

Instruction Encoding																	Instruction Cycle						
syntax	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	type	ifetch1	ifetch2	ifetch3	dfetch	execute	write_back
mv rX, rY	0	0	0	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	mvxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RY), RXen, done		
mv rX, #D	0	0	0	1	X	X	X	D	D	D	D	D	D	D	D	D	mvxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(Im9), RXen, done		
b label	0	0	1	0	0	0	0	D	D	D	D	D	D	D	D	D	bu	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
beq label	0	0	1	0	0	0	1	D	D	D	D	D	D	D	D	D	beq	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if !Z done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bne label	0	0	1	0	0	1	0	D	D	D	D	D	D	D	D	D	bne	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if Z done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bcc label	0	0	1	0	0	1	1	D	D	D	D	D	D	D	D	D	bcc	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if C done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bcs label	0	0	1	0	1	0	0	D	D	D	D	D	D	D	D	D	bcs	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if !C done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bpl label	0	0	1	0	1	0	1	D	D	D	D	D	D	D	D	D	bpl	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if N done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bmi label	0	0	1	0	1	1	0	D	D	D	D	D	D	D	D	D	bmi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, if !N done	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
bl label	0	0	1	0	1	1	1	D	D	D	D	D	D	D	D	D	bl	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(PC), RAen, LRen	MXsel(Im9), ALUsel(add), RBen	MXsel(RB), PCen, done
mvt rX, #D	0	0	1	1	X	X	X	0	D	D	D	D	D	D	D	D	mvtxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(It), RXen, done		
add rX, rY	0	1	0	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	addy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(add), RBen, RFen	MXsel(RB), RXen, done
add rX, #D	0	1	0	1	X	X	X	D	D	D	D	D	D	D	D	D	addxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im9), ALUsel(add), RBen, RFen	MXsel(RB), RXen, done
sub rX, rY	0	1	1	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	subxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(sub), RBen, RFen	MXsel(RB), RXen, done
sub rX, #D	0	1	1	1	X	X	X	D	D	D	D	D	D	D	D	D	subxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im9), ALUsel(sub), RBen, RFen	MXsel(RB), RXen, done
ld rX, [rY]	1	0	0	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	ldxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RY), ADen	wait	MXsel(data), RXen, done
pop rX	1	0	0	1	X	X	X	0	0	0	0	0	0	1	0	1	popx	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(SP), ADen, SPinc	wait	MXsel(data), RXen, done
st rX, [rY]	1	0	1	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	stxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RY), ADen	MXsel(RX), DAen, WRen, done	
push rX	1	0	1	1	X	X	X	0	0	0	0	0	0	1	0	1	pushx	MXsel(PC), ADen, PCinc	wait	IRen	SPdec	MXsel(SP), ADen	MXsel(RX), DAen, WRen, done
and rX, rY	1	1	0	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	andxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(and), RBen	MXsel(RB), RXen, done
and rX, #D	1	1	0	1	X	X	X	D	D	D	D	D	D	D	D	D	andxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im9), ALUsel(and), RBen	MXsel(RB), RXen, done
cmp rX, rY	1	1	1	0	X	X	X	0	0	0	0	0	0	Y	Y	Y	cmpxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(sub), RFen, done	
cmp rX, #D	1	1	1	1	X	X	X	D	D	D	D	D	D	D	D	D	cmpxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im9), ALUsel(sub), RFen, done	
lsl rX, rY	1	1	1	0	X	X	X	1	0	0	0	0	0	Y	Y	Y	lslxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(lsl), RBen, RFen	MXsel(RB), RXen, done
lsl rX, #D	1	1	1	0	X	X	X	1	1	0	0	0	D	D	D	D	lslxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im4), ALUsel(lsl), RBen, RFen	MXsel(RB), RXen, done
lsr rX, rY	1	1	1	0	X	X	X	1	0	0	1	0	0	Y	Y	Y	lsrxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(lsr), RBen, RFen	MXsel(RB), RXen, done
lsr rX, #D	1	1	1	0	X	X	X	1	1	0	1	0	D	D	D	D	lsrxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im4), ALUsel(lsr), RBen, RFen	MXsel(RB), RXen, done
asr rX, rY	1	1	1	0	X	X	X	1	0	1	0	0	0	Y	Y	Y	asrxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(asr), RBen, RFen	MXsel(RB), RXen, done
asr rX, #D	1	1	1	0	X	X	X	1	1	1	0	0	D	D	D	D	asrxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im4), ALUsel(asr), RBen, RFen	MXsel(RB), RXen, done
ror rX, rY	1	1	1	0	X	X	X	1	0	1	1	0	0	Y	Y	Y	rorxy	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(RY), ALUsel(ror), RBen, RFen	MXsel(RB), RXen, done
ror rX, #D	1	1	1	0	X	X	X	1	1	1	1	0	D	D	D	D	rorxi	MXsel(PC), ADen, PCinc	wait	IRen	MXsel(RX), RAen	MXsel(Im4), ALUsel(ror), RBen, RFen	MXsel(RB), RXen, done

Immediate Addressing	
Im4	(15:4) → '0'
It	IR(7:0)
Im9	(15:9) → IR(8)

Register Set
RX, RY : R0, R1, ... R7
R5 ≡ SP ... stack pointer
R6 ≡ LR ... link register
R7 ≡ PC ... program counter

Flag Register RF(C,N,Z)
C ... carry flag
N ... negative flag
Z ... zero flag