Linguagens de montagem
Capítulo 9 - ARM - caraterísticas gerais

Ricardo Anido
Instituto de Computação
Unicamp

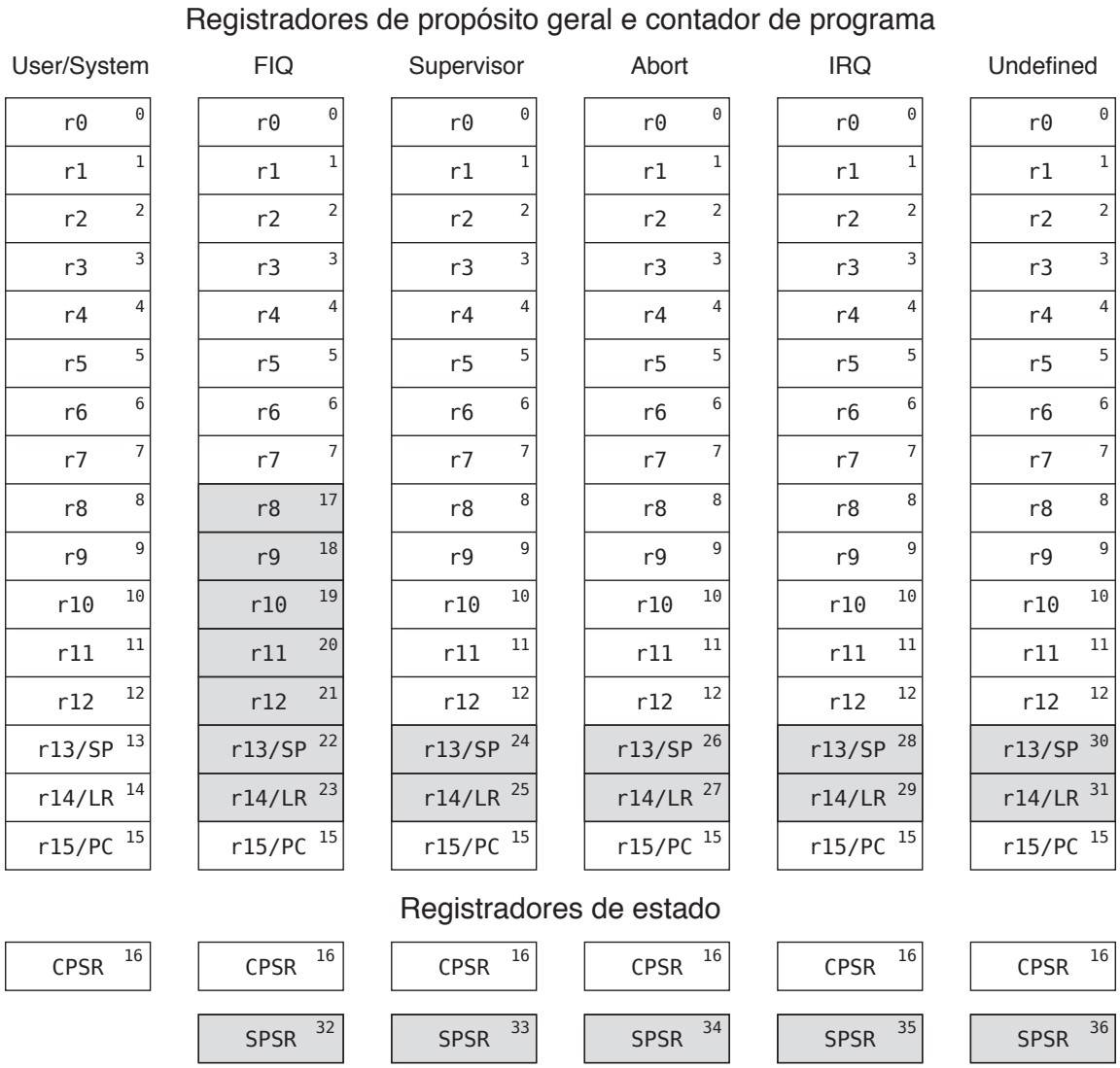
Modos de Operação

Nome	Descrição
User	Modo usuário, único modo sem nenhum privilégio de execução.
FIQ	Entra neste modo quando uma interrupção do tipo FIQ é aceita.
Interrupt	Entra neste modo quando uma interrupção do tipo IRQ é aceita.
Supervisor	Modo supervisor, entra neste modo quando o processador inicia, reinicia ou executa de chamada ao sistema.
Abort	Modo aborto de acesso, entra neste modo quando uma exceção de acesso a memória é disparada (acesso desalinhado de palavra, por exemplo).
Undefined	Modo instrução indefinida, entra neste modo quando uma exceção de instrução indefinida é disparada.
System	Modo sistema, único modo em que o processador entra por programa.

Registradores

- O processador ARM tem 37 registradores, mas apenas 17 (ou 18, em alguns modos de operação) são acessíveis a cada momento. D
- 13 são registradores de propósito geral (`r0` a `r12`).
- Os outros quatro registradores têm funções específicas:
  - `sp` (*stack pointer*), apontador de pilha, similar ao registrador homônimo do LEG, também acessado pelo nome `r13`.
  - `lr` (*link register*), registrador de ligação, também acessado pelo nome `r14`. Recebe o endereço de retorno em chamadas de procedimento.
  - `pc` (*program counter*), contador de programa, também acessado pelo nome `r15`.
  - `CPSR` registrador de estado corrente do programa.

Registradores

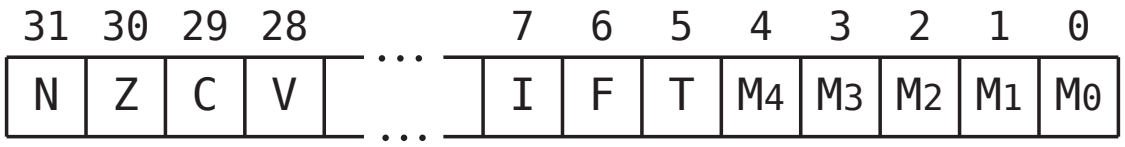


Execução condicional

Código	Sufixo	Condição	Descrição
0000	EQ	$Z = 1$	Igual
0001	NE	$Z = 0$	Diferente
0010	CS	$C = 1$	Maior ou igual, valor sem sinal
0011	CC	$C = 0$	Menor, valor sem sinal
0100	MI	$N = 1$	Negativo
0101	PL	$N = 0$	Positivo ou zero
0110	VS	$V = 1$	Estouro de campo, valor com sinal
0111	VC	$V = 0$	Não estouro de campo, valor com sinal
1000	HI	$C = 1 \wedge Z = 0$	Maior, valor sem sinal
1001	LS	$C = 0 \vee Z = 1$	Menor ou igual, valor sem sinal
1010	GE	$N = V$	Maior ou igual, valor com sinal
1011	LT	$N \neq V$	Menor, valor com sinal
1100	GT	$Z = 0 \wedge (N = V)$	Maior, valor com sinal
1101	LE	$Z = 1 \vee (N \neq V)$	Menor ou igual, valor com sinal
1110	AL	–	Sempre

Registradores de estado

- Os registradores de estado mantêm bits de estado que indicam o resultado de operações lógicas e aritméticas, controlam interrupções e o modo de operação do processador.
- Bits são divididos em bits de condição e bits de controle.



Bits de condição

- C: vai-um (*carry*). Ligado (ou seja, tem valor 1) se a operação causou vai-um (*carry-out*) ou empresta-um (*carry-in*), desligado caso contrário.
- Z: zero. Ligado se o resultado foi zero, desligado caso contrário.
- N: sinal. Cópia do bit mais significativo do resultado; considerando aritmética com sinal, se N igual a zero, o resultado é maior ou igual a zero. Se N igual a 1, resultado é negativo.
- V: estouro de campo (*overflow*), para operações com números com sinal em complemento de dois. Ligado se ocorreu estouro de campo, desligado caso contrário. Calculado como o ou-exclusivo entre o vai-um do bit mais significativo do resultado e o vai-um do segundo bit mais significativo do resultado.

Bits de controle

- I: interrupção. Quando I é igual a 1, interrupções do tipo IRQ estão desabilitadas.
- F: interrupção rápida. Quando F é igual a 1, interrupções do tipo FIQ estão desabilitadas.
- T: estado *Arm/Thumb*. O processador ARMv7 pode executar dois conjuntos de instruções: o conjunto normal, em que instruções têm 32 bits (denominado *arm*), e um conjunto reduzido (denominado *thumb*), em que instruções têm 16 bits, permitindo um código mais compacto. O bit de controle T reflete o estado em que o processador está operando; quando T é igual a 1, o processador está executando no estado *thumb*, quando T é igual a 0 o processador está executando no estado *arm*.
- M[4:0]: modo de operação. Determinam o modo de operação.

Modos de operação

M[4:0]	Modo de Operação
10000	*User*
10001	*FIQ*
10010	*IRQ*
10011	*Supervisor*
10111	*Abort*
11011	*Undefined*
11111	*System*

Instruções

- Codificadas em uma palavra de 32 bits.
- Praticamente todas executam em apenas um ciclo do relógio (em regime).
- Praticamente as instruções são executadas condicionalmente: quatro bits são usados para codificar uma condição.

Execução condicional

Código	Sufixo	Condição	Descrição
0000	EQ	$Z = 1$	Igual
0001	NE	$Z = 0$	Diferente
0010	CS	$C = 1$	Maior ou igual, valor sem sinal
0011	CC	$C = 0$	Menor, valor sem sinal
0100	MI	$N = 1$	Negativo
0101	PL	$N = 0$	Positivo ou zero
0110	VS	$V = 1$	Estouro de campo, valor com sinal
0111	VC	$V = 0$	Não estouro de campo, valor com sinal
1000	HI	$C = 1 \wedge Z = 0$	Maior, valor sem sinal
1001	LS	$C = 0 \vee Z = 1$	Menor ou igual, valor sem sinal
1010	GE	$N = V$	Maior ou igual, valor com sinal
1011	LT	$N \neq V$	Menor, valor com sinal
1100	GT	$Z = 0 \wedge (N = V)$	Maior, valor com sinal
1101	LE	$Z = 1 \vee (N \neq V)$	Menor ou igual, valor com sinal
1110	AL	–	Sempre

Execução condicional - Exemplos

Desvio (incodicional):

```
b longe
```

Desvio condicional:

```
beq longe
```

Cópia entre registradores:

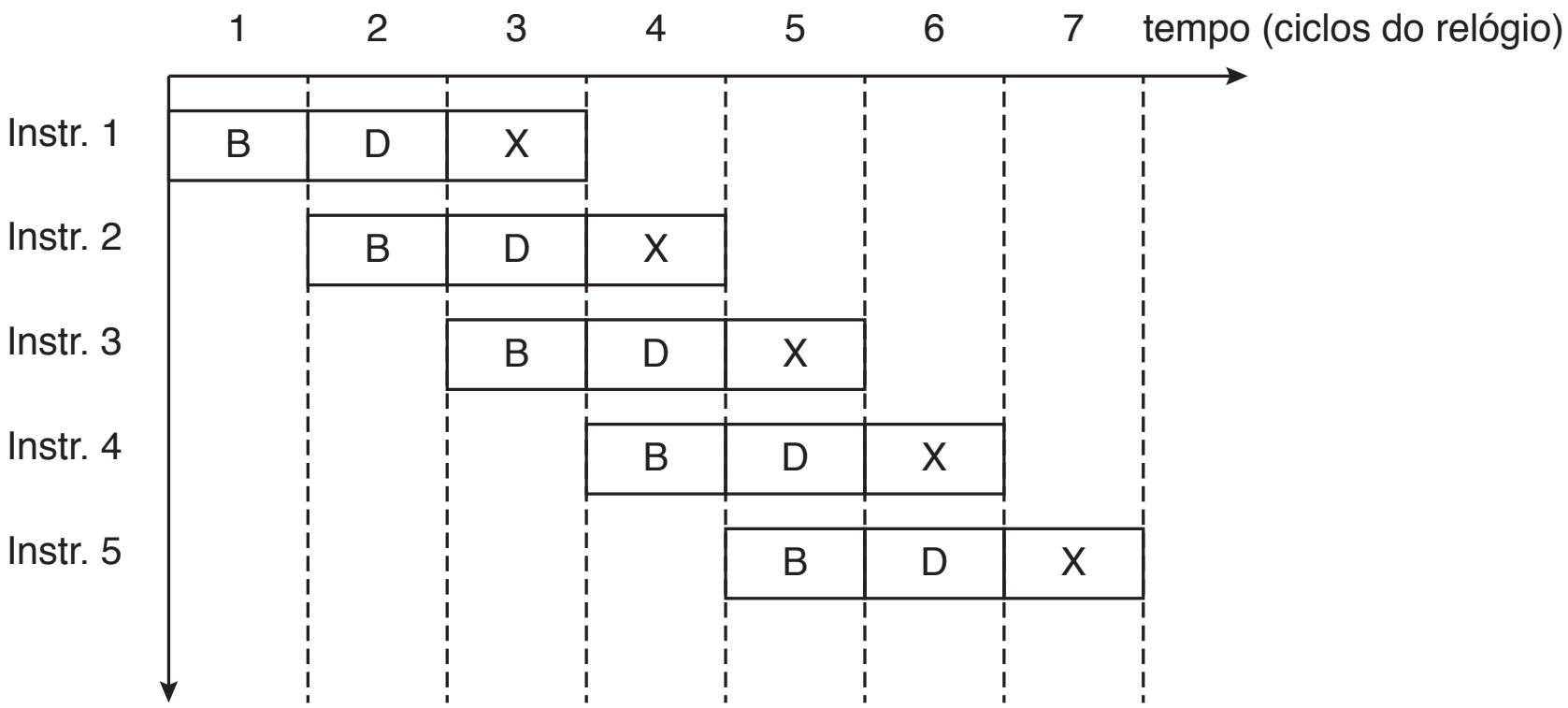
```
mov r1,r2
```

Execução condicional:

```
movne r1,r2
```

Pipeline

- A execução de uma instrução é dividida em *estágios*, cada um responsável por uma tarefa independente.
- Estágios são executados concorrentemente.



Arquitetura *Load/Store*

- Como o LEG, no ARM acessos à memória são feitos através de instruções específicas, como "carrega registrador de memória" e "armazena registrador de memória".
- Em todas as instruções de processamento de dados os operandos são registradores.
- Há processadores em que operandos de instruções aritméticas, por exemplo, podem ter um dos operandos em memória (e não em registradores).

Unidade de deslocamento

- Pode ser acionada para efetuar operações de deslocamento ou rotação em um operando *antes* de o operando ser usado em instruções aritméticas, lógicas ou de transferência de dados.
- Na maioria das instruções o deslocamento é feito dentro do mesmo ciclo de relógio em que a instrução é executada.
- Não há instruções específicas de deslocamento/rotação (podemos por exemplo usar uma instrução MOV, usando a unidade de deslocamento).

Convenções do montador GNU-ARM

- Similares ao montador `lasm`: .SKIP, .BYTE, .WORD, .ORG
- Um pouco diferente: .EQU nome, valor
- Nova diretiva: .ALIGN *expressao_inteira*
  altera o ponto de montagem para o menor inteiro maior do que o ponto de montagem e é divisível por $2^{expressao\_inteira}$.

```
1  @ definição de constante
2            .equ MAXVAL,256
3
4  @ alteração do ponto de montagem
5            .org   0x1000
6
7  @ reserva de espaço sem inicialização
8  vetor:    .skip  0x200          @ reserva 512 bytes
9  um:       .skip  1              @ reserva um byte
10
11 @ ponto de montagem aqui é 0x1201
12 @ alinha em palavra
13           .align 2 @ alinha em endereço múltiplo de 4 (2^2)
14 @ ponto de montagem aqui é 0x1204
15
16 @ reserva de espaço e inicialização
17 var1:     .byte  0x01           @ um byte, valor hexadecimal
18 var2:     .word  1000,2000      @ duas palavras (decimal, 32
19 mensagem: .ascii "uma linha\n" @ cadeia de caracteres
```