

select	a	b	c	d	ram_ptr	5_h_cnt	6_zeros	7_input
	0	1	2	3	4	5	6	7
load	a	b	c	d	ram_ptr	F_cmp	G_out	H_cnt
	0	1	2	3	4	5	6	7
operation	add	sub	mult	div	cmp	and	or	not
	0	1	2	3	4	5	6	7

命令一覧

cmd	アセンブリ言語	命令	第一引数	第二引数
0	add a,im	add	a	im
1	sub a,im	sub	a	im
2	mult a,im	mult	a	im
3	div a,im	div	a	im
4	cmp a,im	cmp	a	im
5	and a,im	and	a	im
6	or a,im	or	a	im
7	mov a,not a	mov	a	not a
8	add b,im	add	b	im
9	sub b,im	sub	b	im
10	mult b,im	mult	b	im
11	div b,im	div	b	im
12	cmp b,im	cmp	b	im
13	and b,im	and	b	im
14	or b,im	or	b	im
15	mov b,not b	mov	b	not b
16	add c,im	add	c	im
17	sub c,im	sub	c	im
18	mult c,im	mult	c	im
19	div c,im	div	c	im
20	cmp c,im	cmp	c	im
21	and c,im	and	c	im
22	or c,im	or	c	im
23	mov c,not c	mov	c	not c
24	add d,im	add	d	im
25	sub d,im	sub	d	im
26	mult d,im	mult	d	im
27	div d,im	div	d	im
28	cmp d,im	cmp	d	im
29	and d,im	and	d	im
30	or d,im	or	d	im
31	mov d,not d	mov	d	not d
32				
33				

アセンブラ

ROM_ADDR	命令	第一引数	第二引数	label_a1	label_a2	key_a1	key_a2	検索key	命令番号	即値値	HEX	HEX			
0	:start							add a,im	0	0 00	00				0
1	mov	a	0	a	0	a	im	mov a,im	71	0 47	00				1
2	mov	b	0	b	0	b	im	mov b,im	79	0 4F	00				2
3	mov	c	0	c	0	c	im	mov c,im	87	0 57	00				3
4	mov	d	0	d	0	d	im	mov d,im	95	0 5F	00				4
5	mov	ptr	0	ptr	0	ptr	im	mov ptr,im	103	0 67	00				5
6	mov	d	1	d	1	d	im	mov d,im	95	1 5F	01				6
7	mov	xptr	d	xptr	d	xptr	d	mov xptr,c	151	0 97	00				7
8	//はじめ							add a,im	0	0 00	00				8
9	:main							add a,im	0	0 00	00				9
10	//dはloop1のカウンタで掛ける数							add a,im	0	0 00	00				10
11	mov	d	1	d	1	d	im	mov d,im	95	1 5F	01				11
12	:loop1start							add a,im	0	0 00	00				12
13	//cはloop2のカウンタでポインタ							add a,im	0	0 00	00				13
14	mov	c	0	c	0	c	im	mov c,im	87	0 57	00				14
15	:loop2start							add a,im	0	0 00	00				15
16	//かけて戻す							add a,im	0	0 00	00				16
17	mov	ptr	c	ptr	c	ptr	c	mov ptr,c	98	0 62	00				17
18	mov	a	xptr	a	xptr	a	xptr	mov a,xptr	144	0 90	00				18
19	mov	b	d	b	d	b	d	mov b,d	75	0 4B	00				19
20	mult	a	b	a	b	a	b	mult a,b	130	0 82	00				20
21	mov	xptr	a	xptr	a	xptr	a	mov xptr,a	148	0 94	00				21
22															22
23	add	c	1	c	1	c	im	add c,im	16	1 10	01				23
24	cmp	c	10	c	10	c	im	cmp c,im	20	10 14	0A				24
25	je	:loop2end			27		im	je im	207	27 CF	1B				25
26	jmp	:loop2start			15		im	jmp im	127	15 7F	0F				26
27	:loop2end							add a,im	0	0 00	00				27
28															28
29	//cはloop3のカウンタでポインタ							add a,im	0	0 00	00				29
30	mov	c	0	c	0	c	im	mov c,im	87	0 57	00				30
31	:loop3start							add a,im	0	0 00	00				31
32															32
33	mov	ptr	c	ptr	c	ptr	c	mov ptr,c	98	0 62	00				33

	34			
	35			
	36			
	37			
	38			
	39			
	40			
	41			
	42			
	43			
	44			
	45			
	46			
	47			
	48			
	49			
	50			
	51			
	52			
	53			
	54			
	55			
	56			
	57			
	58			
	59			
	60			
	61			
	62			
	63			
	64	mov a,a	mov a	a
	65	mov a,b	mov a	b
	66	mov a,c	mov a	c
	67	mov a,d	mov a	d
	68			
	69			
	70			
	71	mov a,im	mov a	im
	72	mov b,a	mov b	a
	73	mov b,b	mov b	b
	74	mov b,c	mov b	c
	75	mov b,d	mov b	d
	76			
	77			
	78			
	79	mov b,im	mov b	im
	80	mov c,a	mov c	a
	81	mov c,b	mov c	b

34	mov	a	xptr	a	xptr	a	xptr	mov a,xptr	144	0	90	00			34
35	cmp	a	10	a	10	a	im	cmp a,im	4	10	04	0A			35
36	jl	:loop3_add		70		im		jl im	215	70	D7	46			36
37															37
38	//b=int(a/10)							add a,im	0	0	00	00			38
39	mov	b	a	b	a	b	a	mov b,a	72	0	48	00			39
40	div	b	10	b	10	b	im	div b,im	11	10	0B	0A			40
41															41
42	//繰り上がりを退避							add a,im	0	0	00	00			42
43	mov	ptr	20	ptr	20	ptr	im	mov ptr,im	103	20	67	14			43
44	mov	xptr	b	xptr	b	xptr	b	mov xptr,b	149	0	95	00			44
45															45
46	//b=10*int(a/10)							add a,im	0	0	00	00			46
47	mult	b	10	b	10	b	im	mult b,im	10	10	0A	0A			47
48															48
49	//a=mod(a,10)							add a,im	0	0	00	00			49
50	sub	a	b	a	b	a	b	sub a,b	129	0	81	00			50
51															51
52	//arr[c]=mod(arr[c],10)							add a,im	0	0	00	00			52
53	mov	ptr	c	ptr	c	ptr	c	mov ptr,c	98	0	62	00			53
54	mov	xptr	a	xptr	a	xptr	a	mov xptr,a	148	0	94	00			54
55															55
56	//b=int(arr[c]/10)							add a,im	0	0	00	00			56
57	mov	ptr	20	ptr	20	ptr	im	mov ptr,im	103	20	67	14			57
58	mov	b	xptr	b	xptr	b	xptr	mov b,xptr	145	0	91	00			58
59															59
60	//ptr++							add a,im	0	0	00	00			60
61	add	c	1	c	1	c	im	add c,im	16	1	10	01			61
62	mov	ptr	c	ptr	c	ptr	c	mov ptr,c	98	0	62	00			62
63	sub	c	1	c	1	c	im	sub c,im	17	1	11	01			63
64															64
65	//arr[c+1]+=int(arr[c]/10)							add a,im	0	0	00	00			65
66	mov	a	xptr	a	xptr	a	xptr	mov a,xptr	144	0	90	00			66
67	add	a	b	a	b	a	b	add a,b	128	0	80	00			67
68	mov	xptr	a	xptr	a	xptr	a	mov xptr,a	148	0	94	00			68
69															69
70	:loop3_add							add a,im	0	0	00	00			70
71	add	c	1	c	1	c	im	add c,im	16	1	10	01			71
72	cmp	c	10	c	10	c	im	cmp c,im	20	10	14	0A			72
73	je	:loop3end		75		im		je im	207	75	CF	4B			73
74	jmp	:loop3start		31		im		jmp im	127	31	7F	1F			74
75	:loop3end							add a,im	0	0	00	00			75
76															76
77															77
78	add	d	1	d	1	d	im	add d,im	24	1	18	01			78
79	cmp	d	10	d	10	d	im	cmp d,im	28	10	1C	0A			79
80	je	:loop1end		82		im		je im	207	82	CF	52			80
81	jmp	:loop1start		12		im		jmp im	127	12	7F	0C			81

[illegible]