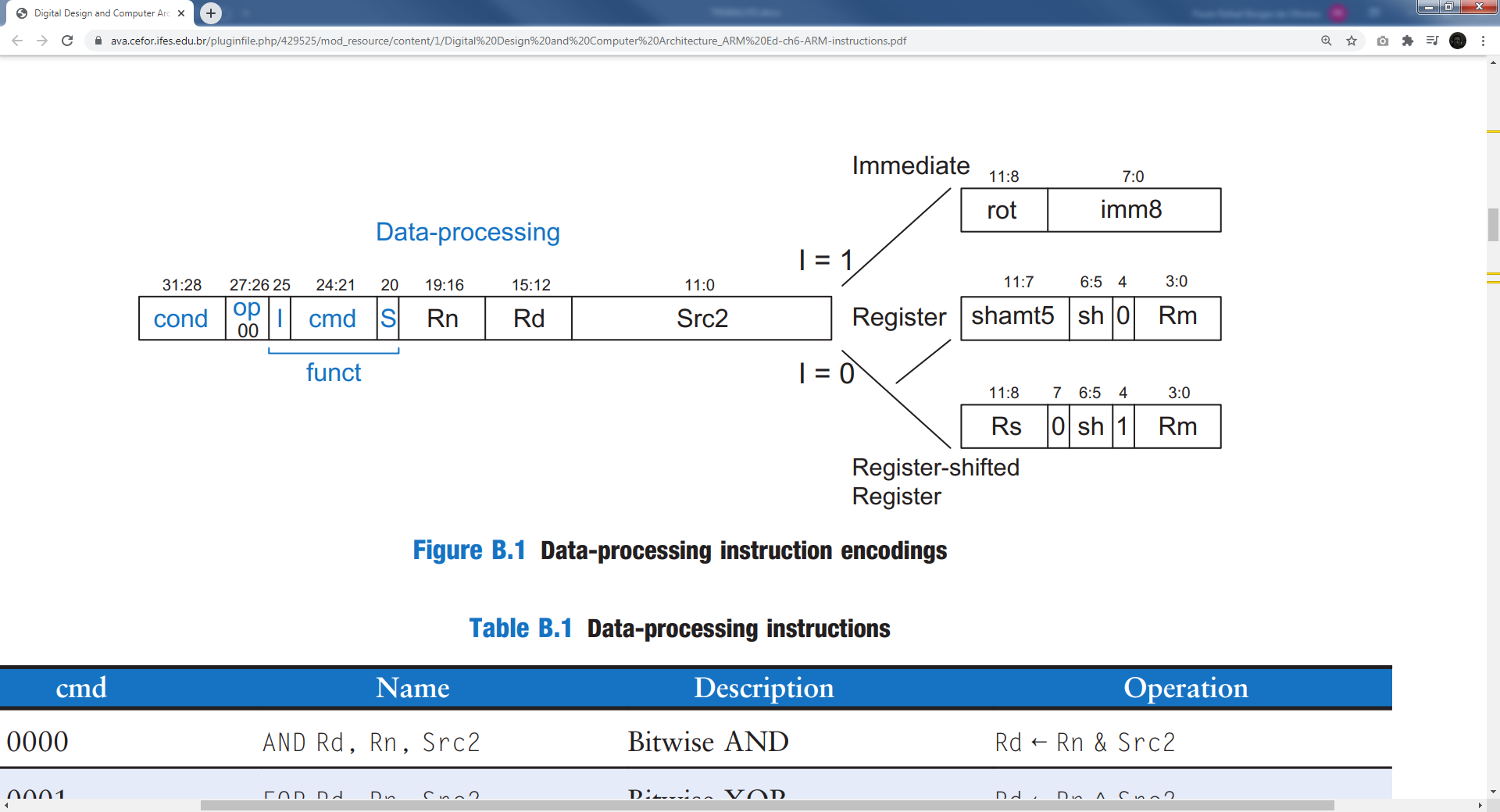
DATA

Para instruções do tipo **DATA-PROCESSING** temos a seguinte organização do frame:



Sendo:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| X | X | X | X | **0** | **0** | **1** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S | SHAMT5 | | | | | SH | | 0 | RM | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| X | X | X | X | **0** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S | RS | | | | 0 | SH | | 1 | RM | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| X | X | X | X | **0** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

COND:

OP

FUNCT

RN

RD

SRC2

O frame FUNCT se subdivide em:

I (25) usado para informar se o SRC vai ser um IMEDIATO (1) ou um REGISTRADOR (0);

S (20) usado para setar as flags: 1 – Setar e 2 – não setar;

Os bits 24:21 são os COMANDOS, totalizando 16 possibilidades:

A configuração padrão das operações básicas implementadas do tipo data apresentam-se conforme o seguinte:

AND

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| **1** | **1** | **1** | **0** | **0** | **0** | X | **0** | **0** | **0** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

SUB

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| **1** | **1** | **1** | **0** | **0** | **0** | X | **0** | **0** | **1** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

ADD

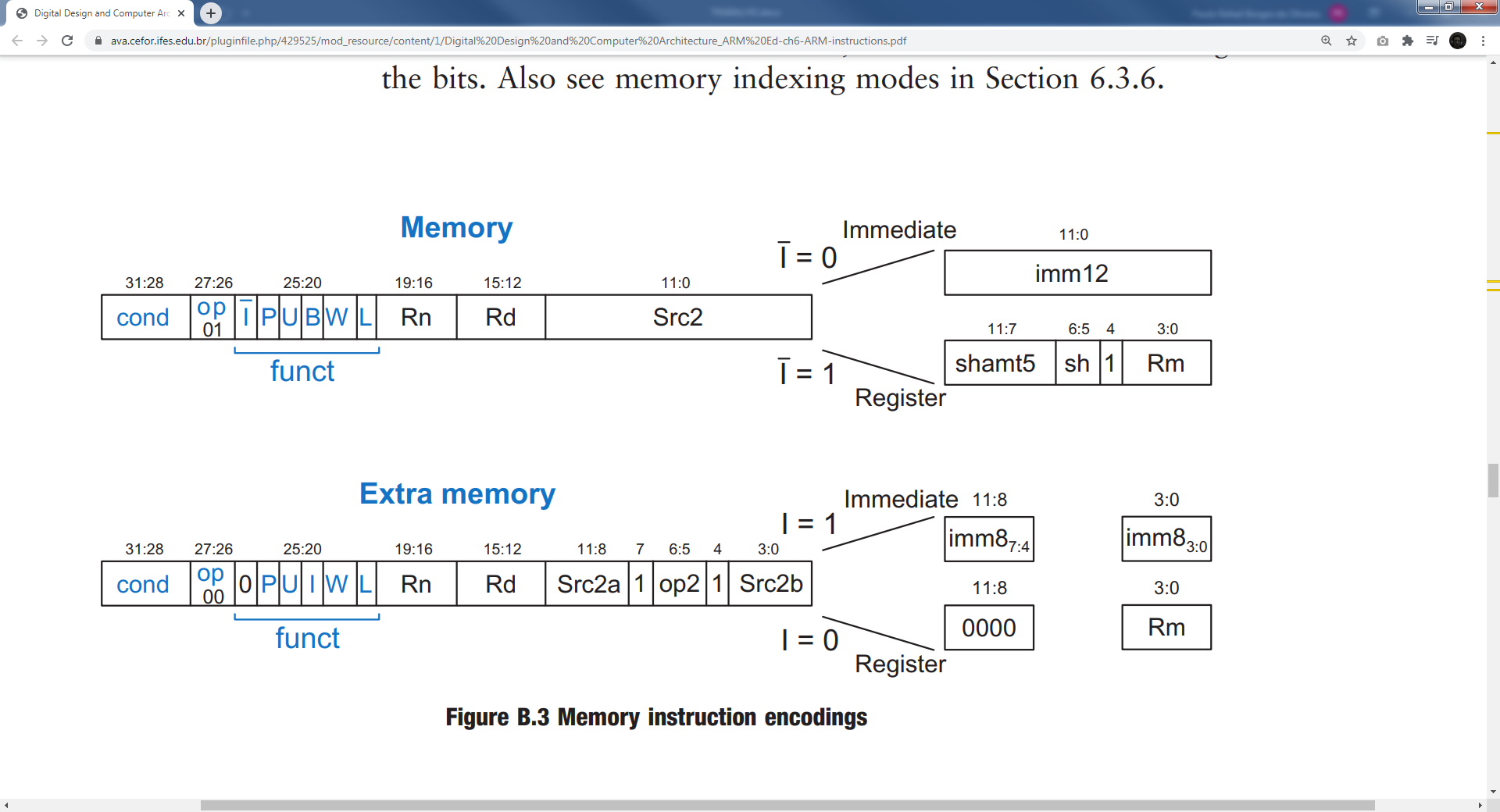
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| **1** | **1** | **1** | **0** | **0** | **0** | X | **0** | **1** | **0** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

ORR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| **1** | **1** | **1** | **0** | **0** | **0** | X | **1** | **1** | **0** | **0** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

MEMORY

Para instruções do tipo **MEMORY** temos a seguinte organização do frame:



Î – immediate barrado

P - Preindex  
U – Add (Adicionar ou subtrair)  
B – Byte (pegar o byte ou a palavra inteira)  
W – Writeback (Atualizar ou não o registrador)  
L – Load (Load ou Store)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| Î | P | U | B | W | L |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| X | X | X | X | **0** | **1** | **0** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| Î | P | U | B | W | L | SHAMT5 | | | | | SH | | 1 | RM | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| X | X | X | X | **0** | **1** | **1** | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

LDR

STR

BRANCH

B

Primeira Instrução:

ADD R2, R0, #5 ; R2 = 5

TRABALHO:

Implementar o seguintes comandos:

PRIMEIRO TESTE:

SUB R4, R15, R15 ; E04F400F // 1110.00.0.0010.0.1111.0100.000000001111

SUB R5, R4, R4 ;

ADD R4, R4, #10 // E284400A // 1110.00.1.0100.0.0100.0100.000000001010

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

ADD R5, R5, #5 // E2855005 // 1110.00.1.0100.0.0101.0101.000000000101

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

AND R6, R4, R5 // E0046005 // 1110.00.0.0000.0.0100.0110.000000000101

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

ORR R7, R4, R5 // E1847005 // 1110.00.0.1100.0.0100.0111.000000000101

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

SUB R8, R4, R5 // E0448005 // 1110.00.0.0010.0.0100.1000.000000000101

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

1) PRIMEIRA IMPLEMENTAÇÃO:

Em razão da limitação do tamanho dos bits que operacionalizam o comando da ULA torna-se necessário a ampliação do tamanho “variável” ALUControl de 3 bits para 4 bits para permitir que a ULA que possui as operações implementadas ADD, SUB, AND e OR possa incluir as operações EOR, TST, CMP, LSL, MOV E LDRB.

2) IMPLEMENTAÇÃO DE EOR

EOR [CMD: 0001, S=0] : Bitwise XOR – RD <- RN ^ Src2

EOR RD, RN, SRC2

A título de exemplo foi usado o seguinte comando:

EOR R9, R4, R5

Que possui representação em hexadecimal E0249005, sendo sua forma binária o seguinte:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | **0** | **0** | **0** | **1** | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Considerando que todos os elementos do frame já estão implementados (COND, OP, I, S, RN, RD e SRC2), torna-se necessário apenas a alteração no decoder para que a instrução 0001 seja reconhecida e envie comando para a ULA realizar a operação correspondente.

3) IMPLEMENTAÇÃO DO TST

TST [CMD: 1000, S=1] : Seta flags (N, Z, C) baseadas no RN & Src2 e não atualiza o registrador

TST RD, RN, SRC2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | **1** | **0** | **0** | **0** | **1** | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

1110.00.0.1000.1.0100.1010.000000000101

4) IMPLEMENTAÇÃO DO CMP

CMP [CMD: 1010, S=1] : Seta flags baseadas no RN - Src2

CMP RN, SRC2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | **1** | **0** | **1** | **0** | **1** | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Shifts:

LSL – Logical Shift Left – RD <- RM << SRC2

LSL RD, RM RS/SHAMT5

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COND | | | | OP | | FUNCT | | | | | | RN | | | | RD | | | | SRC2 | | | | | | | | | | | |
| I | CMD | | | | S |  | | | | IMM8 | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | **1** | **0** | **1** | **0** | **1** | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

//E1A0C384 - LSL R12, R4, #7 Register

1110.00.0.1101.0.0000.1100.00111.00.0.0100

//E1A0C714 - LSL R12, R4, R7 Register Shifted Register

1110.00.0.1101.0.0000.1100.0111.0.00.1.0100

DATA

MOV – MOVE – RD <- SRC2

MOV RD, SRC2

MEMORIA

LDRB – Load Byte