

**Reducerad
INSTRUKTIONSLISTA
för
FLIS-processorn**

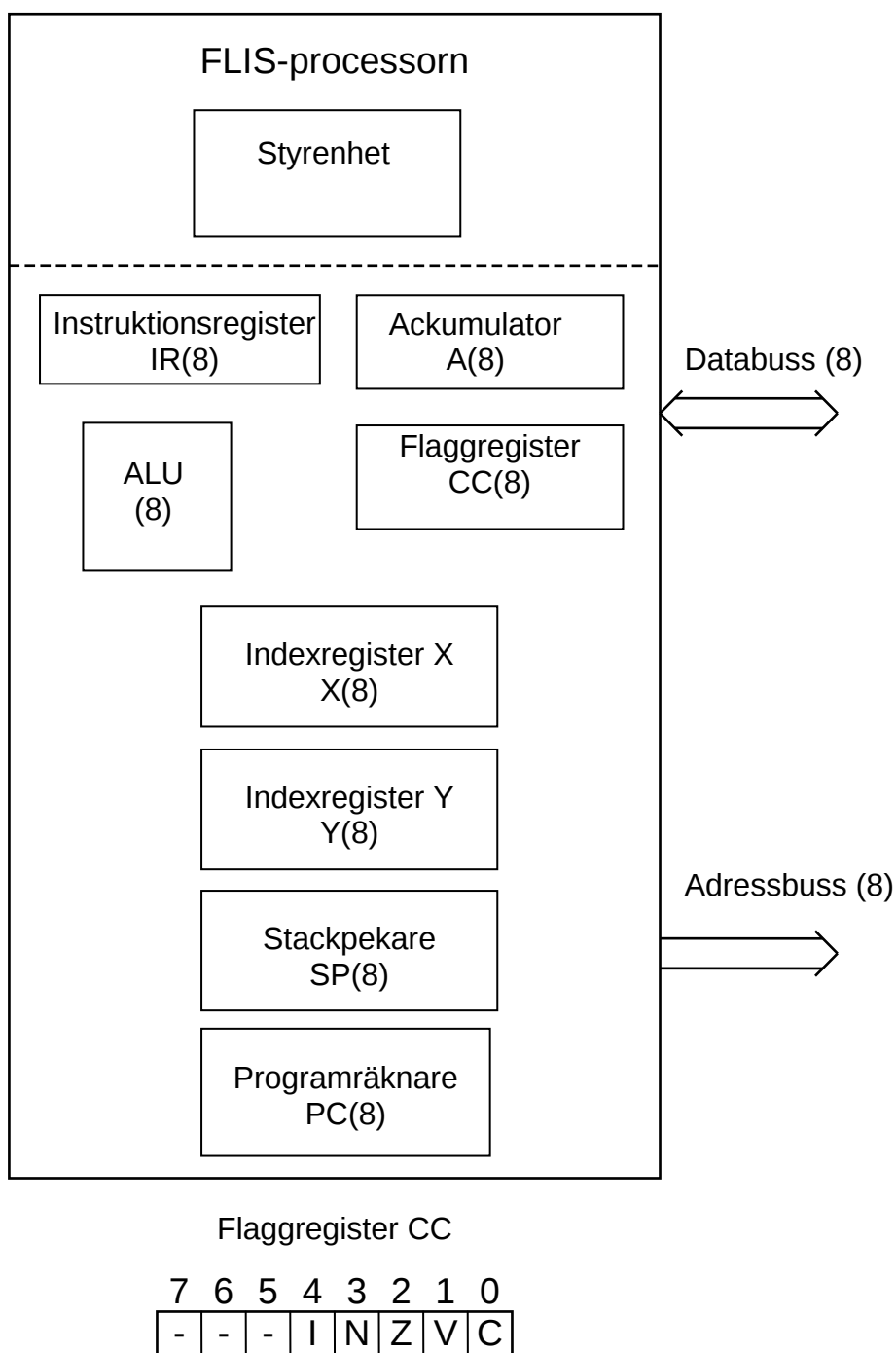
2013-11-08

Innehåll

Sidan

3	Programmerarens bild av FLIS-processorn
4	Förklaring av beteckningar i instruktionslistan
5	Enkel dataflyttning
5	Logik
5	No operation
6	Aritmetik
6	Skift och rotation
7	Test
7	Hopp (Jump)
7	Hopp eller Branch (förgrening) med PC-relativ adressering
8	Dataflyttning med adressering via X-, Y- eller SP-registret
9	Manipulering av X-, Y- eller SP-registrets innehåll
9	Hopp till subrutin samt återhopp från subrutin och avbrott
9	Branch till subrutin
9	Lagring av data på stack och hämtning av data från stack
10	Karta över samtliga operationskoder

Programmerarens bild av FLIS-processorn



På de följande sidorna beskrivs kortfattat maskininstruktionerna för FLIS-processorn.

För varje instruktion anges den mnemoniska beteckningen, operationskod (OP), antal bytes (#), antal klockcykler (~), operationsbeskrivning och flaggpåverkan.

Tal i form av data, adress eller avstånd (offset) kan uttryckas antingen hexadecimalt, binärt eller decimalt på följande sätt:

\$siffror = hexadecimalt
 % siffror = binärt
 siffror = decimalt

Avstånd (offset) är alltid ett tal med inbyggt tecken.

Förklaring av beteckningar i instruktionslistan:

OP	Hexadecimal operationskod för instruktion.
#	Antal bytes i instruktion (Används också för att beteckna adresseringsmoden "Immediate").
~	Antal klockcykler som krävs för att utföra en instruktion.
A	Innehåll i register A.
A _{1k}	Innehåll i register A komplementeras (inverteras) bitvis.
M(Adr)	Minnesinnehåll på adressen Adr.
M _{1k} (Adr)	Minnesinnehåll komplementeras (inverteras) bitvis.
A(0)<<1	Vänsterskift av register A med inskiftad nolla till höger.
M(Adr)(0)<<1	Vänsterskift av M(Adr) med inskiftad nolla till höger.
A(C)<<1	Vänsterskift av register A med inskiftad C-flagga till höger.
M(Adr)(C)<<1	Vänsterskift av M(Adr) med inskiftad C-flagga till höger.
(0)A>>1	Högerskift av register A med inskiftad nolla till vänster.
(0)M(Adr)>>1	Högerskift av M(Adr) med inskiftad nolla till vänster.
(C)A>>1	Högerskift av register A med inskiftad C-flagga till vänster.
(C)M(Adr)>>1	Högerskift av M(Adr) med inskiftad C-flagga till vänster.
(d ₇)A>>1	Högerskift av register A med kopierad d ₇ till vänster.
(d ₇)M(Adr)>>1	Högerskift av M(Adr) med kopierad d ₇ till vänster.
I	Avbrottsflaggan ("Interrupt"). Maskeringsbit för avbrott.
N	Teckenflaggan ("Negative").
Z	Nollflaggan ("Zero").
V	Overflowflaggan för 2-komplement-overflow.
C	Carryflaggan.
Δ	Används för att visa att en flagga påverkas.
-	Används för att visa att en flagga <u>ej</u> påverkas.
0	Nollställs. (Används oftast för att visa att en flagga nollställs).
1	Ettställs. (Används oftast för att visa att en flagga ettställs).
CC	"Condition Code" (Innehållet i flaggregistret, dvs samtliga flaggor).
n	Avstånd (offset). (Används i samband med adressering via ett indexregister. n är ett tal med inbyggt tecken.)
EA	Effektiv adress. För hopp- och branchinstruktioner avses adressen dit hoppet skall ske. För övriga instruktioner avses adressen till data.

Enkel dataflyttning (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion			Adressering									Operations- beskrivning*	Flaggor			
Operation	Beteckning	Inherent			Immediate			Absolute			3 N		2 Z	1 V	0 C	
		OP	#	~	OP	#	~	OP	#	~						
Transfer	TFR A,CC	18	1	2							A → CC	Δ	Δ	Δ	Δ	
	TFR CC,A	19	1	2							CC → A	-	-	-	-	
	TFR X,Y	1A	1	2							X → Y	-	-	-	-	
	TFR Y,X	1B	1	2							Y → X	-	-	-	-	
	TFR X,SP	1C	1	2							X → SP	-	-	-	-	
	TFR SP,X	1D	1	2							SP → X	-	-	-	-	
	TFR Y,SP	1E	1	2							Y → SP	-	-	-	-	
	TFR SP,Y	1F	1	2							SP → Y	-	-	-	-	
Exchange	EXG A,CC	9F	1	4							A ↔ CC	Δ	Δ	Δ	Δ	
	EXG X,Y	AF	1	4							X ↔ Y	-	-	-	-	
	EXG X,SP	BF	1	4							X ↔ SP	-	-	-	-	
	EXG Y,SP	CF	1	4							Y ↔ SP	-	-	-	-	
Load	LDA #Data				F0	2	2				Data → A	Δ	Δ	0	-	
	LDX #Data				90	2	2				Data → X	Δ	Δ	0	-	
	LDY #Data				91	2	2				Data → Y	Δ	Δ	0	-	
	LDSP #Data				92	2	2				Data → SP	Δ	Δ	0	-	
	LDA Adr							F1	2	3	M(Adr) → A	Δ	Δ	0	-	
	LDX Adr							A0	2	3	M(Adr) → X	Δ	Δ	0	-	
	LDY Adr							A1	2	3	M(Adr) → Y	Δ	Δ	0	-	
LDSP Adr							A2	2	3	M(Adr) → SP	Δ	Δ	0	-		
Store	STA Adr							E1	2	3	A → M(Adr)	-	-	-	-	
	STX Adr							30	2	3	X → M(Adr)	-	-	-	-	
	STY Adr							31	2	3	Y → M(Adr)	-	-	-	-	
	STSP Adr							32	2	3	SP → M(Adr)	-	-	-	-	

Logik (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering									Operations- beskrivning*	Flaggor				
Operation	Beteckning	Inherent			Immediate			Absolute				4	3	2	1	0
		OP	#	~	OP	#	~	OP	#	~		I	N	Z	V	C
AND	ANDA #Data				99	2	4				A AND Data → A	-	Δ	Δ	0	-
	ANDA Adr							A9	2	5	A AND M(Adr) → A	-	Δ	Δ	0	-
OR	ORA #Data				9A	2	4				A OR Data → A	-	Δ	Δ	0	-
	ORA Adr							AA	2	5	A OR M(Adr) → A	-	Δ	Δ	0	-
Exclusive-OR	EORA #Data				9B	2	4				A XOR Data → A	-	Δ	Δ	0	-
	EORA Adr							AB	2	5	A XOR M(Adr) → A	-	Δ	Δ	0	-
Complement	COMA	0A	1	3							A _{1k} → A	-	Δ	Δ	0	-
	COM Adr							3A	2	4	M _{1k} (Adr) → M(Adr)	-	Δ	Δ	0	-
Flag manipulation	ANDCC #Data				01	2	4				CC AND Data → CC	Δ	Δ	Δ	Δ	Δ
	ORCC #Data				02	2	4				CC OR Data → CC	Δ	Δ	Δ	Δ	Δ

No operation

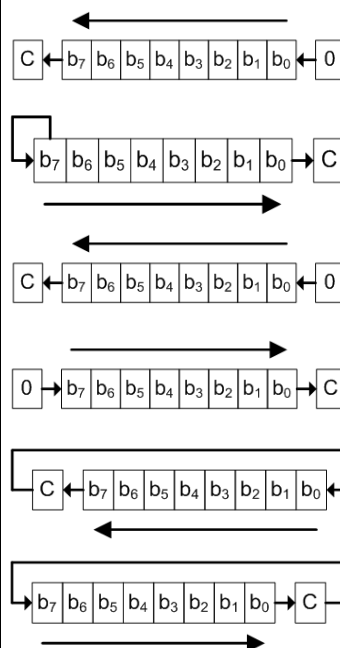
Instruktion		Adressering			Operations- beskrivning*	Flaggor				
Operation	Beteckning	Inherent					3	2	1	0
		OP	#	~			N	Z	V	C
No operation	NOP	00	1	2	No operation	-	-	-	-	

Aritmetik (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering									Operations- beskrivning*	Flaggor			
Operation	Beteckning	Inherent			Immediate			Absolute				3 N	2 Z	1 V	0 C
		OP	#	~	OP	#	~	OP	#	~					
Add	ADDA #Data				96	2	4				A+Data → A	Δ	Δ	Δ	Δ
	ADDA Adr							A6	2	5	A+M(Adr) → A	Δ	Δ	Δ	Δ
Add with carry	ADCA #Data				95	2	4				A+Data+C → A	Δ	Δ	Δ	Δ
	ADCA Adr							A5	2	5	A+M(Adr)+C → A	Δ	Δ	Δ	Δ
Subtract	SUBA #Data				94	2	4				A–Data → A	Δ	Δ	Δ	Δ
	SUBA Adr							A4	2	5	A–M(Adr) → A	Δ	Δ	Δ	Δ
Subtract with borrow	SBCA #Data				93	2	4				A–Data–C → A	Δ	Δ	Δ	Δ
	SBCA Adr							A3	2	5	A–M(Adr)–C → A	Δ	Δ	Δ	Δ
Negate (2's-compl)	NEGA	06	1	3							A _{1k} +1 → A	Δ	Δ	Δ	Δ
	NEG Adr							36	2	4	M _{1k} (Adr)+1 → M(Adr)	Δ	Δ	Δ	Δ
Increment	INCA	07	1	3							A+1→A	Δ	Δ	Δ	-
	INC Adr							37	2	4	M(Adr)+1 → M(Adr)	Δ	Δ	Δ	-
Decrement	DECA	08	1	3							A–1→A	Δ	Δ	Δ	-
	DEC Adr							38	2	4	M(Adr)–1 → M(Adr)	Δ	Δ	Δ	-
Clear	CLRA	05	1	3							0 → A	0	1	0	0
	CLR Adr							35	2	3	0 → M(Adr)	0	1	0	0

Skift och rotation (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering						Operations- beskrivning*	Flaggor			
Operation	Beteckning	Inherent			Absolute				3	2	1	0
		OP	#	~	OP	#	~					
Arithmetic shift left (ASL = LSL)	ASLA	0B	1	3				$A(0) \ll 1 \rightarrow A$	Δ	Δ	Δ	Δ
	ASL Adr				3B	2	4	$M(\text{Adr})(0) \ll 1 \rightarrow M(\text{Adr})$	Δ	Δ	Δ	Δ
Arithmetic shift right	ASRA	0F	1	3				$(d_7)A \gg 1 \rightarrow A$	Δ	Δ	Δ	Δ
	ASR Adr				3F	2	4	$(d_7)M(\text{Adr}) \gg 1 \rightarrow M(\text{Adr})$	Δ	Δ	Δ	Δ
Logic shift left (LSL = ASL)	LSLA	0B	1	3				$A(0) \ll 1 \rightarrow A$	Δ	Δ	Δ	Δ
	LSL Adr				3B	2	4	$M(\text{Adr})(0) \ll 1 \rightarrow M(\text{Adr})$	Δ	Δ	Δ	Δ
Logic shift right	LSRA	0C	1	3				$(0)A \gg 1 \rightarrow A$	0	Δ	Δ	Δ
	LSR Adr				3C	2	4	$(0)M(\text{Adr}) \gg 1 \rightarrow M(\text{Adr})$	0	Δ	Δ	Δ
Rotate left	ROLA	0D	1	3				$A(C) \ll 1 \rightarrow A$	Δ	Δ	Δ	Δ
	ROL Adr				3D	2	4	$M(\text{Adr})(C) \ll 1 \rightarrow M(\text{Adr})$	Δ	Δ	Δ	Δ
Rotate right	RORA	0E	1	3				$(C)A \gg 1 \rightarrow A$	Δ	Δ	Δ	Δ
	RORAdr				3E	2	4	$(C)M(\text{Adr}) \gg 1 \rightarrow M(\text{Adr})$	Δ	Δ	Δ	Δ



Test (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering									Operationsbeskrivning*	Flaggor				
Operation	Beteckning	Inherent			Immediate			Absolute					3	2	1	0
		OP	#	~	OP	#	~	OP	#	~			N	Z	V	C
Compare	CMPA #Data				97	2	3				A-Data	Δ	Δ	Δ	Δ	
	CMPX #Data				9C	2	3				X-Data	Δ	Δ	Δ	Δ	
	CMPY #Data				9D	2	3				Y-Data	Δ	Δ	Δ	Δ	
	CMPSP #Data				9E	2	3				SP-Data	Δ	Δ	Δ	Δ	
	CMPA Adr							A7	2	4	A-M(Adr)	Δ	Δ	Δ	Δ	
	CMPX Adr							AC	2	4	X-M(Adr)	Δ	Δ	Δ	Δ	
	CMPY Adr							AD	2	4	Y-M(Adr)	Δ	Δ	Δ	Δ	
	CMPSP Adr							AE	2	4	SP-M(Adr)	Δ	Δ	Δ	Δ	
Test, zero or minus	TSTA	09	1	2							A-0	Δ	Δ	0	0	
	TST Adr							39	2	3	M(Adr)-0	Δ	Δ	0	0	
Bit test	BITA #Data				98	2	3				A AND Data	Δ	Δ	0	-	
	BITA Adr							A8	2	4	A AND M(Adr)	Δ	Δ	0	-	

Hopp (Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering									Operationsbeskrivning*	Flaggor			
Operation	Beteckning	Inherent			Immediate			Absolute				3	2	1	0
		OP	#	~	OP	#	~	OP	#	~					
Unconditional jump	JMP Adr							33	2	2	Adr → PC	-	-	-	-

Hopp eller Branch (Förgrening) med PC-relativ adressering

Instruktion			Adressering			Operationsbeskrivning*	Flaggor			
Operation	Beteckning	Relative			3 N		2 Z	1 V	0 C	
		OP	#	~						
Unconditional branch	BRA Adr	21	2	4	PC+Offs → PC		-	-	-	-
Conditional branch										
Simple conditions	BMI Adr	22	2	4	If N = 1:	PC+Offs → PC	-	-	-	-
	BPL Adr	23	2	4	If N = 0:	PC+Offs → PC	-	-	-	-
	BEQ Adr	24	2	4	If Z = 1:	PC+Offs → PC	-	-	-	-
	BNE Adr	25	2	4	If Z = 0:	PC+Offs → PC	-	-	-	-
	BVS Adr	26	2	4	If V = 1:	PC+Offs → PC	-	-	-	-
	BVC Adr	27	2	4	If V = 0:	PC+Offs → PC	-	-	-	-
(BCS = BLO)	BCS Adr	28	2	4	If C = 1:	PC+Offs → PC	-	-	-	-
(BCC = BHS)	BCC Adr	29	2	4	If C = 0:	PC+Offs → PC	-	-	-	-
Unsigned numbers	BHI Adr	2A	2	4	If C'Z' = 1:	PC+Offs → PC	-	-	-	-
(BHS = BCC)	BHS Adr	29	2	4	If C = 0:	PC+Offs → PC	-	-	-	-
	BEQ Adr	24	2	4	If Z = 1:	PC+Offs → PC	-	-	-	-
	BNE Adr	25	2	4	If Z = 0:	PC+Offs → PC	-	-	-	-
	BLS Adr	2B	2	4	If C+Z = 1:	PC+Offs → PC	-	-	-	-
(BLO = BCS)	BLO Adr	28	2	4	If C = 1:	PC+Offs → PC	-	-	-	-
Signed numbers	BGT Adr	2C	2	4	If (N⊕V) + Z = 0:	PC+Offs → PC	-	-	-	-
	BGE Adr	2D	2	4	If (N⊕V) = 0:	PC+Offs → PC	-	-	-	-
	BEQ Adr	24	2	4	If Z = 1:	PC+Offs → PC	-	-	-	-
	BNE Adr	25	2	4	If Z = 0:	PC+Offs → PC	-	-	-	-
	BLE Adr	2E	2	4	If (N⊕V) + Z = 1:	PC+Offs → PC	-	-	-	-
	BLT Adr	2F	2	4	If (N⊕V) = 1:	PC+Offs → PC	-	-	-	-

Dataflyttning med adressering via X-, Y- eller SP-registret

Instruktion		Adressering			Operations- beskrivning*	Flaggor			
Operation	Beteckning	Indexerat				3 N	2 Z	1 V	0 C
		OP	#	~					
Load (indexed)	LDA n,SP	F2	2	3	$M(n+SP) \rightarrow A$	Δ	Δ	0	-
	LDA n,X	F3	2	3	$M(n+X) \rightarrow A$	Δ	Δ	0	-
	LDA A,X	F4	1	3	$M(A+X) \rightarrow A$	Δ	Δ	0	-
	LDA ,X+	F5	1	4	$M(X) \rightarrow A, X+1 \rightarrow X$	Δ	Δ	0	-
	LDA ,X-	F6	1	4	$M(X) \rightarrow A, X-1 \rightarrow X$	Δ	Δ	0	-
	LDA ,+X	F7	1	4	$X+1 \rightarrow X, M(X) \rightarrow A$	Δ	Δ	0	-
	LDA ,-X	F8	1	4	$X-1 \rightarrow X, M(X) \rightarrow A$	Δ	Δ	0	-
	LDA n,Y	F9	2	3	$M(n+Y) \rightarrow A$	Δ	Δ	0	-
	LDA A,Y	FA	1	3	$M(A+Y) \rightarrow A$	Δ	Δ	0	-
	LDA ,Y+	FB	1	4	$M(Y) \rightarrow A, Y+1 \rightarrow Y$	Δ	Δ	0	-
	LDA ,Y-	FC	1	4	$M(Y) \rightarrow A, Y-1 \rightarrow Y$	Δ	Δ	0	-
	LDA ,+Y	FD	1	4	$Y+1 \rightarrow Y, M(Y) \rightarrow A$	Δ	Δ	0	-
	LDA ,-Y	FE	1	4	$Y-1 \rightarrow Y, M(Y) \rightarrow A$	Δ	Δ	0	-
	LDX n,SP	B0	2	3	$M(n+SP) \rightarrow X$	Δ	Δ	0	-
	LDX n,X	C0	2	3	$M(n+X) \rightarrow X$	Δ	Δ	0	-
	LDX n,Y	D0	2	3	$M(n+Y) \rightarrow X$	Δ	Δ	0	-
	LDY n,SP	B1	2	3	$M(n+SP) \rightarrow Y$	Δ	Δ	0	-
	LDY n,X	C1	2	3	$M(n+X) \rightarrow Y$	Δ	Δ	0	-
	LDY n,Y	D1	2	3	$M(n+Y) \rightarrow Y$	Δ	Δ	0	-
	LDSP n,SP	B2	2	3	$M(n+SP) \rightarrow SP$	Δ	Δ	0	-
	LDSP n,X	C2	2	3	$M(n+X) \rightarrow SP$	Δ	Δ	0	-
	LDSP n,Y	D2	2	3	$M(n+Y) \rightarrow SP$	Δ	Δ	0	-
Store (indexed)	STA n,SP	E2	2	3	$A \rightarrow M(n+SP)$	-	-	-	-
	STA n,X	E3	2	3	$A \rightarrow M(n+X)$	-	-	-	-
	STA A,X	E4	1	3	$A \rightarrow M(A+X)$	-	-	-	-
	STA ,X+	E5	1	4	$A \rightarrow M(X), X+1 \rightarrow X$	-	-	-	-
	STA ,X-	E6	1	4	$A \rightarrow M(X), X-1 \rightarrow X$	-	-	-	-
	STA ,+X	E7	1	4	$X+1 \rightarrow X, A \rightarrow M(X)$	-	-	-	-
	STA ,-X	E8	1	4	$X-1 \rightarrow X, A \rightarrow M(X)$	-	-	-	-
	STA n,Y	E9	2	3	$A \rightarrow M(n+Y)$	-	-	-	-
	STA A,Y	EA	1	3	$A \rightarrow M(A+Y)$	-	-	-	-
	STA ,Y+	EB	1	4	$A \rightarrow M(Y), Y+1 \rightarrow Y$	-	-	-	-
	STA ,Y-	EC	1	4	$A \rightarrow M(Y), Y-1 \rightarrow Y$	-	-	-	-
	STA ,+Y	ED	1	4	$Y+1 \rightarrow Y, A \rightarrow M(Y)$	-	-	-	-
	STA ,-Y	EE	1	4	$Y-1 \rightarrow Y, A \rightarrow M(Y)$	-	-	-	-
	STX n,SP	40	2	3	$X \rightarrow M(n+SP)$	-	-	-	-
	STX n,X	50	2	3	$X \rightarrow M(n+X)$	-	-	-	-
	STX A,X	60	1	3	$X \rightarrow M(A+X)$	-	-	-	-
	STX n,Y	70	2	3	$X \rightarrow M(n+Y)$	-	-	-	-
	STX A,Y	80	1	3	$X \rightarrow M(A+Y)$	-	-	-	-
	STY n,SP	41	2	3	$Y \rightarrow M(n+SP)$	-	-	-	-
	STY n,X	51	2	3	$Y \rightarrow M(n+X)$	-	-	-	-
	STY A,X	61	1	3	$Y \rightarrow M(A+X)$	-	-	-	-
	STY n,Y	71	2	3	$Y \rightarrow M(n+Y)$	-	-	-	-
	STY A,Y	81	1	3	$Y \rightarrow M(A+Y)$	-	-	-	-
	STSP n,SP	42	2	3	$SP \rightarrow M(n+SP)$	-	-	-	-
	STSP n,X	52	2	3	$SP \rightarrow M(n+X)$	-	-	-	-
	STSP A,X	62	1	3	$SP \rightarrow M(A+X)$	-	-	-	-
	STSP n,Y	72	2	3	$SP \rightarrow M(n+Y)$	-	-	-	-
	STSP A,Y	82	1	3	$SP \rightarrow M(A+Y)$	-	-	-	-

Manipulering av X-, Y- eller SP-registrets innehåll

Instruktion		Adressering			Operations- beskrivning*	Flaggor			
Operation	Beteckning	Via X, Y eller SP				3	2	1	0
		OP	#	~		N	Z	V	C
Load effective address	LEAX n,X	CC	2	4	$X + n \rightarrow X$	-	-	-	-
	LEAX n,SP	DC	2	4	$SP + n \rightarrow X$	-	-	-	-
	LEAY n,Y	CD	2	4	$Y + n \rightarrow Y$	-	-	-	-
	LEAY n,SP	DD	2	4	$SP + n \rightarrow Y$	-	-	-	-
	LEASP n,SP	BE	2	4	$SP + n \rightarrow SP$	-	-	-	-
	LEASP n,X	CE	2	4	$X + n \rightarrow SP$	-	-	-	-
	LEASP n,Y	DE	2	4	$Y + n \rightarrow SP$	-	-	-	-

Hopp till subrutin samt återhopp från subrutin och avbrott

(Se den fullständiga instruktionslistan för adressering via index.)

Instruktion		Adressering						Operations- beskrivning*	Flaggor				
Operation	Beteckning	Inherent			Absolute					4	3	2	1
		OP	#	~	OP	#	~	I		N	Z	V	C
Jump to subroutine	JSR Adr				34	2	4	SP-1 → SP, PC → M(SP) Adr → PC	-	-	-	-	-
Return from subroutine	RTS	43	1	2				M(SP) → PC, SP+1 → SP	-	-	-	-	-
Return from interrupt	RTI	44	1	6				M(SP) → CC; SP+1 → SP	Δ	Δ	Δ	Δ	Δ
								M(SP) → A; SP+1 → SP					
								M(SP) → X; SP+1 → SP					
								M(SP) → Y; SP+1 → SP					
								M(SP) → PC; SP+1 → SP					

Branch till subrutin

Instruktion		Adressering			Operationsbeskrivning*	Flaggor			
Operation	Beteckning	Relative				3	2	1	0
		OP	#	~					
Branch to subroutine	BSR Adr	20	2	5	SP-1 → SP, PC → M(SP) PC+Offs → PC	-	-	-	-

Lagring av data på stack och hämtning av data från stack

Instruktion		Adressering			Operationsbeskrivning*	Flaggor			
Operation	Beteckning	Inherent				3	2	1	0
		OP	#	~		N	Z	V	C
Push accumulator A	PSHA	10	1	3	SP-1 → SP, A → M(SP)	-	-	-	-
Push register X	PSHX	11	1	3	SP-1 → SP, X → M(SP)	-	-	-	-
Push register Y	PSHY	12	1	3	SP-1 → SP, Y → M(SP)	-	-	-	-
Push register CC	PSHC	13	1	3	SP-1 → SP, CC → M(SP)	-	-	-	-
Pull accumulator A	PULA	14	1	3	M(SP) → A, SP+1 → SP	-	-	-	-
Pull register X	PULX	15	1	3	M(SP) → X, SP+1 → SP	-	-	-	-
Pull register Y	PULY	16	1	3	M(SP) → Y, SP+1 → SP	-	-	-	-
Pull register CC	PULC	17	1	3	M(SP) → CC, SP+1 → SP	Δ	Δ	Δ	Δ

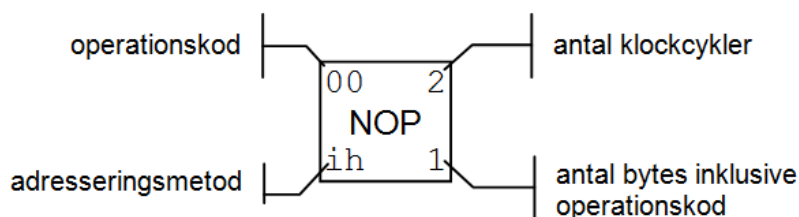
*Tillägg till operationsbeskrivning:

Lägg märke till att varje instruktion dessutom ökar innehållet i PC.

Alla instruktioner ökar PC med ett under FETCH-sekvensen.

Instruktioner som består av två ord ökar sedan PC med ytterligare ett i samband med att ord nummer två hämtas under EXECUTE-sekvensen. Ökningarna av PC görs innan offset adderas för branchinstruktioner och innan PC-värdet lagras som återhopsadress på stacken vid subrutinanrop.

Karta över (samtliga) operationskoder



Anm:

“Blanka” fält utgör ”otillåten operationskod” och medför undantagshantering

00	2	10	3	20	5	30	3	40	3	50	3	60	3	70	3	80	3	90	2	A0	3	B0	3	C0	3	D0	3	E0		F0	2
NOP	ih	1	PSHA	1	BSR	2	STX	2	STX	2	STX	2	STX	1	STX	2	STX	1	LDX	2	LDX	2	LDX	2	LDX	2			LDA	2	
01	4	11	3	21	4	31	3	41	3	51	3	61	3	71	3	81	3	91	2	A1	3	B1	3	C1	3	D1	3	E1	3	F1	3
ANDCC	im	2	PSHX	1	BRA	2	STY	2	STY	2	STY	2	STY	1	STY	2	STY	1	LDY	2	LDY	2	LDY	2	LDY	2	STA	2	LDA	2	
02	4	12	3	22	4	32	3	42	3	52	3	62	3	72	3	82	3	92	2	A2	3	B2	3	C2	3	D2	3	E2	3	F2	3
ORCC	im	2	PSHY	1	BMI	2	STSP	2	STSP	2	STSP	2	STSP	1	STSP	2	STSP	1	LDSP	2	LDSP	2	LDSP	2	LDSP	2	STA	2	LDA	2	
03		13	3	23	4	33	2	43	2	53	4	63	4	73	4	83	4	93	4	A3	5	B3	5	C3	5	D3	5	E3	3	F3	3
	ih	1	PSHC	1	BPL	2	JMP	2	RTS	1	JMP	2	JMP	1	JMP	2	JMP	1	SBCA	2	SBCA	2	SBCA	2	SBCA	2	STA	2	LDA	2	
04		14	3	24	4	34	4	44	6	54	5	64	5	74	5	84	5	94	4	A4	5	B4	5	C4	5	D4	5	E4	3	F4	3
	ih	1	PULA	1	BEQ	2	JSR	2	RTI	1	JSR	2	JSR	1	JSR	2	JSR	1	SUBA	2	SUBA	2	SUBA	2	SUBA	2	STA	1	LDA	1	
05	3	15	3	25	4	35	3	45	3	55	3	65	3	75	3	85	3	95	4	A5	5	B5	5	C5	x	D5	5	E5	4	F5	4
CLRA	ih	1	PULX	1	BNE	2	CLR	2	CLR	2	CLR	2	CLR	1	CLR	2	CLR	1	ADCA	2	ADCA	2	ADCA	2	ADCA	2	STA	1	LDA	1	
06	3	16	3	26	4	36	4	46	4	56	4	66	4	76	4	86	4	96	4	A6	5	B6	5	C6	5	D6	5	E6	4	F6	4
NEGA	ih	1	PULY	1	BVS	2	NEG	2	NEG	2	NEG	2	NEG	1	NEG	2	NEG	1	ADDA	2	ADDA	2	ADDA	2	ADDA	2	STA	1	LDA	1	
07	3	17	3	27	4	37	4	47	4	57	4	67	4	77	4	87	4	97	3	A7	4	B7	4	C7	4	D7	4	E7	4	F7	4
INCA	ih	1	PULC	1	BVC	2	INC	2	INC	2	INC	2	INC	1	INC	2	INC	1	CMPA	2	CMPA	2	CMPA	2	CMPA	2	STA	1	LDA	1	
08	3	18	2	28	4	38	4	48	4	58	4	68	4	78	4	88	4	98	3	A8	4	B8	4	C8	4	D8	4	E8	4	F8	4
DECA	ih	1	TFR A,C	1	BCS	2	DEC	2	DEC	2	DEC	2	DEC	1	DEC	2	DEC	1	BITA	2	BITA	2	BITA	2	BITA	2	STA	1	LDA	1	
09	2	19	2	29	4	39	3	49	3	59	3	69	3	79	3	89	3	99	4	A9	5	B9	5	C9	5	D9	5	E9	3	F9	3
TSTA	ih	1	TFR C,A	1	BCC	2	TST	2	TST	2	TST	2	TST	1	TST	2	TST	1	ANDA	2	ANDA	2	ANDA	2	ANDA	2	STA	2	LDA	2	
0A	3	1A	2	2A	4	3A	4	4A	4	5A	4	6A	4	7A	4	8A	4	9A	4	AA	5	BA	5	CA	5	DA	5	EA	3	FA	3
COMA	ih	1	TFR X,Y	1	BHI	2	COM	2	COM	2	COM	2	COM	1	COM	2	COM	1	ORA	2	ORA	2	ORA	2	ORA	2	STA	1	LDA	1	
0B	3	1B	2	2B	4	3B	4	4B	4	5B	4	6B	4	7B	4	8B	4	9B	4	AB	5	BB	5	CB	5	DB	5	EB	4	FB	4
LSLA	ih	1	TFR Y,X	1	BLS	2	LSL	2	LSL	2	LSL	2	LSL	1	LSL	2	LSL	1	EORA	2	EORA	2	EORA	2	EORA	2	STA	1	LDA	1	
0C	3	1C	2	2C	4	3C	4	4C	4	5C	4	6C	4	7C	4	8C	4	9C	3	AC	4	BC	4	CC	4	DC	4	EC	4	FC	4
LSRA	ih	1	TFR X,S	1	BGT	2	LSR	2	LSR	2	LSR	2	LSR	1	LSR	2	LSR	1	CMPX	2	CMPX	2	CMPX	2	LEAX	2	STA	1	LDA	1	
0D	3	1D	2	2D	4	3D	4	4D	4	5D	4	6D	4	7D	4	8D	4	9D	3	AD	4	BD	4	CD	4	DD	4	ED	4	FD	4
ROLA	ih	1	TFR S,X	1	BGE	2	ROL	2	ROL	2	ROL	2	ROL	1	ROL	2	ROL	1	CMPLY	2	CMPLY	2	CMPLY	2	LEAY	2	STA	1	LDA	1	
0E	3	1E	2	2E	4	3E	4	4E	4	5E	4	6E	4	7E	4	8E	4	9E	3	AE	4	BE	4	CE	4	DE	4	EE	4	FE	4
RORA	ih	1	TFR Y,S	1	BLE	2	ROR	2	ROR	2	ROR	2	ROR	1	ROR	2	ROR	1	CMPSP	2	CMPSP	2	LEASP	2	LEASP	2	STA	1	LDA	1	
0F	3	1F	2	2F	4	3F	4	4F	4	5F	4	6F	4	7F	4	8F	4	9F	4	AF	4	BF	4	CF	4	DF		EF		FF	
ASRA	ih	1	TFR S,Y	1	BLT	2	ASR	2	ASR	2	ASR	2	ASR	1	ASR	2	ASR	1	EXG A,C	1	EXG X,Y	1	EXG X,S	1	EXG Y,S	1					