Control Signals

	WR.MX	R.MX	AD.MX	INC.MX	LUOP	SF.MX	C.MX	A.MX	NX.MX
0	DPL	A	*PC	PC+	"0"	None	"0"	"0"	NEXT
1	DPH	DPH	*DP	PC-	!A * !B	OPCODE	"1"	A	EXIT.CC
2	SP	SP	*DPt	None	A * !B	SEI/CLD	C	B	INCDPH.C
3	X	X	None	None	!B	Z	IC	"0"	EXIT.BTF
4	Y	Y	*zDP	DPL+	!A * B	NZ	LSR		END
5	T	T	*SP	DPL-	!A	NZC	ROR		END.D
6	AXS	SPI	*fDP	SP+	A XOR B	NZCV	BIT		END.INT
7	P	P	*fCP	SP-	!A + !B	NZV	ADSIC		END.ARNC
8	PCL	PCL	DPH.LD	IR.LD; PC+	A * B	ALU(None)			
9	PCH	PCH	*DP	PC-	!(A XOR B	ALU(OPCODE)			
A	A	DPL	*DPt	None	A	ALU(SEI/CLD)			
B	None	"0"	None	None	A + !B	ALU(Z)			
C	ML	CFG	*zDP	DPL+	B	ALU(NZ)			
D	WAI	STP	*SP	DPL-	!A + B	ALU(NZC)			
E	DPH+1	A&X	*fDP	SP+	A + B	ALU(NZCV)			
F	MEM	BCG	*fCP	SP-	"FF"	ALU(NZV)			

ROM Control Signals Decoding

R.MX	Selects register or constant to load into R bus.
	P will return /INT for the B flag and will therefore push a 1 during a BRK.
WR.MX	Selects W Bus WRITE destination, including MEMory or None.
	ML, WAI and STP are used as triggers for the corresponding 65C02 functions
INC.MX	DEC/INC register (parallel to register write WR.MX). INC.MX0 = DEC/INC operation, INC.MX0/1 target register
	INC.MX3 = IR.LD enables a parallel load of the I register from Data Bus for Opode fetch (i.e. IR := *PC; PC += 1; END;)
INC16	INC.S, INC.C: (00 = +1), (01 = +2), (10 = -2), (11 = -1).
AD.MX	Address Bus Select: "DP", "PC" or "SP". ADH set as follows: $00 for "zDP", $01 for "SP", $FF for "fDP"
	"DPt" sets DPH as ADH and T as the ADL. "fCP" sets ADH to $FF and ADL depends on the interrupt value
	"DPH.LD" enables parallel load of DPH from the Data Bus along with the above addressing modes
SF.MX	Set Flags. SEI/CLD sets I (and clears D if CLD Jumper is enabled)
	OPCODE ***0/1*** decodes opcode for flag, uses the W bus value to ***set/clear*** flag. Use ***LUOP F or 0*** accordingly.
	ALU.EN (bit 3) enables the ALU. !ALU.EN bypasses the ALU for Load operations
NX.MX	State Reg Sequencer Control
	NEXT - Will increment the Q register by 1
	END - Will set the Q register to 0
	EXIT.CC - For Branch instructions - Exit (fetch next Opcde) if no page crossing (i.e. Carry and branch offset same sign)
	EXIT.BTF -For Brnach Instructions - Exit (fetch next Opcode) if the Branch Test Fialed.
	END.D - Same as END except that it enables the BCD circuitry if D flag is on
	INCDPH.C - On Carry set, Insert DPH := DPH + 1 instruction to incrrement high-byte of trget address.
	END.ARNC - Same as END but it indicates an ARR or ANC illegal opcode operations
C.MX	Selects Carry. Fixed value "0" or "1", ALU Carry (C), Internal Carry (IC),
	LSR - Enables Shifter circuit with "0" as carry. ROR - Enables shifter with C as carry.
	BIT - Signals BIT operation … used to select appropriate handling of the flags
	ADSIC - ADD operation with sign extension (i.e. uses latched sign bit from previous cycle). Use IC
A.MX	Selects A input to ALU

External Control Signals Decoding

BE	Bus Enable: Data and Address Bus to High impedance when held low
RDY	Holds clock high pausing CPU. Will take effect next cycle is phase 2 is already underway.
!IRQ	Maskeable interrupt request. Level sensitive. Will take effect once current instruction has completed.
!NMI	Non-Maskeable interrupt request. Edge sensitive. Will take effect once the current instrufition has completed.
!RS	Invokes the soft-reset sequence once the current instruction has completed.

!WAIT	Holds clock high pausing CPU. Should be enabled in Phase 1, never after Phase 2 has begun.

ALU CONROL

Function	LUOP (B out)		A.MX		C.MX	
ORA	"A + B"	E	"0"	0	"0"	0
AND	"A * B"	8	"0"	0	"0"	0
EOR	"A XOR B"	6	"0"	0	"0"	0
ADC	"B"	C	"A"	1	C	2
CMP (SUB)	"!B"	3	"A"	1	"1"	1
SBC	"!B"	3	"A"	1	C	2
ASL A	"A"	A	"A"	1	"0"	0
ASL B	"B"	C	"B"	2	"0"	0
ROL A	"A"	A	"A"	1	C	2
ROL B	"B"	C	"B"	2	C	2
LSR A	"A"	A	"0"	0	LSR	4
LSR B	"B"	C	"0"	0	LSR	4
ROR A	"A"	A	"0"	0	ROR	5
ROR B	"B"	C	"0"	0	ROR	5
DEC	"FF"	F	"A"	1	"1"	1
INC	"0"	0	"A"	1	"1"	1
PASSA	"0"	0	"A"	1	"0"	0
PASSB	"B"	C	"0"	0	"0"	0
ADD	"B"	C	"A"	1	"0"	0
ADIC	"B"	C	"A"	1	IC	3
ADSIC	"B"	C	"A"	1	ADSIC	7

ALR	"A * B"	8	"0"	0	LSR	4

Branch Decoding	
OPCODE	AAABBBCD
Branches	AAX
N	0
V	1
C	2
Z	3
VALUE	XXA

SET/RESET Flags	
OPCODE	AAABBBCD
FLAG:	AAX
C	0
I	1
V	2
D	3
VALUE	XXA