

Figura 1

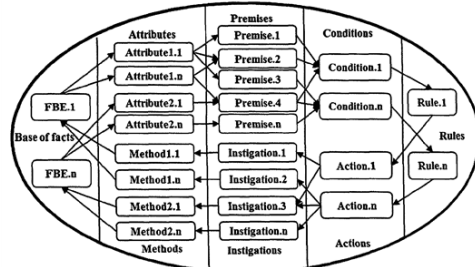


Figura 2

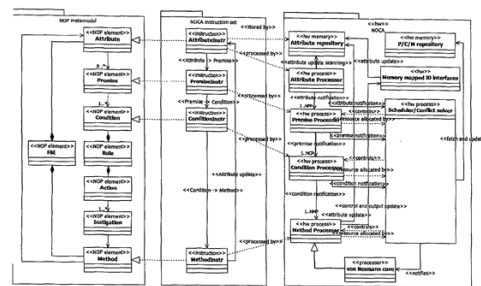


Figura 3

Mnemônico:
ATTRIBUTE-DECL, [E], [D],[N],[R], NP, Value, $[AP_0, AP_1, \dots, AP_{NP-1}]$

3	1	1	1	1	9	16	32	NP*32		
001	E	D	N	R	Reserved	NP	Value	AP ₀	AP ₁	AP _{NP-1}

Figura 4

Código	Mnemônico	Operação	Semântica (linguagem C)
0000	&&	AND	<i>opn1 && opn2</i>
0001	!(&&)	NAND	<i>!(opn1 && opn2)</i>
0010		OR	<i>opn1 opn2</i>
0011	!()	NOR	<i>!(opn1 opn2)</i>
0100	^	XOR	<i>(opn1 && !opn2) (!opn1 && opn2)</i>
0101	!&&	NOT first AND sec	<i>!opn1 && opn2</i>
0110	&!&	first AND NOT sec	<i>opn1 && !opn2</i>
0111	!	NOT first OR sec	<i>!opn1 opn2</i>
1000	!	first OR NOT sec	<i>opn1 !opn2</i>
1001	!	NOT	<i>!opn1</i>
1010	!!	BYPASS	<i>opn1</i>

Figura 8

Mnemônico:
METHOD-OP, [E], [D], [MM],[K],[A/V], Op, [IC=ITCOUNT],NDM, Add C/MM, A/V AT
opn1, A/V AT opn2,Add AT res,[ADM₀,ADM₁,...,ADM_{NDM-1}]

Codificação:									
2	1	1	1	1	4	2	5	13	2
10	E	D	MM	K	Reserved	A/V	Op	ITCOUNT	NDM

32	32	32	32	NDM*32		
Add C/MM	A/V AT opn1	A/V AT opn2	Add AT res	ADM ₀	ADM ₁	ADM _{NDM-1}

Figura 9

Código	Mnemonico	Operação	Semântica (linguagem C)
00000	=	Atribuição	$res = opm1$
00001	&	Bitwise AND	$res = opm1 \& opm2$
00010	&&	Bitwise NAND	$res = \neg(opm1 \& opm2)$
00011		Bitwise OR	$res = opm1 opm2$
00100	^	Bitwise XOR	$res = \neg opm1 \wedge opm2$
00101	^	Bitwise NOR	$res = \neg(opm1 \vee opm2) \vee opm1 \& opm2$
00110	^&	Bitwise XNOR	$res = \neg opm1 \wedge opm2 \vee opm1 \& opm2$
00111	~	Bitwise NOT	$res = \neg opm1$
01000	<<	Shift left	$res = opm1 << opm2$
01001	>>	Shift right	$res = opm1 >> opm2$
01010	+	Adição com sinal	$res = opm1 + opm2$
01011	-	Subtração com sinal	$res = opm1 - opm2$
01100	*%	Multiplicação com sinal (word mais significativa)	$res = (opm1 * opm2) >> 32$
01101	*	Multiplicação com sinal (word menos significativa)	$res = (opm1 * opm2) \& 0xFFFFFFFF$
01110	/	Divisão com sinal	$res = opm1 / opm2$
01111	CPARAV	Cópia de referência a partir de AT para valor de AT	$res = *opm1 + opm2$
10000	CPAVAR	Cópia de valor de AT para referência a partir de AT	$*res = opm1$
10001	CPARMEM	Cópia de referência a partir de AT para memória	$*Add(At\ res) = *opm1 + opm2$
10010	CPMEMAR	Cópia de memória para referência a partir de AT	$*res = *(At\ opm1)$
10011	CPARAR	Cópia de referência a partir de AT para referência a partir de AT	$*res = *opm1 + opm2$
10100	CPMEMAV	Cópia de memória para valor de AT	$res = *(At\ opm1)$

Mnemonic: PREMISE-OP, [E], [D], [R], [I], [Z], [A/V], Op, NC, A/V AT1, A/V AT2, [AC₀AC₁...AC_{N-1}]

Codeflag:

2	1	1	1	1	1	4	2	1	1	1	1	7	8
00	E	D	R	I	Z	Op	A/V	V1	V2	LR	INi	Reserved	NC

32				32				NC*32							
A/V AT1				A/V AT2				AC ₀				AC _{N-1}			

Figura 5

Código	Mnemônico	Operação	Semântica (linguagem C)
0000	==	Igualdade	<i>opn1 == opn2</i>
0001	!=	Desequaldade	<i>opn1 != opn2</i>
0010	>	Maior que	<i>opn1 > opn2</i>
0011	<	Menor que	<i>opn1 < opn2</i>
0100	>=	Maior ou igual a	<i>opn1 >= opn2</i>
0101	<=	Menor ou igual a	<i>opn1 <= opn2</i>
0110	&	Mascarado 1=0	<i>(opn1 & opn2) != 0</i>

Figura 6

Mnemônio:
 CONDITION-OP, [E], [D], [R], [S], *Prio*, *Op*, *NM*, *Add P/S1*, *Add P/S2*, *[AM₀AM₁...AM₃₁]*.
 Codificação:

2	1	1	1	1	1	4	2	1	1	1	1	1	3	8
01	E	D	R	S	Op	Prio	VP	P1	P2	LR	NP	US	Reserved	NM

32	32	NM*32
Add P/S1	Add P/S2	AM ₀ AM ₃₁₋₁

Figura 7

10101	CPAVMEM	Cópia de valor de AT para memória	$*(Add\ AT\ res) = opn1$
10110	SETMEM	Alteração de bits em memória para '1'	$*(Add\ AT\ res) = *(A/V\ AT\ opn1) \cdot opn2$
10111	CLRMEM	Alteração de bits em memória para '0'	$*(Add\ AT\ res) = *(A/V\ AT\ opn1) \cdot \sim opn2$
11000	INC	Incremento	$res = opn1 + 1$
11001	DEC	Decremento	$res = opn1 - 1$

Figura 10

Mnemônico:
METHOD-VN-OP, *Add MVN, Arg*

Codificação:		
2	30	32
11	Add MVN (30 bits MSB)	Arg

Figura 11

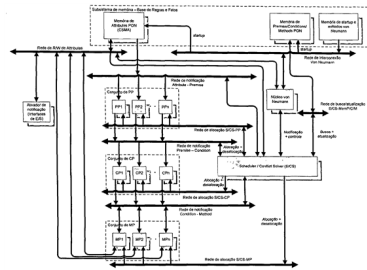


Figura 12

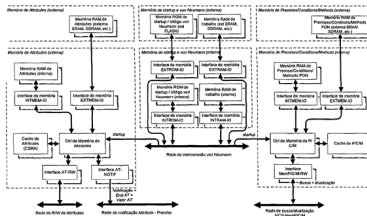


Figura 13

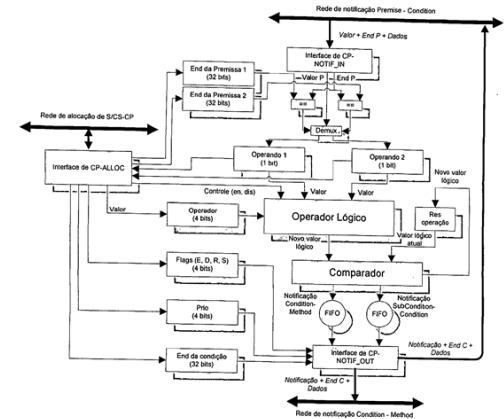


Figura 15

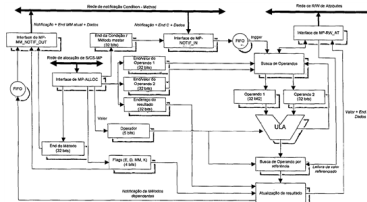


Figura 16



Figura 17

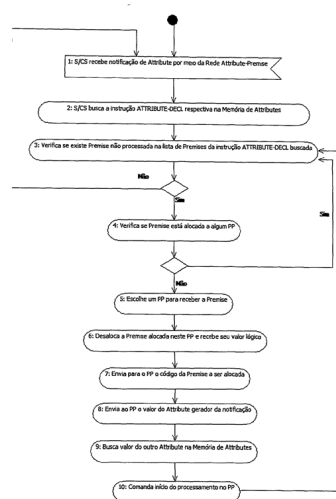


Figura 18

