erro).

A codificação ASCII é usada para representar textos em computadores, equipamentos de comunicação, entre outros dispositivos que trabalham com texto. Desenvolvida a partir de 1960, grande parte das codificações de caracteres modernas a herdaram como base.

O código ASCII é muito utilizado para conversão de Código Binário para Letras do alfabeto Maiúsculas ou minúsculas.

## 2 SAM4s-EK2

### 2.1 Gravador/Debug

A gravação via JTAG funciona com o acesso aos pinos TMS, TCK, TDI, TDO, nRESET, VTG, GND, KEY e permite que um programa seja depurado, diretamente no microcontrolador. O JTAG Daisy Chain é o conjunto de 3 registros que estão dispostos em cadeia, os registros são:

- device ID register - para saber que dispositivo é;
- bypass register - para pular o dispositivo e ir para o proximo na cadeia JTAG;
- boundary scan register - para controlar as entradas do dispositivo;

### 2.2 Jumpers

JP3 - jumper responsável por reinicializar a memória Flash e alguns dos seus bits não voláteis ao ser fechado.

JP9 - jumper responsável por, ao ser aberto, ativar o chip NCS0 Nand Flash selecionado para propostas customizadas.

JP13 - jumper responsável por, ao ser fechado, selecionar o chip NCS1 do LCD.

## 2.3 Clock

O clock da placa mãe de termina a velocidade do processador o microcontrolador ele pode ser controlado por 4 modos via software:

- MAINCK – Saída do oscilador principal de clock com a seleção do oscilador de cristal ou o oscilador Fast Rc de 4/8/12 MHz
- SLCK – Clock lento que é o único clock permanente do sistema.
- PLLACK – saída do Divider e um PLL programável de 60 a 130 MHz.
- PLLBCK = saída do Divider e um PLL programável de 60 a 130 MHz.

## 2.4 Alimentação

O nível de tensão de operação do microcontrolador é de 5 V e é alimentado pelo pino JP9.

## 2.5 LEDs

Existem 3 LEDs na placa, o azul(D2) e o verde(D3) são controlados pela GPIO, ambos se colocar 0V acendem. Já o LED vermelho indica se a placa está ligada em 3,3V, é também possível de se controlar pela GPIO como os outros, mas o controle dele é feito pelo resistor pull-up em que ao fechar a chave acende o LED.

## 2.6 Botões

Existem 2 botões mecânicos na placa que são conectados aos pinos PB3 e PB2 e definidos para serem os botões padrão esquerda e direita. Além do mais, tem um botão responsável por resetar o sistema o qual está ligado ao sinal NRST.

## 2.7 Periféricos

O SD/MMC Card serve para trabalhar com arquivo, podendo fazer leitura deles, além de utilizar a memória para melhorar o processamento e para isso a placa tem uma interface ultra rápida de 4 bits. Os pinos PA26(USART CTS), PA27(bit 2), PA28(bit 3), PA29(comando), PA30(clock do barramento), PA31(bit1), PA31(bit 0) e PA6 (detecção do cartão) controlam essa transferência de dados.

## 3 SAM4SD32C

### 3.1 Memória

- 2MB de memória flash e 160 KB de SRAM
- 2 KB de memória cache integrado

### 3.2 IOH, IOL

- IOH: 200uA
- IOL: 3uA

### 3.3 Brownout

Um Brownout é uma queda em um sistema de alimentação elétrica que pode ser de forma intencional ou não. Caso seja sem intenção, pode ser por causa de um efeito de rompimento da rede elétrica, mas se for intencional, pode ser graças a uma redução de carga do sistema em casos de emergências. A redução pode variar de horas a minutos, dependendo das situações e da tensão que alimentava o sistema.

### 3.4 Watchdog Timer

Watchdog Timer fiscaliza o programa principal e se houver alguma condição de erro, é disparado um reset ao sistema. Portanto, o dispositivo eletrônico Watchdog Timer é um sistema emergencial tentando a prevenção contra falhas. Caso o sistema do watchdog não for zerado ele pode reiniciar o sistema todo.

### 3.5 PIO

O Parallel Input/Output é o desempenho de várias entradas e saídas que são feitas simultaneamente em dispositivos. São importantes em sistemas operacionais e, portanto, para micro controladores. Para o caso do SAM4SD32C, tem-se as seguintes aplicações:

- Uma mudança entrada de interrupção permitindo a detecção mudança de nível em qualquer linha de I / O.

- Modos adicionais de interrupção permitindo borda de subida, borda de descida, de baixo nível ou detecção de alto nível em qualquer I / O.
- Um filtro contra falha de rejeição menor do que metade do ciclo do relógio periférico.
- Um filtro proporcionando a rejeição de impulsos indesejados de operações de chave ou botão.
- Capacidade multi-drive e controle de pull-up e pull-down de I / O.
- Visibilidade na Entrada e controle de Saída.

### 3.6 Custo

As faixas de preços em lojas americanas estão por volta de (fonte: Loja Verical em dólares):

- 10.41
- 11.56
- 9.13