

# NANDだけで作る自作16bitCPU「NLP-16A」 ISA

## ■命令ビットフィールド

	15	12 11	8 7	4 3	0
	命令名				
	内部動作（太字は命令名の動作を参照）				
IR1	OP Code 4/2bit	ALU op code 4/6bit	Flag 4bit	Reg Address1(RA1)	
IR2	Reg Address2(RA2)	Reg Address3(RA3)	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				

命令は可変長で、IR1, 2, 3はそれぞれ1, 2, 3ワード目

## ■Flagフィールド 仕様

Flagフィールド設定				
実行条件	フラグが真で実行		フラグが偽で実行	
	サフィックス	値	サフィックス	値
NOP(No Operation)	.nop	0x0	無し	0x1
CarryもしくはBorrow	.c	0x9	.nc	0x8
演算結果がオーバーフロー	.v	0xB	.nv	0xA
演算結果が0	.z	0xD	.nz	0xC
演算結果が負	.s	0xF	.ns	0xE

0x1(サフィックス無し)は常に実行(通常)

## ■ALU op codeフィールド 仕様

演算機能	ALU 6bit opcode	ALU 4bit opcode	補足
ADD	0x0A	0x0A	
ADC	0x0E	0x0E	carry付
SUB	0x09	0x09	
SBB	0x0D	0x0D	borrow付
INC	0x1B		
INCC	0x1F		carry付
DEC	0x18		
DECB	0x1C		borrow付
AND	0x06		
OR	0x12		
NOT	0x14		
XOR	0x16		
SHL	0x20		論理シフト
SHR	0x30		
SAL	0x24		算術シフト
SAR	0x34		
ROL	0x22		ロール
ROR	0x32		
転送	0x00	0x00	

## ■Reg Addressフィールド 仕様

レジスタ種類	記号 アドレス			
汎用レジスタ	RegA 0x5	RegB 0x6	RegC 0x7	RegD 0x8
ポインタレジスタ	IP 0xD	SP 0xE	IV 0x2	
命令レジスタ	IR1 0x0	IR2 0x1	IR3 0x3	
特殊レジスタ	ZR 0xF	Addr 0xC	Flag 0x4	
MEM, MMIO	MEM 0xB			

グレイアウトは命令内で使用不可

IVは割込みベクタ  
IR2は8bit定数  
IR3は16bit定数  
ZRはゼロレジスタ  
Addrはアクセス不可  
MEM, I/Oアクセス用

## ■命令セット

命令名			内部動作 (太字は命令名の動作を参照) ALU opはALU op codeで演算機能を指定		
IR1	OP Code 4/2bit	ALU op code 4/6bit	Flag	Reg Address 1	
IR2	Reg Address 2	Reg Address 3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
演算系命令			RA2 (ALU op) RA3 ⇒ RA1		
IR1	0 0	ALU op code 6bit	Flag	RA1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
1word 演算命令			RA1 (ALU op) Acc ⇒ RA1		
IR1	0 1	ALU op code 6bit	Flag	RA1	
MOV命令(即値代入も含む)			RA2 ⇒ RA1		
IR1	0 0	0 0 0 0 0 0	Flag	RA1	
IR2	RA2	X X X X	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
JMP命令			RA2 (ALU op) RA3 ⇒ IP		
IR1	0 0	ALU op code 6bit	Flag	1 1 0 1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
PUSH命令			SP--, RA1 ⇒ MEM[SP]		
IR1	1 1 0 1	0 0 0 0	Flag	RA1	
POP命令			MEM[SP] ⇒ RA1, SP++		
IR1	1 1 0 0	0 0 0 0	Flag	RA1	
CALL命令			PUSH IP, RA2 (ALU op) RA3 ⇒ IP		
IR1	1 0 1 1	ALU op code 4bit	Flag	1 1 0 1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
RET命令			MEM[SP] ⇒ IP, SP++		
IR1	1 1 0 0	0 0 0 0	Flag	1 1 0 1	
LOAD命令			MEM[RA2 (ALU op) RA3] ⇒ RA1		
IR1	1 0 0 0	ALU op code 4bit	Flag	RA1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
STORE命令			RA1 ⇒ MEM[RA2 (ALU op) RA3]		
IR1	1 0 0 1	ALU op code 4bit	Flag	RA1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
CMP(比較)命令			RA2 >= RA3 ⇒ Flag		
IR1	0 X	0 0 1 0 0 1	Flag	1 1 1 1	
IR2	RA2	RA3	8bit 即値(RA2, 3どちらかで 0x01を指定時)		
IR3	16bit即値(RA2, 3どちらかで0x03を指定時)				
INT(ソフトウェア割り込み)命令			PUSH IP, PUSH Flag, JMP IV		
IR1	1 1 1 1	1 1 0 0	Flag	X X X X	
IRET(割り込みリターン)命令			POP Flag, RET		
IR1	1 1 1 0	0 0 0 0	Flag	X X X X	
IE(割り込み許可)命令			IE ⇒ INT mask Reg		
IR1	1 1 1 1	1 1 1 1	Flag	X X X X	
ID(割り込み禁止)命令			ID ⇒ INT mask Reg		
IR1	1 1 1 1	1 1 1 0	Flag	X X X X	