February 2001 REV.2.0

Function Description

The C68MX11 CPU core is based on the Motorola M68HC11 micro-controller, but has an enhanced full 16 bit architecture, thus requiring less clock cycles for completing a large number of instructions compared with original device. In addition to executing all M6800 and M6801 instructions, the C68MX11 instruction set includes more than 90 new op-codes as in the original M68HC11.

A large number of peripheral modules can be added to achieve highly sophisticated, on-chip capabilities, as for example a complete 16 bit timer system with 3 input capture lines, 5 output compare lines and real-time interrupt function ,and an asynchronous or synchronous serial communication interface .
Pins for complete monitoring of internal registers have been added for test purpose.

Features

- Enhanced 16 bit architecture
- Machine Software compatible with industry standard 68HC11
- Byte efficient instructions, powerful addressing modes, 8x8 multiplication supported
- Less machine-cycles per operation
- Memory mapped I/O
- Parallel I/O system
- Real time interrupt system (RTI)
- Synchronous Serial Peripheral Interface system (SPI)
- Full-duplex UART system (SCI)
- 16 bits timer system includes 3 input capture and 5 output compare systems
- 8 bits pulse accumulator
- Watchdog system (COP) interrupt available
- Clock monitor fail interrupt available
- Address space of 64 KBytes
- No internal reset generator or gated clock
- SYNCHRONOUS RESET ; the C68MX11 has 3 reset vectors sources, which easy identify a cause of system reset.
- Fully synthesizable, static synchronous design with no internal tri-states
- VHDL Test bench provided
- De-multiplexed Address/Data Bus to allow easy connection to memory available on request

Symbol

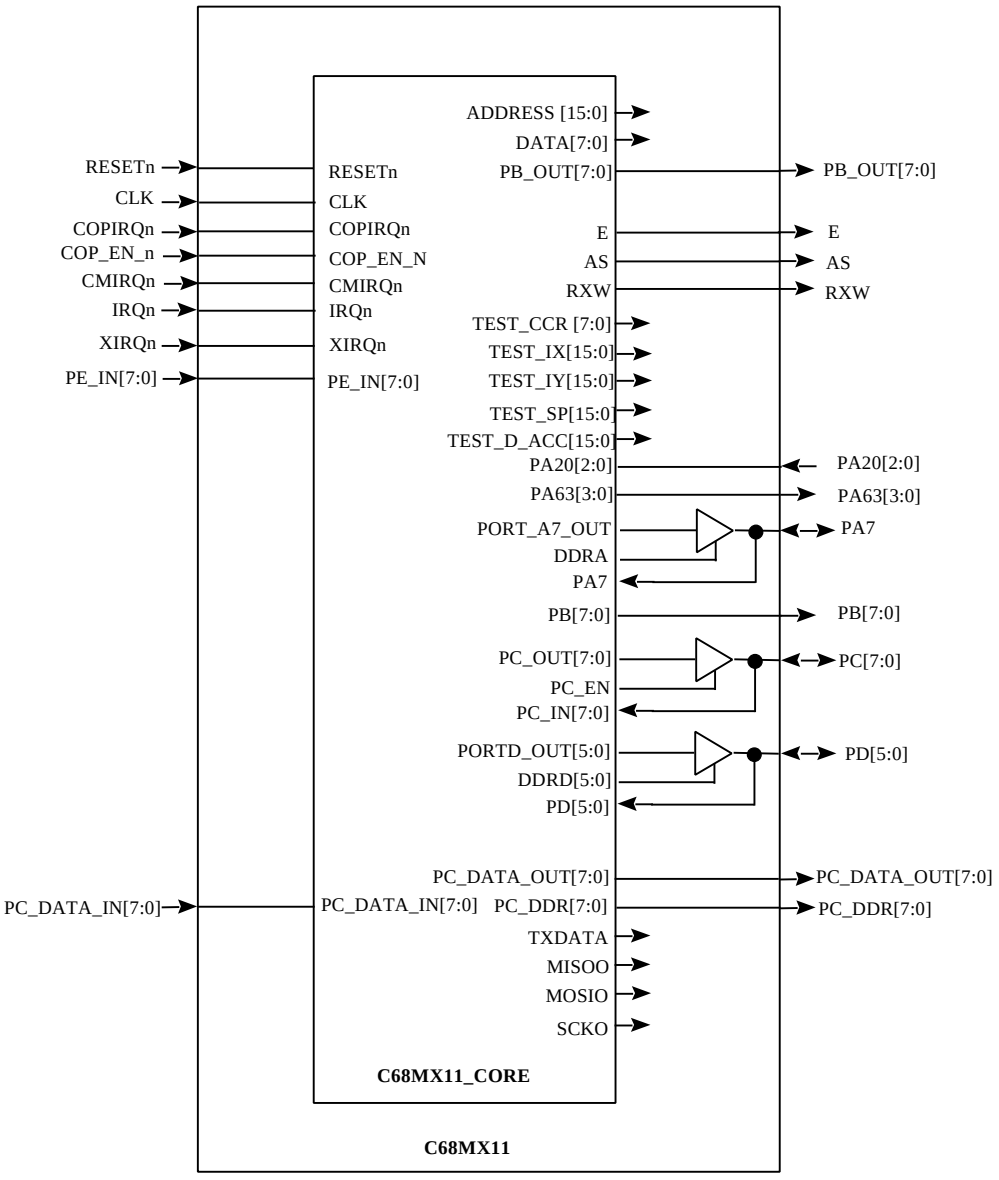

RESETn
CLK
COPIRQn
COP_EN_n
CMIRQn
IRQn
XIRQn
PE_IN[7:0]
PC_DATA_IN[7:0]
RESETn
CLK
COPIRQn
COP_EN_N
CMIRQn
IRQn
XIRQn
PE_IN[7:0]
PC_DATA_IN[7:0]
C68MX11_CORE
ADDRESS [15:0]
DATA[7:0]
PB_OUT[7:0]
E
AS
RXW
TEST_CCR [7:0]
TEST_IX[15:0]
TEST_IY[15:0]
TEST_SP[15:0]
TEST_D_ACC[15:0]
PA20[2:0]
PA63[3:0]
PORT_A7_OUT
DDRA
PA7
PB[7:0]
PC_OUT[7:0]
PC_EN
PC_IN[7:0]
PORTD_OUT[5:0]
DDRD[5:0]
PD[5:0]
PC_DATA_OUT[7:0]
PC_DDR[7:0]
TXDATA
MISOO
MOSIO
SCKO
PB_OUT[7:0]
E
AS
RXW
PA20[2:0]
PA63[3:0]
PA7
PB[7:0]
PC[7:0]
PD[5:0]
PC_DATA_OUT[7:0]
PC_DDR[7:0]
C68MX11

Fig.1

Pin Description

Name	Type	Polarity	Description
RESETN	In	Low	External Reset
CLK	In	-	Master Clock
E	Out	-	Master Clock E
AS	Out	High	Address Strobe
RXW	Out	-	Read/Write Control
CMIRQN	In	Low	Clock Monitor Interrupt
COPIRQN	In	Low	COP Interrupt
COP_EN_N	In	Low	COP Interrupt Enable
XIRQN	In	Low	External Interrupt
IRQN	In	Low/Edge	Non-maskable Interrupt
PE_IN[7:0]	In	-	Port E Input
PC_DATA_IN[7:0]	In	-	Port C Input (I/O)
PC_DATA_OUT[7:0]	Out	-	Port C Output (I/O)
PC_DDR[7:0]	Out	-	Port C data direction (I/O)
PB_OUT[7:0]	Out	-	Port B Output (I/O)
PA20[2:0]	In	-	Port A Input (I/O) Bit 2 to 0
PA63[6:3]	Out	-	Port A Output (I/O) Bit 6 to 3
PA7	In	-	Port A Input (I/O) Bit 7
DDRA	Out	-	Port A bit 7 tri-state buffer enable
PORT_A7_OUT	Out	-	Port A Output (I/O) Bit 7
DDRD[5:0]	Out	-	Port D data direction (I/O)
PORTD_OUT[5:0]	Out	-	Port D Output (I/O)
PD[5:0]	In	-	Port D Input (I/O)
PC_IN[7:0]	In	-	Port C Input (Multiplexed address/data)
PC_OUT[7:0]	Out	-	Port C Output (Multiplexed address/data)
PC_EN	Out	-	Port C tri-state buffers enable
PB[7:0]	Out	-	Port B Output (address [15:8])
TXDATA	Out	-	SCI transmit data
MISOO	Out	-	SPI Master In Slave Out Output
MOSIO	Out	-	SPI Master Out Slave In Output
SCKO	Out	-	SPI Clock Output
RXW	Out	-	Read/Write Control
TEST_IX[15:0]	Out	-	Test Index Register IX (test purpose only)
TEST_IY[15:0]	Out	-	Test Index Register IY (test purpose only)
TEST_SP[15:0]	Out	-	Test Stack Pointer SP (test purpose only)
TEST_D_ACC[15:0]	Out	-	Test Accumulator D (test purpose only)
TEST_CCR[7:0]	Out	-	Test CCR Register (test purpose only)
DATA[7:0]	Out	-	Data Bus (demultiplexed)
ADDRESS[15:0]	Out	-	Address Bus (demultiplexed)

Programmer's Model

7 ACCUMULATOR A 0	7 ACCUMULATOR B 0
15 DOUBLE ACCUMULATOR D 0	

15 INDEX REGISTER X 0

15 INDEX REGISTER Y 0

15 STACK POINTER 0

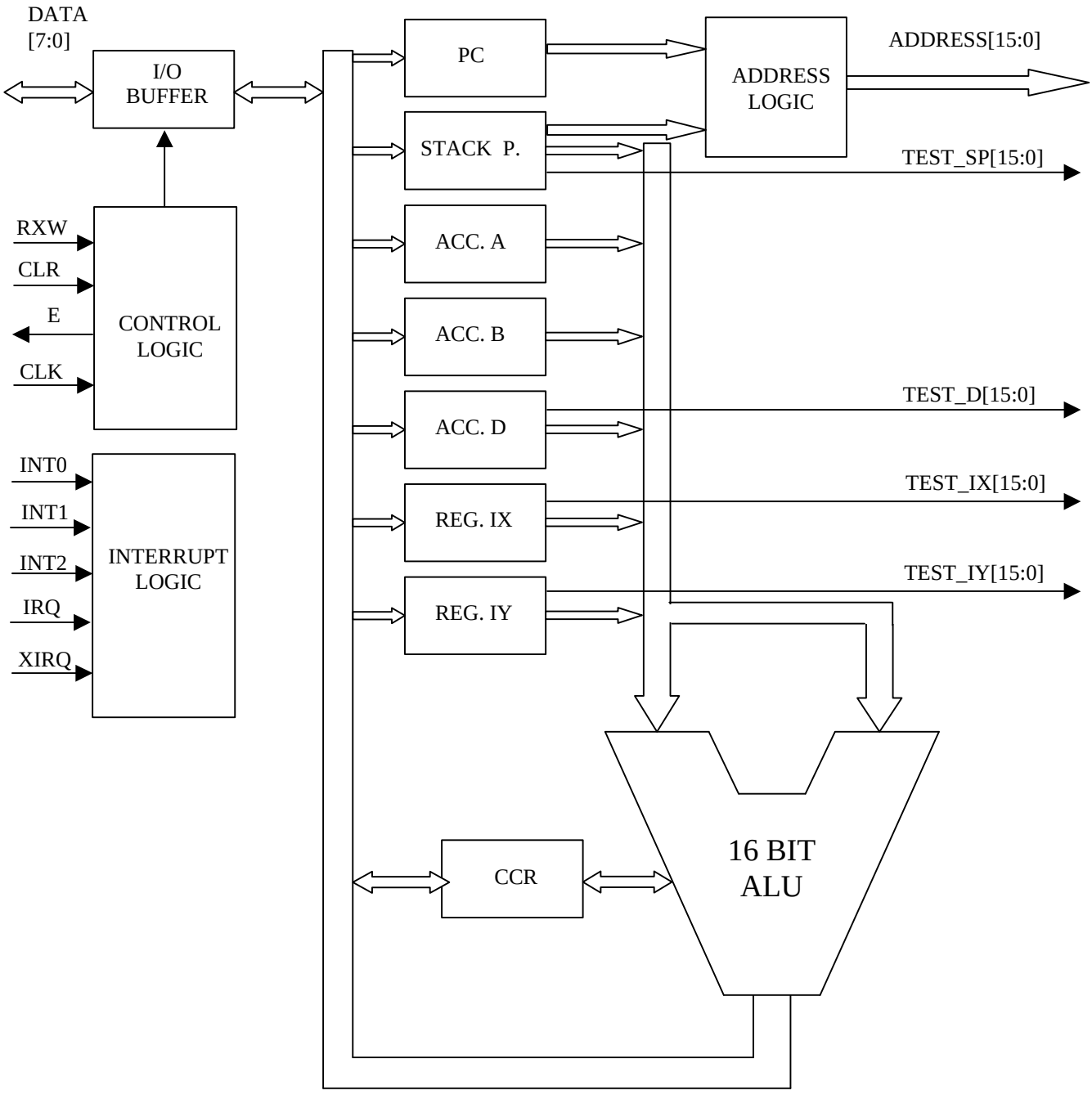
15 PROGRAM COUNTER 0

7 0

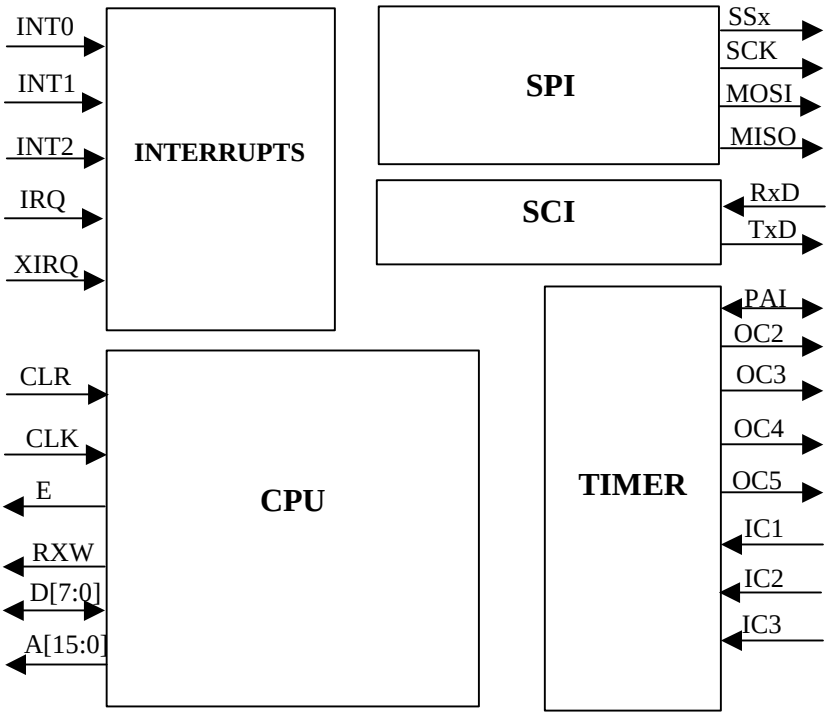
S	X	H	I	N	Z	V	C

| | | | | | | | |_____ CARRY
| | | | | | | |_____ OVERFLOW
| | | | | | |_____ ZERO
| | | | | |_____ NEGATIVE
| | | | |_____ INTERRUPT MASK
| | | |_____ HALF-CARRY
| | |_____ X INTERRUPT MASK
|_____ STOP DISABLE

Block Diagrams



CPU CORE



C68MX11 MICROCONTROLLER

SECTION 1
Introduction

1.1 Supplied Files

1.1.1 Documentation files

\c68mx11\doc

\readme.txt	:last minute details
\c68mx11_um.pdf	:User manual
\c68mx11.pdf	:C68MX11 data sheet

1.1.2 Source code files

\c68mx11\src

\c68mx11.vhd	:Wrapper file with Tri-state buffers
\c6811_pack.vhd	:Definitions package
\c68mx11_core.vhd	:Source Top level
\port_reg_logic.vhd	:Port registers control Unit
\mpu.vhd	:Control Processor Unit
\sequencer.vhd	:Opcodes decoder
\sci.vhd	:Serial Interface Unit
\spi.vhd	:Synchronous Serial Peripheral Interface Unit
\timer_11.vhd	:Timer Unit
\ram.vhd	:Internal RAM

1.1.3 LPM related files

\c68mx11\src\lpm	:files available only for netlist version

\ sequencer_lpm.vhd
\ ram_lpm.vhd
\ lpm_rom_50_0_0.hex
\ lpm_rom_50_0_0.mif
\ c6811ram.vhd
\ c6811rom.vhd
\ lpm_rom_4_0_0.hex
\ lpm_rom_4_0_0.mif

1.1.4 Test Bench related files

a) Test Bench files

\c68mx11\tb

\ c68mx11tb.vhd	:Basic Test Bench for C68MX11
\ c6811tb_pack.vhd	:Test Bench definitions package for C68MX11
\ c68mx11tb_sci.vhd	:Test Bench for SCI tests
\ c68mx11tb_tim.vhd	:Test Bench for TIMER tests
\ c68mx11tb_spi.vhd	:Test Bench for SPI tests
\ test_spi.vhd	:Test SPI to be used with SPI Testbench
\ extram.vhd	:External RAM memory

\ extrom.vhd :External ROM memory

b) Test Bench support files

\c68mx11\tb\tests

\ default\ : default test files
\ Test_mpu\ : MPU test files
\ Test_tim\ : Timer Unit test files
\ Test_sci\ : Serial interface test files
\ Test_spi\ : Synchronous Serial interface test files
\ Test_ports\ : Ports test files

c) MPU Test Bench support files

C68mx11\tb\tests\test_mpu

\ Test_all\ : Operation codes test files
\ Test_jmp\ : Jump Operation codes test files
\ Test_int\ : Interrupts test files
\ Test_iedg\ : Interrupts (edge mode) test files
\ Test_daa\ : DAA Operation code test files
\ Test_subd\ : SUBD Operation code test files
\ Test_cp\ : CP Operation code test files
\ Testram1\ : internal RAM test files (1st part)
\ Testram2\ : internal RAM test files (2nd part)
\ Testram3\ : internal RAM test files (3rd part)
\ Testram4\ : internal RAM test files (3rd part)
\ Testram5\ : internal RAM test files (4th part)
\ Testram6\ : internal RAM test files (6th part)
\ Testram7\ : internal RAM test files (7th part)
\ Testram8\ : internal RAM test files (8th part)
\ Testram9\ : internal RAM test files (9th part)
\ Testram10\ : internal RAM test files (10th part)
\ Testram11\ : internal RAM test files (11th part)
\ Testram12\ : internal RAM test files (12th part)
\ Testram13\ : internal RAM test files (13th part)
\ Testram14\ : internal RAM test files (14th part)
\ Testram15\ : internal RAM test files (15th part)
\ Testram16\ : internal RAM test files (16th part)
\ Testram17\ : internal RAM test files (17th part)
\ Testram18\ : internal RAM test files (18th part)

d) PORTS Test Bench support files

\C68mx11\tb\tests\test_ports

\ Test_p12\ : Port 1_2 test files
\ Test_p13\ : Port 1_3 test files

e) SCI Test Bench support files

\c68mx11\tb\tests\test_sci

\ Tst_sc02\ : Serial Interface test files (Baud rate E/16)
\ Tst_sc03\ : Serial Interface test files (Baud rate E/64)
\ Tst_sc04\ : Serial Interface test files (Baud rate E/128)
\ Tst_scw1\ : Serial Interface test files (Wake-up feature)
\ Tst_scw2\ : Serial Interface test files (Wake-up feature)

f) SPI Test-Bench support files

\c68mx11\tb\tests\test_spi

\ Tst_spi1\ : Serial Interface test files (Baud rate E/16)
\ Tst_spi2\ : Serial Interface test files (Baud rate E/16)
\ Tst_spi3\ : Serial Interface test files (Baud rate E/64)
\ Tst_spi4\ : Serial Interface test files (Baud rate E/128)
\ Tst_spii\ : Serial Interface test files (Wake-up feature)

g) TIMER Test Bench support files
\c68mx11\tb\tests\test_tim

\Test_ic1\ : Timer Unit test files (Input capture 1)
\Tst_ic1f\ : Timer Unit test files (Input capture 1)
\Tst_ic1r\ : Timer Unit test files (Input capture 1)
\Tst_ic1b\ : Timer Unit test files (Input capture 1)
\Tst_ic1i\ : Timer Unit test files (Input capture 1)
\Test_ic2\ : Timer Unit test files (Input capture 2)
\Tst_ic2f\ : Timer Unit test files (Input capture 2)
\Tst_ic2r\ : Timer Unit test files (Input capture 2)
\Tst_ic2b\ : Timer Unit test files (Input capture 2)
\Tst_ic2i\ : Timer Unit test files (Input capture 2)
\Test_ic3_xxxx\ : Timer Unit test files (Input capture 3)
\Tst_ic3f\ : Timer Unit test files (Input capture 3)
\Tst_ic3r\ : Timer Unit test files (Input capture 3)
\Tst_ic3b\ : Timer Unit test files (Input capture 3)
\Tst_ic3i\ : Timer Unit test files (Input capture 3)
\Test_oc1\ : Timer Unit test files (Output capture 1)
\Test_oc2\ : Timer Unit test files (Output capture 2)
\Test_oc3\ : Timer Unit test files (Output capture 3)
\Tst_oc10\ : Timer Unit test files (Output capture 1)
\Tst_oc11\ : Timer Unit test files (Output capture 1)
\Tst_oc12\ : Timer Unit test files (Output capture 1)
\Tst_oc13\ : Timer Unit test files (Output capture 1)
\Tst_oc14\ : Timer Unit test files (Output capture 1)
\Tst_oc15\ : Timer Unit test files (Output capture 1)
\Tst_oc1i\ : Timer Unit test files (Output capture 1)
\Tst_oc20\ : Timer Unit test files (Output capture 2)
\Tst_oc21\ : Timer Unit test files (Output capture 2)
\Tst_oc22\ : Timer Unit test files (Output capture 2)
\Tst_oc2i\ : Timer Unit test files (Output capture 2)
\Tst_oc30\ : Timer Unit test files (Output capture 3)
\Tst_oc31\ : Timer Unit test files (Output capture 3)
\Tst_oc32\ : Timer Unit test files (Output capture 3)
\Tst_oc3i\ : Timer Unit test files (Output capture 3)
\Tst_oc40\ : Timer Unit test files (Output capture 4)
\Tst_oc41\ : Timer Unit test files (Output capture 4)
\Tst_oc42\ : Timer Unit test files (Output capture 4)

\Tst_oc4i\	: Timer Unit test files (Output capture 4)
\Tst_oc50\	: Timer Unit test files (Output capture 5)
\Tst_oc51\	: Timer Unit test files (Output capture 5)
\Tst_oc52\	: Timer Unit test files (Output capture 5)
\Tst_oc5i\	: Timer Unit test files (Output capture 5)
\Tst_frc1\	: Timer Unit test files (Force capture 1)
\Tst_frc2\	: Timer Unit test files (Force capture 1)
\Tst_frc3\	: Timer Unit test files (Force capture 1)
\Tst_rtf1\	: Real-Time Flag RTIF test files (divide by 1)
\Tst_rtf2\	: Real-Time Flag RTIF test files (divide by 2)
\Tst_rtf3\	: Real-Time Flag RTIF test files (divide by 4)
\Tst_rtf4\	: Real-Time Flag RTIF test files (divide by 8)
\Tst_rtfi\	: Real-Time Interrupt RTI test files (divide by 1)
\Tst_tof \	: Timer Overflow Flag test files
\Tst_toi \	: Timer Overflow Interrupt test files
\Tst_pai1\	: Pulse Accumulator Flag test files (event counting)
\Tst_pai2\	: Pulse Accumulator Flag test files (event counting)
\Tst_paii\	: Pulse Accumulator Interrupt test files (event counting)
\Tst_pav1\	: Pulse Accumulator Flag test files (gated time)
\Tst_pav2\	: Pulse Accumulator Flag test files (gated time)
\Tst_pavi\	: Pulse Accumulator Interrupt test files (gated time)
\Tst_pre1\	: Timer Unit Prescaler test files (divide by 4)
\Tst_pre2\	: Timer Unit Prescaler test files (divide by 8)
\Tst_pre3\	: Timer Unit Prescaler test files (divide by 16)

Each test is composed of 4 files:

extrom.txt	: Content of external ROM in Intel hex file format
C68mx11.ref	: Expected Test Bench results. The simulation results from test are automatically compared with this file
C68mx11.stm	: Stimuli file for Test Bench
Test_xxxx.s07	: Assembler code of program executed during simulation

1.1.5 MTI Modelsim PE/SE support files

C68mx11\tb\tests\test_mpu

\ compile.do :MTI Compilation Macro

C68mx11\tb\tests\test_sci

\ compile.do :MTI Compilation Macro

C68mx11\tb\tests\test_spi

\ compile.do :MTI Compilation Macro

C68mx11\tb\tests\test_tim

\ compile.do :MTI Compilation Macro

SECTION 2

CPU Functional description

The central processing unit (CPU) of the C68MX11 has more than 300 instruction op-codes and 6 addressing modes that can be used to reference memory:

- IMM - **immediate** (the actual argument is contained in the byte(s) immediately following the instruction)
- DIR - **direct** (the least significant byte of the effective address of the instruction is contained in the byte following the op-code. The high-order byte of the effective address is assumed to be hex 00)
- EXT - **extended** (the effective address explicitly appears in the 2 bytes following the op-code)
- IND - **indexed** (either index register IX or IY is used for effective address calculation)
- INH - **inherent** (the operands are CPU registers and they are inherently known by the CPU)
- REL - **relative** (used only for branch instructions)

The CPU is able of addressing 64 KBytes of memory. I/O access is memory-mapped. Although the data busses have a width of 8 bit, most instructions have a 16 bit equivalent instruction. The C68MX11 offers multiply, add, subtract, compare, increment & decrement, load & store, and shift instructions of 16 bit operands. The CPU consists of two general-purpose 8 bit **accumulators** used to hold operands and results of arithmetic calculations or data manipulations. The accumulator A and B can be combined into a 16 bit double accumulator D. The 16 bit **index registers** IX and IY are used for indexed addressing modes. The CPU automatically supports a program stack. This stack may be located anywhere in the 64 KByte address space through the **stack pointer** and may be of any size up to the amount of memory available in the system. The **condition code register (CCR)** contains five status indicators (carry, overflow, zero, negative, and the half carry flag), two interrupt masking bits (IRQ and XIRQ mask), and a STOP disable bit.

Table 2.1 Register Map

		7	6	5	4	3	2	1	0		
R/W	$1000	Bit 7	-	-	-	-	-	-	Bit 0	**PORTA**	I/0 Port A
R/W	$1001										RESERVED
R	$1002	0	0	0	0	0	0	0	0	**PIOC**	NOT AVAILABLE
R/W	$1003	Bit 7	-	-	-	-	-	-	Bit 0	**PORTC**	I/0 Port C
R/W	$1004	Bit 7	-	-	-	-	-	-	Bit 0	**PORTB**	OUT Port B
R	$1005	0	0	0	0	0	0	0	0	**PORTCL**	NOT AVAILABLE
R	$1006	0	0	0	0	0	0	0	0		RESERVED
R/W	$1007	Bit 7	-	-	-	-	-	-	Bit 0	**DDRC**	Data Direction C
R/W	$1008	0	0	Bit 5	-	-	-	-	Bit 0	**PORTD**	I/0 Port D
R/W	$1009	0	0	Bit 5	-	-	-	-	Bit 0	**DDRD**	Data Direction D
R	$100A	Bit 7	-	-	-	-	-	-	Bit 0	**PORTE**	IN Port E
R/W	$100B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	**CFORC**	Compare Force
R/W	$100C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	**OC1M**	OC1 Mask
R/W	$100D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	**OC1D**	OC1 Data
R/W	$100E	Bit 15	-	-	-	-	-	-	Bit 8	**TCNT**	Timer Counter
R/W	$100F	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$1010	Bit 15	-	-	-	-	-	-	Bit 8	**TIC1**	Input Capture 1
R/W	$1011	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$1012	Bit 15	-	-	-	-	-	-	Bit 8	**TIC2**	Input Capture 2
R/W	$1013	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$1014	Bit 15	-	-	-	-	-	-	Bit 8	**TIC3**	Input Capture 3
R/W	$1015	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$1016	Bit 15	-	-	-	-	-	-	Bit 8	**TOC1**	Out Capture 1
R/W	$1017	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$1018	Bit 15	-	-	-	-	-	-	Bit 8	**TOC2**	Out Capture 2
R/W	$1019	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$101A	Bit 15	-	-	-	-	-	-	Bit 8	**TOC3**	Out Capture 3
R/W	$101B	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$101C	Bit 15	-	-	-	-	-	-	Bit 8	**TOC4**	Out Capture 4
R/W	$101D	Bit 7	-	-	-	-	-	-	Bit 0		
R/W	$101E	Bit 15	-	-	-	-	-	-	Bit 8	**TI4O5**	Out Capture 5
R/W	$101F	Bit 7	-	-	-	-	-	-	Bit 0		

Register Map

		7	6	5	4	3	2	1	0		
R/W	$1020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	**TCTL1**	Timer Control 1
R/W	$1021	0	0	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	**TCTL2**	Timer Control 2
R/W	$1022	OC1I	OC2I	OC3I	OC4I	I4O5I	IC1I	IC2I	IC3I	**TMSK1**	Timer Interrupt Mask 1
R/W	$1023	OC1F	OC2F	OC3F	OC4F	I4O5F	IC1F	IC2F	IC3F	**TFLG1**	Timer Interrupt Flag 1
R/W	$1024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	**TMSK2**	Timer Interrupt Mask 2
R/W	$1025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	**TFLG2**	Timer Interrupt Flag 2
R/W	$1026	DDRA	PAEN	PAMOD	PEDGE	0	0	RTR1	RTR0	**PACTL**	Pulse Accum. Control
R/W	$1027	Bit 7	-	-	-	-	-	-	Bit 0	**PACNT**	Pulse Accum. Count
R/W	$1028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	**SPCR**	SPI Control
R	$1029	SPIF	WCOL	0	MODF	0	0	0	0	**SPSR**	SPI Status
R/W	$102A	Bit 7	-	-	-	-	-	-	Bit 0	**SPDR**	SPI Data
R/W	$102B	TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0	**BAUD**	SCI Baud Rate
R/W	$102C	R8	T8	0	M	WAKE	0	0	0	**SCCR1**	SCI Control 1
R/W	$102D	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK	**SCCR2**	SCI Control 2
R	$102E	TDRE	TC	RDRF	IDLE	OR	NF	FE	0	**SCSR**	SCI Status
R/W	$102F	Bit 7	-	-	-	-	-	-	Bit 0	**SCDR**	SCI Data
-	$1030	Reserved									
-	$1031										
-	$1032										
-	$1033										
-	$1034										
-	$1035										
-	$1036										
-	$1037										
-	$1038										
R	$1039	0	0	IRQE	DLY	CME	0	CR1	CR0	**OPTION**	System Conf.
R	$103A	0	0	0	0	0	0	0	0	**COPRST**	NOT AVAILABLE
R	$103B	0	0	0	0	0	0	0	0	**PPROG**	NOT AVAILABLE
R	$103C	0	0	1	0	PSEL3	PSEL2	PSEL1	PSEL0	**HPRIO**	High Priority I-bit Interrupt
R/W	$103D	RAM3	RAM2	RAM0	RAM1	REG3	REG2	REG1	REG0	**INIT**	RAM I/O Mapping
R	$103E	0	0	0	0	0	0	0	0	**TEST1**	NOT AVAILABLE
R	$103F	0	0	0	0	1	NOCOP	0	0	**CONFIG**	COP –ROM – EEPROM Enable

SECTION 3
ON-CHIP MEMORY

3.1 RAM

C68MX11 has 256-byte on-chip RAM. This 256-byte RAM can be mapped to the beginning of any 4-Kbyte block in the 64-Kbyte address space.

SECTION 4
RESETS AND INTERRUPTS

The reset structure in the C68MX11 is discussed in this section.
The C68MX11 includes 18 separate interrupt sources. On-chip peripheral systems generate maskable interrupts, which are recognized only if the global interrupt mask bit (I) in the condition code register (CCR) is clear. Maskable interrupts are prioritized according to a default arrangement; however, any one source may be elevated to the highest maskable priority position by a software-accessible control register. This highest priority interrupt (HPRIO) register may be written at any time provided the I bit in the CCR is set.
When interrupt conditions occur in an on-chip peripheral system, an interrupt status flag is set to indicate the condition. When the user's program has properly responded to this interrupt request, the status flag must be cleared.

4.1.1 System Initial Conditions

Once the reset condition is recognized, internal registers and control bits are forced to an initial state. These initial states, in turn, control on-chip peripheral systems to force them to known start-up states. The following paragraphs summarize the initial conditions of the MCU after reset.

4.1.1.1 CPU

After reset, the CPU fetches the restart vector from locations $FFFE,FFFF during the first three cycles and begins executing instructions. The X and I interrupt mask bits in the CCR are set to mask any interrupt requests. Also, the S bit in the CCR is set to disable the STOP mode. STOP mode is not supported.

4.1.1.2 Memory Map

After reset, the RAM and I/O mapping (INIT) register is initialized to $01, putting the 256 bytes of random-access memory (RAM) at locations $0000–$00FF and the control registers at locations $1000–$103F.

4.1.1.3 Timer

During reset, the timer system is initialized to a count of $0000. The prescaler bits are cleared, and all output-compare registers are initialized to $FFFF. All input-capture registers are indeterminate after reset. The output-compare 1 (OC1M) mask register is cleared so that successful OC1 compares do not affect any I/O pins. The other four output compares are configured to not affect any I/O pins on successful compares. All three input-capture edge-detector circuits are configured for capture-disabled operation. The timer overflow interrupt flag and all eight timer function interrupt flags are cleared. All nine timer interrupts are disabled since their mask bits are cleared.

4.1.1.4 Real-Time Interrupt

The real-time interrupt flag is cleared, and automatic hardware interrupts are masked.
The rate control bits are cleared after reset and may be initialized by software before the real-time interrupt system is used.

4.1.1.5 Pulse Accumulator

The pulse accumulator system is disabled at reset so that the pulse accumulator input (PAI) pin defaults to being a general-purpose input pin.

4.1.1.6 Serial Communications Interface (SCI)

At reset, the SCI baud rate register is reset to 0000.All transmit and receive interrupts are masked, and both the transmitter and receiver are disabled so the port pins at default are general-purpose I/O lines. The SCI frame format is initialized to an 8-bit character size. The receiver wake-up functions are disabled. The transmit data register empty (TDRE) and transmit complete (TC) status bits in the SCI status register are both set, indicating that there is no transmit data in either the transmit data register or the transmit serial shift register. The receive data register full (RDRF), IDLE, overrun (OR), and framing error (FE) receive-related status bits are all cleared.

4.1.1.7 Serial Peripheral Interface (SPI)

The SPI system is disabled by reset. The port pins associated with this function at default are general-purpose I/O lines.

4.2 Causes Of Reset

In the C68MX11, there are on-chip systems that can detect MCU system failures . To distinguish between these causes, separate reset Vectors are used. The primary reset vector is used when the cause of reset is the internal power-on reset circuit or application of a low level to the RESET pin. This vector is located at $FFFE,FFFF.

Table 4-1 Reset Vector vs. Cause of Reset

Cause of Reset	Vector Special
Vector Special	
POR or RESET Pin	$FFFE,FFFF
Clock Monitor Fail	$FFFC,FFFD
COP Watchdog Time-Out	$FFFA,FFFB

4.2.1 Clock Monitor Fail (CMI)

4.2.2 COP Watchdog Timer Reset

CMI and COP reset are modelled as external reset input to the core.
CM interrupt is active when CMIRQn input pin is low level.
COP interrupt is valid when COPIRQn pin is low and COP_EN_n (COP interrupt enable) pin is low.

4.2.3 External Reset

In addition to the internal sources, reset can be forced by applying a low level to the RESETn pin.

SECTION 5
Peripherals

SYNCHRONOUS SERIAL PERIPHERAL INTERFACE (SPI)

The serial peripheral interface (SPI) is one of two independent serial communications subsystems included on the C68MX11. As the name implies, the SPI is primarily used to allow the micro-controller unit (MCU) to communicate with peripheral devices.
The SPI is also capable of inter-processor communications in a multiple-master system.
The SPI system is flexible to interface directly with numerous standard product peripherals from several manufacturers.
The system can be configured as a master or a slave device. Data rates as high as 1 Mbit per second (or more depending from silicon implementation) are accommodated when the system is configured as a master; rates as high as 2 Mbits per second (or more depending from silicon implementation) are accommodated when the system is operated as a slave.
Clock control logic allows a selection of clock polarity and a choice of two fundamentally different clocking protocols to accommodate most available synchronous serial peripheral devices. When the SPI is configured as a master, software selects one of four different bit rates for the serial clock.
Error-detection logic is included. A write collision detector indicates when an attempt is made to write data to the serial shift register while a transfer is in progress. A multiple-master mode-fault detector automatically disables SPI output drivers if more than one MCU simultaneously attempts to become bus master.

ASYNCHRONOUS SERIAL COMMUNICATIONS INTERFACE (SCI)

The SCI is a full-duplex UART-type asynchronous system, using standard non-return-to-zero (NRZ) format (one start bit, eight or nine data bits, and a stop bit). An on-chip baud rate generator derives standard baud-rate frequencies from the MCU CLK input pin.
Both the transmitter and the receiver are double buffered.
The SCI transmitter and receiver are functionally independent but use the same data format and baud rate.
Primarily, the SCI system is configured and controlled by five registers (BAUD, SCCR1, SCCR2, SCSR, and SCDR). In addition, the port D register, data direction register for port D (DDRD), are related to the SCI system.
When the SCI receiver and/or transmitter is enabled, the SCI logic takes control of the pin buffers for the associated port D pin(s). Data directions for the RxD and TxD pins are overridden to input and output, respectively. Even though it does not control the direction of port D pins while the SCI has control, the DDRD can be important to a user because it influences what will be read when port D is read by software. The DDRD also determines how the pin will behave when the SCI gives up control. The port D register is important to an SCI user because the value written to port D can determine what will be driven out of port D when the SCI gives up control.

The baud-rate control register (BAUD) is used to select the baud rate for SCI.

SCCR1 includes three bits associated with the optional 9-bit data format. The WAKE bit is used to select one of two methods of receiver wake up.

SCCR2 contains the main SCI controls. The upper four bits are local interrupt enable controls, which determine whether SCI status flags will be polled or will generate hardware interrupt requests. The TE and RE bits are the respective transmitter and receiver subsystem enable controls. The RWU bit allows software to put the receiver to sleep and hardware to

automatically wake it up by clearing this bit. The send break SBK bit allows software to generate break characters on the TxD line. (Send break is not modelled)

The SCSR contains two transmitter status flags and five receiver-related status flags.
The transmitter generates flags for TDRE and TC. The receiver generates flags for RDRF, OR, idle-line detect (IDLE), a noise flag (NF), and a framing error (FE) indication.

The SCDR is actually two separate registers. TDR is a write-only transmit data buffer register, and RDR is a read-only receive data buffer register. When software reads SCDR, it is accessing RDR; when software writes to SCDR, it is accessing TDR.

MAIN TIMER AND REAL-TIME INTERRUPT

This section describes the main timer system of the C68MX11.
The architecture of the main timer is primarily a software-driven system

5.1 General Description

This timer system is based on a free-running 16-bit counter with a four-stage programmable prescaler. A timer overflow function allows software to extend the timing capability of the system beyond the 16-bit range of the counter. Three independent input-capture functions are used to automatically record (latch) the time when a selected transition is detected at a respective timer input pin.
Five output-compare functions are included for generating output signals or for timing software delays.

A programmable periodic interrupt circuit called the real-time interrupt (RTI) is derived from the main 16-bit timer counter. Software can select one of four rates for the RTI.

Each of the three input-capture functions has its own 16-bit time capture latch (input-capture register) and each of the five output-compare functions has its own 16-bit compare register. All timer functions, including the timer overflow and RTI have their own interrupt controls and separate interrupt vectors. Additional control bits permit software to control the edge(s) that trigger each input-capture function and the automatic actions that result from output-compare functions.

5.1.2 Input-Capture Concept

For the MCU, physical time is represented by the count in the 16-bit free-running counter. Input capture functions, used to record the time at which some external event occurred, are accomplished by latching the contents of the free-running counter when a selected edge is detected at the related timer input pin. The time at which the event occurred is saved in the input capture register (16-bit latch);

5.1.3 Output-Compare Concept

Output-compare functions are used to program an action to occur at a specific time (when 16-bit free-running counter reaches a specific value). For each of the five output-compare functions, there is a separate 16-bit compare register and a dedicated 16-bit comparator. The value in the compare register is compared to the value of the free-running counter on every bus cycle. When the compare register matches the counter value, an output is generated, which sets an output-compare status flag and initiates the automatic actions for that output-compare function. Optional automatic actions initiated by an output compare include generation of a hardware interrupt request and state changes at the related timer output pin(s).

5.1.4 Real-Time Interrupt (RTI) Function

The RTI function can be used to generate hardware interrupts at a fixed periodic rate. In the C68MX11, the RTI system can be used to provide this periodic time reference signal. The clock source for the RTI function is a free-running clock that cannot be stopped or interrupted.

5.1.5 PULSE ACCUMULATOR

The pulse accumulator is based on an 8-bit counter and can be configured to operate as a simple event counter or for gated time accumulation. Unlike the main timer, the 8-bit pulse accumulator counter can be read or written at any time. Control bits allow the user to configure and control the pulse accumulator subsystem. Two maskable interrupts are associated with the system, each having its own controls and interrupt vector.
The port A bit 7 I/O pin (PA7/PAI/OC1) associated with the pulse accumulator can be configured to act as a clock (event counting mode) or as a gate signal to enable a free-running E divided by 64 clock to the 8-bit counter (gated time accumulation mode).

SECTION 6
Modeling Notes

The design is **fully-synchronous**, and doesn t contain any internal tri-state buses, therefore special considerations for synthesis can be avoided. The design offers all of the original 308 op-codes and behaves exactly in the same manner as the original M68HC11, with the exceptions discussed below. Due to this modification the C68MX11 has actually 304 Op-codes.

The variation from original device are as following:

- The following OPCODES and relative mnemonics have been NOT implemented in order to save area, except for TEST which didn t make sense in a core implementation. These opcodes can be added if needed.

1. FDIV INH Op-code 03
2. IDIV INH Op-code 02
3. STOP INH Op-code CF
4. TEST INH Op-code 00

- All registers or bit marked in RED in table 2.1 (Registers Map) have been not modeled.
- "Normal Expanded" mode only supported
- Register HPRIO ="0010" & psel
- Register CONFIG ="00001100"
- POR (Power On reset) not modeled
- Clock Monitor Reset modeled as external interrupt
- COP watchdog Timer modeled as external interrupt with its own enable
- Handshake mode of port C and PIOC,PORTCL registers not modeled
- DLY bit 4 of OPTION Register is always 0 'not modeled
- **SCI:**
- BAUD register is reset to "00000000" instead of "00000UUU"
- TCLR bit 7 and RCKB bit 3 of BAUD register are always 0 '(test mode disabled)
- SCCR1 register is reset to "00000000" instead of " UU000000"
- Send Break Feature is not modeled
- **SPI:**
- SPCR register is reset to "00000100" instead of " 000001UU"
- DWOM bit 5 of SPCR Register is always 0 'not modeled

SECTION 8
Instruction Set Details

This section contains complete information for all C68MX11 instructions. The instruction are arranged in alphabetical order of mnemonics .Op-codes with pre-byte are listed explicitly.

- **Mnemonic**: The instruction's mnemonic.
- **Op-code**: The instruction's op-code hexadecimal value (with pre-byte 18/1A/CD if available).
- **Addressing Mode**: The instruction's addressing mode.
- **Address Instruction Cycles 1 - 5**: Current state of the address bus during cycle-by-cycle execution. Only the initial five instruction cycles are showed.
- **Instruction Cycles**: The values of this column show the number of cycles needed to execute an instruction. The C68MX11 core has an improved cycle behavior compared to the original 68HC11; some instructions can be completed in less clock cycles (= result of the subtraction). For example, the original M68HC11 needs 4 cycles to perform an ADDD instruction, the C68MX11 accomplishes the same task in 3 cycles, resulting in a speed-up of 1 cycle.

Mnemonic	Opcode	Addr Mode	Addr Inst Cyc 1	Addr Inst Cyc 2	Addr Inst Cyc 3	Addr Inst Cyc 4	Addr Inst Cyc 5	Instruction Cycles
ABA	1B	INH	OP	OP+1				2 - 2 = 0
ABX	3A	INH	OP	OP+1	FFFF			3 - 3 = 0
ABY	18 3A	INH	OP	OP+1	OP+2	FFFF		4 - 4 = 0
ADCA	89	IMM	OP	OP+1				2 - 2 = 0
ADCA	99	DIR	OP	OP+1	00dd			3 - 3 = 0
ADCA	A9	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ADCA	18 A9	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ADCA	B9	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ADCB	C9	IMM	OP	OP+1				2 - 2 = 0
ADCB	D9	DIR	OP	OP+1	00dd			3 - 3 = 0
ADCB	E9	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ADCB	18 E9	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ADCB	F9	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ADDA	8B	IMM	OP	OP+1				2 - 2 = 0
ADDA	9B	DIR	OP	OP+1	00dd			3 - 3 = 0
ADDA	AB	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ADDA	18 AB	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ADDA	BB	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ADDB	CB	IMM	OP	OP+1				2 - 2 = 0
ADDB	DB	DIR	OP	OP+1	00dd			3 - 3 = 0
ADDB	EB	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ADDB	18 EB	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ADDB	FB	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ADDD	C3	IMM	OP	OP+1	OP+2			4 - 3 = 1
ADDD	D3	DIR	OP	OP+1	00dd	00dd+1		5 - 4 = 1
ADDD	E3	IND,X	OP	OP+1	IX+ff	IX+ff+1		6 - 4 = 2
ADDD	18 E3	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	7 - 5 = 2
ADDD	F3	EXT	OP	OP+1	OP+2	hhll	hhll+1	6 - 5 = 1
ANDA	84	IMM	OP	OP+1				2 - 2 = 0
ANDA	94	DIR	OP	OP+1	00dd			3 - 3 = 0
ANDA	A4	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ANDA	18 A4	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ANDA	B4	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ANDB	C4	IMM	OP	OP+1				2 - 2 = 0
ANDB	D4	DIR	OP	OP+1	00dd			3 - 3 = 0

ANDB	E4	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ANDB	18 E4	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ANDB	F4	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ASL	68	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
ASL	18 68	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff	7 - 5 = 2
ASL	78	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
ASLA	48	INH	OP	OP+1				2 - 2 = 0
ASLB	58	INH	OP	OP+1				2 - 2 = 0
ASLD	5	INH	OP	OP+1	FFFF			3 - 3 = 0
ASR	67	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
ASR	18 67	IND,Y	OP	OP+1	IY+ff	IY+ff		7 - 5 = 2
ASR	77	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
ASRA	47	INH	OP	OP+1				2 - 2 = 0
ASRB	57	INH	OP	OP+1				2 - 2 = 0
BCC/BHS	24	REL	OP	OP+1	FFFF			3 - 3 = 0
BCLR	15	DIR	OP	OP+1	00dd	OP+2	00dd	6 - 5 = 1
BCLR	1D	IND,X	OP	OP+1	IX+ff	OP+2	IX+ff	7 - 5 = 2
BCLR	18 1D	IND,Y	OP	OP+1	OP+2	IY+ff	(....)	8 - 6 = 2
BCS/BLO	25	REL	OP	OP+1	FFFF			3 - 3 = 0
BEQ	27	REL	OP	OP+1	FFFF			3 - 3 = 0
BGE	2C	REL	OP	OP+1	FFFF			3 - 3 = 0
BGT	2E	REL	OP	OP+1	FFFF			3 - 3 = 0
BHI	22	REL	OP	OP+1	FFFF			3 - 3 = 0
BITA	85	IMM	OP	OP+1				2 - 2 = 0
BITA	95	DIR	OP	OP+1	00dd			3 - 3 = 0
BITA	A5	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
BITA	18 A5	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
BITA	B5	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
BITB	C5	IMM	OP	OP+1				2 - 2 = 0
BITB	D5	DIR	OP	OP+1	00dd			3 - 3 = 0
BITB	E5	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
BITB	18 E5	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
BITB	F5	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
BLE	2F	REL	OP	OP+1	FFFF			3 - 3 = 0
BLS	23	REL	OP	OP+1	FFFF			3 - 3 = 0
BLT	2D	REL	OP	OP+1	FFFF			3 - 3 = 0
BMI	2B	REL	OP	OP+1	FFFF			3 - 3 = 0
BNE	26	REL	OP	OP+1	FFFF			3 - 3 = 0
BPL	2A	REL	OP	OP+1	FFFF			3 - 3 = 0
BRA	20	REL	OP	OP+1	FFFF			3 - 3 = 0
BRCLR	13	DIR	OP	OP+1	00dd	OP+1	OP+2	6 - 6 = 0
BRCLR	1F	IND,X	OP	OP+1	IX+ff	OP+2	OP+3	7 - 6 = 1
BRCLR	18 1F	IND,Y	OP	OP+1	IY+ff	OP+2	OP+3	8 - 7 = 1
BRN	21	REL	OP	OP+1	FFFF			3 - 3 = 0
BRSET	12	DIR	OP	OP+1	00dd	OP+2	OP+3	6 - 6 = 0
BRSET	1E	IND,X	OP	OP+1	IX+ff	OP+2	OP+3	7 - 6 = 1
BRSET	18 1E	IND,Y	OP	OP+1	IY+ff	OP+2	OP+3	8 - 7 = 1
BSET	14	DIR	OP	OP+1	00dd	OP+2	00dd	6 - 5 = 1
BSET	1C	IND,X	OP	OP+1	IX+ff	OP+2	IX+ff	7 - 5 = 2
BSET	18 1C	IND,Y	OP	OP+1	IY+ff	OP+2	IY+ff	8 - 6 = 2
BSR	8D	REL	OP	OP+1	Sub	SP	SP-1	6 - 5 = 1
BVC	28	REL	OP	OP+1	FFFF			3 - 3 = 0
BVS	29	REL	OP	OP+1	FFFF			3 - 3 = 0
CBA	11	INH	OP	OP+1				2 - 2 = 0

CLC	0C	INH	OP	OP+1				2 - 2 = 0
CLI	0E	INH	OP	OP+1				2 - 2 = 0
CLR	6F	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
CLR	18 6F	IND,Y	OP	OP+1	IY+ff	IY+ff		7 - 5 = 2
CLR	7F	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
CLRA	4F	INH	OP	OP+1				2 - 2 = 0
CLRB	5F	INH	OP	OP+1				2 - 2 = 0
CLV	0A	INH	OP	OP+1				2 - 2 = 0
CMPA	81	IMM	OP	OP+1				2 - 2 = 0
CMPA	91	DIR	OP	OP+1	00dd			3 - 3 = 0
CMPA	A1	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
CMPA	18 A1	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
CMPA	B1	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
CMPB	C1	IMM	OP	OP+1				2 - 2 = 0
CMPB	D1	DIR	OP	OP+1	00dd			3 - 3 = 0
CMPB	E1	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
CMPB	18 E1	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
CMPB	F1	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
COM	63	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
COM	18 63	IND,Y	OP	OP+1	IY+ff	IY+ff		7 - 5 = 2
COM	73	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
COMA	43	INH	OP	OP+1				2 - 2 = 0
COMB	53	INH	OP	OP+1				2 - 2 = 0
CPD	1A 83	IMM	OP	OP+1	OP+2			5 - 4 = 1
CPD	1A 93	DIR	OP	OP+1	00dd	00dd+1		6 - 5 = 1
CPD	1A A3	IND,X	OP	OP+1	IX+ff	IX+ff+1		7 - 5 = 2
CPD	CD A3	IND,Y	OP	OP+1	IY+ff	IY+ff+1		7 - 5 = 2
CPD	1A B3	EXT	OP	OP+1	OP+2	hhll	hhll+1	7 - 6 = 1
CPX	8C	IMM	OP	OP+1	OP+2			4 - 3 = 1
CPX	9C	DIR	OP	OP+1	00dd	00dd+1		5 - 4 = 1
CPX	AC	IND,X	OP	OP+1	IX+ff	IX+ff+1		6 - 4 = 2
CPX	CD AC	IND,Y	OP	OP+1	IY+ff	IY+ff+1		7 - 5 = 2
CPX	BC	EXT	OP	OP+1	OP+2	hhll	hhll+1	6 - 5 = 1
CPY	18 8C	IMM	OP	OP+1	OP+2			5 - 4 = 1
CPY	18 9C	DIR	OP	OP+1	00dd	00dd+1		6 - 5 = 1
CPY	1A AC	IND,X	OP	OP+1	IX+ff	IX+ff+1		7 - 5 = 2
CPY	18 AC	IND,Y	OP	OP+1	IY+ff	IY+ff+1		7 - 5 = 2
CPY	18 BC	EXT	OP	OP+1	OP+2	hhll	hhll+1	7 - 6 = 1
DAA	19	INH	OP	OP+1				2 - 2 = 0
DEC	6A	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
DEC	18 6A	IND,Y	OP	OP+1	IY+ff	IY+ff		7 - 5 = 2
DEC	7A	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
DECA	4A	INH	OP	OP+1				2 - 2 = 0
DECB	5A	INH	OP	OP+1				2 - 1 = 0
DES	34	INH	OP	OP+1	SP			3 - 3 = 0
DEX	9	INH	OP	OP+1	FFFF			3 - 3 = 0
DEY	18 09	INH	OP	OP+1	OP+2	FFFF		4 - 4 = 0
EORA	88	IMM	OP	OP+1				2 - 2 = 0
EORA	98	DIR	OP	OP+1	00dd			3 - 3 = 0
EORA	A8	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
EORA	18 A8	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
EORA	B8	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
EORB	C8	IMM	OP	OP+1				2 - 2 = 0
EORB	D8	DIR	OP	OP+1	00dd			3 - 3 = 0

EORB	E8	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
EORB	18 E8	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
EORB	F8	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
FDIV								*
IDIV								*
INC	6C	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
INC	18 6C	IND,Y	OP	OP+1	IY+ff	IY+ff		7 - 5 = 2
INC	7C	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
INCA	4C	INH	OP	OP+1				2 - 2 = 0
INCB	5C	INH	OP	OP+1				2 - 2 = 0
INS	31	INH	OP	OP+1	SP			3 - 3 = 0
INX	8	INH	OP	OP+1	FFFF			3 - 3 = 0
INY	18 08	INH	OP	OP+1	OP+2	FFFF		4 - 4 = 0
JMP	6E	IND,X	OP	OP+1	FFFF			3 - 3 = 0
JMP	18 6E	IND,Y	OP	OP+1	OP+2	FFFF		4 - 4 = 0
JMP	7E	EXT	OP	OP+1	OP+2			3 - 3 = 0
JSR	9D	DIR	OP	OP+1	00dd	SP	SP-1	5 - 5 = 0
JSR	AD	IND,X	OP	OP+1	IX+ff	SP	SP-1	6 - 5 = 1
JSR	18 AD	IND,Y	OP	OP+1	OP+2	IY+ff	SP	7 - 6 = 1
JSR	BD	EXT	OP	OP+1	OP+2	hhll	SP	6 - 6 = 0
LDAA	86	IMM	OP	OP+1				2 - 2 = 0
LDAA	96	DIR	OP	OP+1	00dd			3 - 3 = 0
LDAA	A6	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
LDAA	18 A6	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
LDAA	B6	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
LDAB	C6	IMM	OP	OP+1				2 - 2 = 0
LDAB	D6	DIR	OP	OP+1	00dd			3 - 3 = 0
LDAB	E6	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
LDAB	18 E6	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
LDAB	F6	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
LDD	CC	IMM	OP	OP+1	OP+2			3 - 3 = 0
LDD	DC	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
LDD	EC	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
LDD	18 EC	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
LDD	FC	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
LDS	8E	IMM	OP	OP+1	OP+2			3 - 3 = 0
LDS	9E	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
LDS	AE	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
LDS	18 AE	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
LDS	BE	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
LDX	CE	IMM	OP	OP+1	OP+2			3 - 3 = 0
LDX	DE	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
LDX	EE	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
LDX	CD EE	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
LDX	FE	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
LDY	18 CE	IMM	OP	OP+1	OP+2			4 - 4 = 0
LDY	18 DE	DIR	OP	OP+1	00dd	00dd+1		5 - 5 = 0
LDY	1A EE	IND,X	OP	OP+1	IX+ff	IX+ff+1		6 - 5 = 1
LDY	18 EE	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
LDY	18 FE	EXT	OP	OP+1	OP+2	hhll	hhll+1	6 - 6 = 0
LSR	64	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
LSR	18 64	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff	7 - 5 = 2
LSR	74	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
LSRA	44	INH	OP	OP+1				2 - 2 = 0

LSRB	54	INH	OP	OP+1				2 - 2 = 0
LSRD	4	INH	OP	OP+1	FFFF			3 - 3 = 0
MUL	3D	INH	OP					10 - 10 = 0
NEG	60	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
NEG	18 60	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff	7 - 5 = 2
NEG	70	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
NEGA	40	INH	OP	OP+1				2 - 2 = 0
NEGB	50	INH	OP	OP+1				2 - 2 = 1
NOP	1	INH	OP	OP+1				2 - 2 = 0
ORAA	8A	IMM	OP	OP+1				2 - 2 = 0
ORAA	9A	DIR	OP	OP+1	00dd			3 - 3 = 0
ORAA	AA	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ORAA	18 AA	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ORAA	BA	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
ORAB	CA	IMM	OP	OP+1				2 - 2 = 0
ORAB	DA	DIR	OP	OP+1	00dd			3 - 3 = 0
ORAB	EA	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
ORAB	18 EA	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
ORAB	FA	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
PSHA	36	INH	OP	(OP+1)	SP			3 - 3 = 0
PSHB	37	INH	OP	(OP+1)	SP			3 - 3 = 0
PSHX	3C	INH	OP	(OP+1)	SP	SP-1		4 - 4 = 0
PSHY	18 3C	INH	OP	(OP+1)	OP+2	SP	SP-1	5 - 5 = 0
PULA	32	INH	OP	(OP+1)	SP+1			4 - 3 = 1
PULB	33	INH	OP	(OP+1)	SP+1			4 - 3 = 1
PULX	38	INH	OP	(OP+1)	SP+1	SP+2		5 - 4 = 1
PULY	18 38	INH	OP	(OP+1)	(OP+2)	SP+1	SP+2	6 - 5 = 1
ROL	69	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
ROL	18 69	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff	7 - 5 = 2
ROL	79	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
ROLA	49	INH	OP	OP+1				2 - 2 = 0
ROLB	59	INH	OP	OP+1				2 - 2 = 0
ROR	66	IND,X	OP	OP+1	IX+ff	IX+ff		6 - 4 = 2
ROR	18 66	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff	7 - 5 = 2
ROR	76	EXT	OP	OP+1	OP+2	hhll	hhll	6 - 5 = 1
RORA	46	INH	OP					2 - 2 = 0
RORB	56	INH	OP					2 - 2 = 0
RTI	3B	INH	OP	(OP+1)	SP+1	SP+2	(...)	12 - 9 = 3
RTS	39	INH	OP	SP+1	SP+2			5 - 3 = 2
SBA	10	INH	OP	OP+1				2 - 2 = 0
SBCA	82	IMM	OP	OP+1				2 - 2 = 0
SBCA	92	DIR	OP	OP+1	00dd			3 - 3 = 0
SBCA	A2	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
SBCA	18 A2	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
SBCA	B2	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
SBCB	C2	IMM	OP	OP+1				2 - 2 = 0
SBCB	D2	DIR	OP	OP+1	00dd			3 - 3 = 0
SBCB	E2	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
SBCB	18 E2	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
SBCB	F2	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
SEC	0D	INH	OP	OP+1				2 - 2 = 0
SEI	0F	INH	OP	OP+1				2 - 2 = 0
SEV	0B	INH	OP	OP+1				2 - 2 = 0
STAA	97	DIR	OP	OP+1	00dd			3 - 3 = 0

STAA	A7	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
STAA	18 A7	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
STAA	B7	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
STAB	D7	DIR	OP	OP+1	00dd			3 - 3 = 0
STAB	E7	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
STAB	18 E7	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
STAB	F7	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
STD	DD	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
STD	ED	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
STD	18 ED	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
STD	FD	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
STOP								*
STS	9F	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
STS	AF	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
STS	18 AF	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
STS	BF	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
STX	DF	DIR	OP	OP+1	00dd	00dd+1		4 - 4 = 0
STX	EF	IND,X	OP	OP+1	IX+ff	IX+ff+1		5 - 4 = 1
STX	CD EF	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
STX	FF	EXT	OP	OP+1	OP+2	hhll	hhll+1	5 - 5 = 0
STY	18 DF	DIR	OP	OP+1	00dd	00dd+1		5 - 5 = 0
STY	1A EF	IND,X	OP	OP+1	IX+ff	IX+ff+1		6 - 5 = 1
STY	18 EF	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	6 - 5 = 1
STY	18 FF	EXT	OP	OP+1	OP+2	hhll	hhll+1	6 - 6 = 0
SUBA	80	IMM	OP	OP+1				2 - 2 = 0
SUBA	90	DIR	OP	OP+1	00dd			3 - 3 = 0
SUBA	A0	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
SUBA	18 A0	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
SUBA	B0	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
SUBB	C0	IMM	OP	OP+1				2 - 2 = 0
SUBB	D0	DIR	OP	OP+1	00dd			3 - 3 = 0
SUBB	E0	IND,X	OP	OP+1	FFFF	IX+ff		4 - 4 = 0
SUBB	18 E0	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	5 - 5 = 0
SUBB	F0	EXT	OP	OP+1	OP+2	hhll		4 - 4 = 0
SUBD	83	IMM	OP	OP+1	OP+2			4 - 3 = 1
SUBD	93	DIR	OP	OP+1	00dd	00dd+1		5 - 4 = 1
SUBD	A3	IND,X	OP	OP+1	IX+ff	IX+ff+1		6 - 4 = 2
SUBD	18 A3	IND,Y	OP	OP+1	OP+2	IY+ff	IY+ff+1	7 - 5 = 2
SUBD	B3	EXT	OP	OP+1	OP+2	hhll	hhll+1	6 - 5 = 1
SWI	3F	INH	OP	(OP+1)	SP	SP-1	(...)	14 - 14 = 0
TAB	16	INH	OP	OP+1				2 - 2 = 0
TAP	6	INH	OP	OP+1				2 - 2 = 0
TBA	17	INH	OP	OP+1				2 - 2 = 0
TEST	0	INH	OP	(...)				*
TPA	7	INH	OP	OP+1				2 - 2 = 0
TST	6D	IND,X	OP	OP+1	FFFF	IX+ff		6 - 4 = 2
TST	18 6D	IND,Y	OP	OP+1	OP+2	FFFF	IY+ff	7 - 5 = 2
TST	7D	EXT	OP	OP+1	OP+2	hhll		6 - 4 = 2
TSTA	4D	INH	OP	OP+1				2 - 2 = 0
TSTB	5D	INH	OP	OP+1				2 - 2 = 0
TSX	30	INH	OP	OP+1	SP			3 - 3 = 0
TSY	18 30	INH	OP	OP+1	OP+2	SP		4 - 4 = 0
TXS	35	INH	OP	OP+1	FFFF			3 - 3 = 0
TYS	18 35	INH	OP	OP+1	OP+2	FFFF		4 - 4 = 0

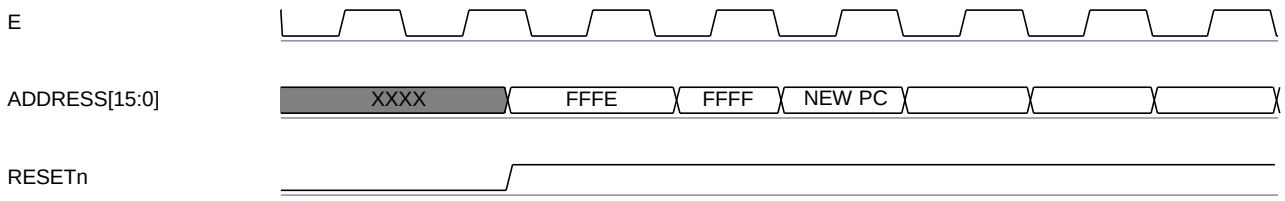
WAI	3E	INH	OP	(OP+1)	SP	SP-1	(...)	12 - 12 = 0
XGDX	8F	INH	OP	OP+1	FFFF			3 - 3 = 0
XGDY	18 8F	INH	OP	OP+1	OP+2	FFFF		4 - 4 = 0

NOMENCLATURE

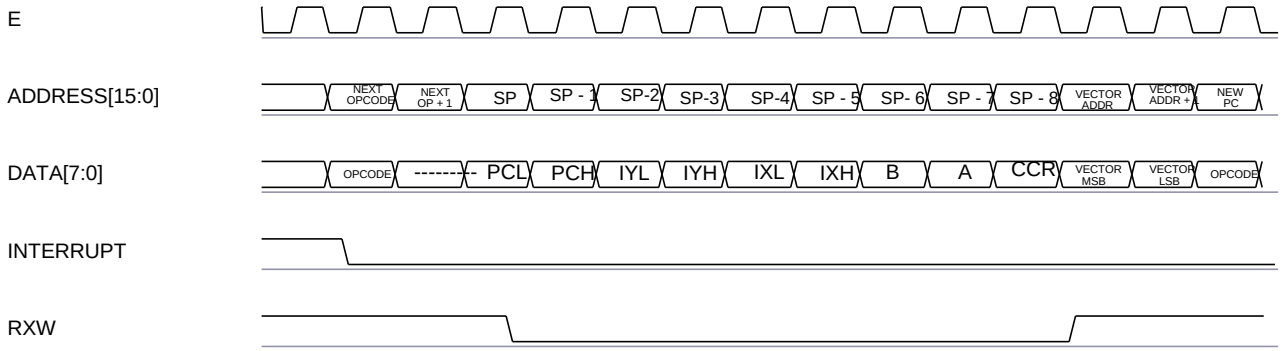
dd =Low Order 8 Bits of Direct Address $0000-$00FF
(High Byte Assumed to be $00)
ff = 8 Bits Forward Offset $00 to $FF (It is added to Index)
hh = High Order Byte of 16-Bit Extended Address
ll = Low Order Byte of 16-Bit Extended Address
IX = Address Pointed to by Index Register X Value
IY = Address Pointed to by Index Register Y Value
OP = Address of Opcode Byte
OP + n = Address of nth Location After Opcode Byte
SP = Address Pointed to by Stack Pointer Value (at the Start of the Instruction)
SP + n = Address of nth Higher Address Past That Pointed to by Stack Pointer
SP - n = Address of nth Lower Address Before That Pointed to by Stack Pointer

the addressing mode of the instruction:
INH(erent)
IMM(ediate)
DIR(ect)
EXT(ended)
IND(exed)
REL(ative)

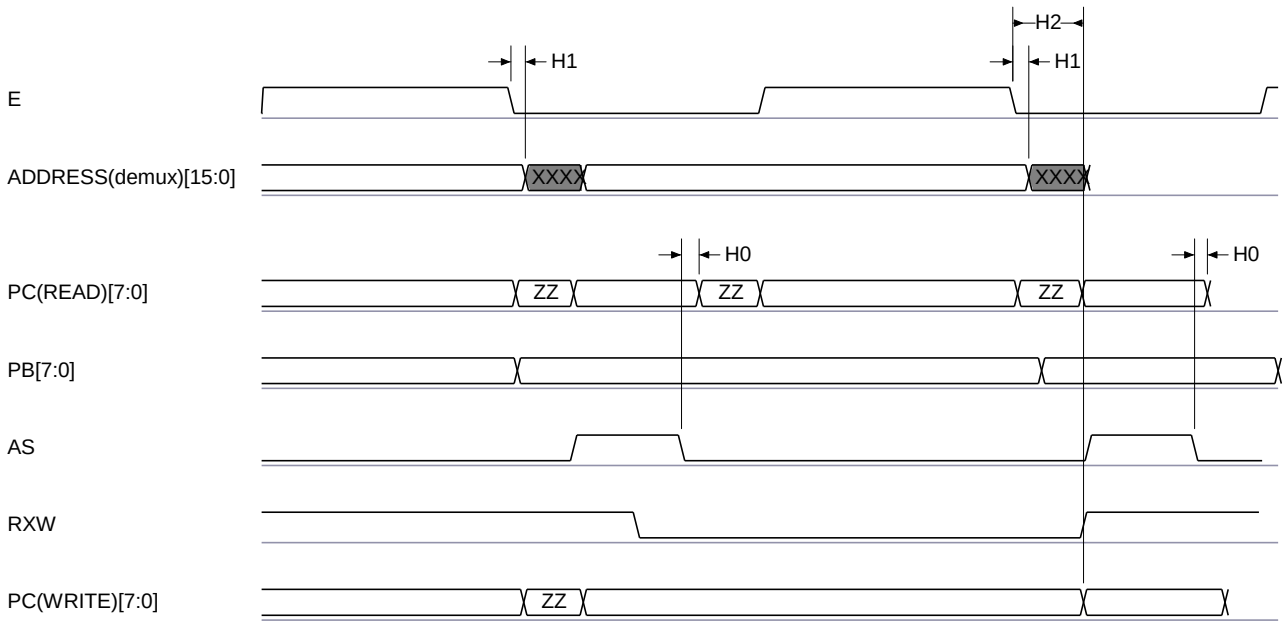
SECTION 8
Timing Diagrams



External Reset Timing diagram



Interrupt Timing diagram



Read Write timing diagram

SECTION 9
Compilation and Simulation Notes

The model follows the IEE 1076-87 VHDL standard.

9.1 C68MX11 Compilation

Source files for the model are in the directory \c68mx11\src

The order of compilation is as follows:

1. Definitions package
   \c68mx11\src
   \c6811_pack.vhd
2. Components for model core
   \c68mx11\src\
   \sequencer.vhd
   \mpu.vhd
   \sci.vhd
   \spi.vhd
   \timer_11.vhd
   \ram.vhd
   \port_reg_logic.vhd
   \c68mx11_core.vhd
   \c68mx11.vhd

   Sequencer.vhd have to be compiled before mpu.vhd
   C68mx11_core.vhd have to be compiled after all other components
   C68mx11.vhd is the top level file in case the tristate buffer have to be added
   Please refer to Fig. 1 (Symbol)

Compilation order can be found in :

C68mx11\tb\tests\test_mpu\compile.do :MTI Compilation Macro

9.2 Test Bench Description

9.2.1 Test Bench environment description

- Device under Test
  All files described in above section 9.1

Test Bench definition package
  C68mx11\tb \c6811tb_pack.vhd

- External 32 K ROM
  C68mx11\tb \extrom.vhd
- External 4K RAM
  C68mx11\tb\ extram.vhd

9.3 Simulation Notes

Several Test Benches are provided to test functionality of MPU and all peripheral.

These various test benches have to be compiled and simulated depending on which kind of functionality have to be tested.
These test-benches can be used for both functional and post synthesis simulation.

9.4 Functional and Post-Synthesis Simulations

9.4.1 CPU

All function of CPU can be simulated using
C68mx11\tb\c68mx11tb.vhd

Compilation order can be found in :
C68mx11\tb\tests\test_mpu\ compile.do :MTI Compilation Macro

Compile.do will compile all VHDL files necessary for functional simulation including Test Bench files. Files for CPU test to be used with C68mx11\tb c68mx11tb.vhd can be found in the subdirectories:
C68mx11\tb\tests\test_mpu\..
C68mx11\tb\tests\test_ports\..
Each subdirectory contains a set of files for each simulation.

9.4.2

Before starting the simulation, copy the following test files into the C68mx11\tb\tests\default directory:

C68MX11.ref
C68MX11.stm
Extrom.txt

9.4.3

Once the Test bench is loaded into the simulator , for MTI use the command "run –all".
At the end of the simulation the following message should appear:

"Note: test completed with no errors"

9.4.4

If your synthesis tool renames the top level pins after routing, it may be necessary to create a dummy wrapper in order to use the supplied test-bench.

Edit the file of test-bench
Add the actual maximum expected delays of your silicon
Compile your post-synthesis or post-layout VHDL netlist file
Recompile the test-bench file just edited
Simulate as above

9.4.5 SCI

All function of CPU can be simulated using
C68mx11\tb c68mx11tb_sci.vhd

Compilation order can be found in :
C68mx11\tb\tests\test_sci\ compile.do:MTI Compilation Macro

Compile.do will compile all VHDL files necessary for functional simulation including Test Bench files.

Files for SCI test to be used with C68mx11\tb\c68mx11tb_sci.vhd can be found in the subdirectories:
C68mx11\tb\tests\test_sci\..
Each subdirectory contains a set of files for each simulation.
Repeat step 9.4.2, 9.4.3 for functional simulation and 9.4.2 and 9.4.4 for post-synthesis simulation.

9.4.6 SPI

All function of SCI can be simulated using
C68mx11\tb\test_spi.vhd
C68mx11\tb\c68mx11tb_spi.vhd

Compilation order can be found in :
C68mx11\tb\tests\test_spi\compile.do :MTI Compilation Macro

Compile.do will compile all VHDL files necessary for functional simulation including Test Bench files. Files for SCI test to be used with C68mx11\tb c68mx11tb_spi.vhd can be found in the subdirectories:

C68mx11\tb\tests\test_spi\..

Each subdirectory contains a set of files for each simulation.
Repeat step 9.4.2, 9.4.3 for functional simulation and 9.4.2 and 9.4.4 for post-synthesis simulation

9.4.6 TIMER UNIT, RTI , PULSE ACCUMULATOR

All function of TIMER UNIT, RTI and PULSE ACCUMULATOR can be simulated using
C68mx11\tb \c68mx11tb_tim.vhd

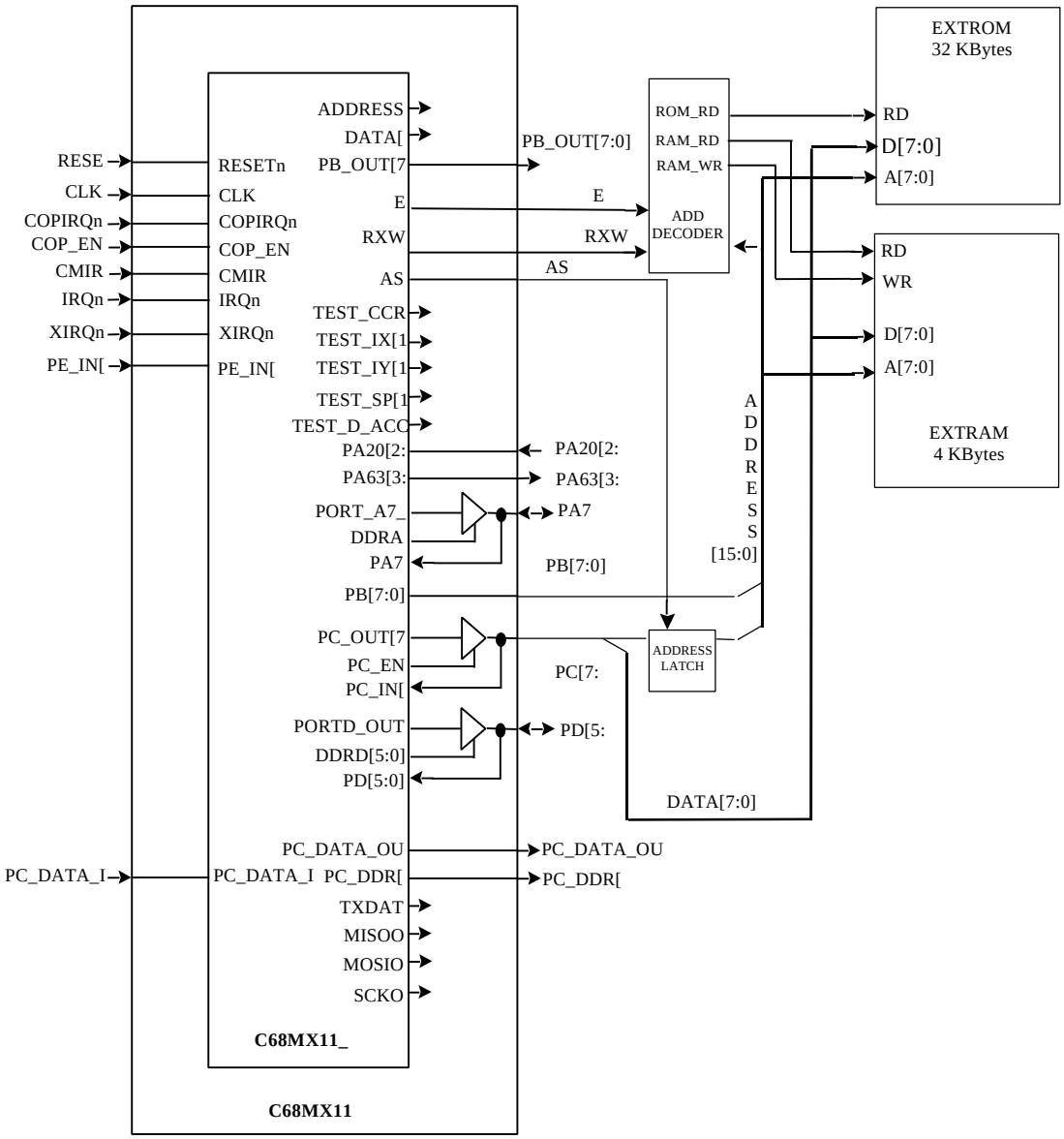
Compilation order can be found in :
C68mx11\tb\tests\test_tim\compile.do:MTI Compilation Macro

Compile.do will compile all VHDL files necessary for functional simulation including Test Bench files.

Files for TIMER UNIT, RTI and PULSE ACCUMULATOR test to be used with C68mx11\tb\c68mx11tb_tim.vhd can be found in the subdirectories:

C68mx11\tb\tests\test_tim\..

Each subdirectory contains a set of files for each simulation.
Repeat step 9.4.2, 9.4.3 for functional simulation and 9.4.2 and 9.4.4 for post-synthesis simulation


EXTROM
32 KBytes
RD
D[7:0]
A[7:0]
ROM_RD
RAM_RD
RAM_WR
ADD
DECODER
RD
WR
D[7:0]
A[7:0]
EXTRAM
4 KBytes
ADDRESS
[15:0]
ADDRESS
LATCH
DATA[7:0]
PB_OUT[7:0]
E
RXW
AS
PA20[2:
PA63[3:
PA7
PB[7:0]
PC[7:
PD[5:
RESE
CLK
COPIRQn
COP_EN
CMIR
IRQn
XIRQn
PE_IN[
ADDRESS
DATA[
RESETn
CLK
COPIRQn
COP_EN
CMIR
IRQn
XIRQn
PE_IN[
PB_OUT[7
E
RXW
AS
TEST_CCR
TEST_IX[1
TEST_IY[1
TEST_SP[1
TEST_D_ACC
PA20[2:
PA63[3:
PORT_A7_
DDRA
PA7
PB[7:0]
PC_OUT[7
PC_EN
PC_IN[
PORTD_OUT
DDRD[5:0]
PD[5:0]
PC_DATA_I
PC_DATA_OU
PC_DDR[
TXDAT
MISOO
MOSIO
SCKO
C68MX11_
C68MX11

SECTION 10

Support

Every effort has been made to ensure that this core functions correctly. If a problem is encountered please contact MOXSYN:

Technical Support Hotline: +39-335-65-74-399
Fax: +39-736-43125
E-mail: mailto:support@moxsyn.com
URL : www.moxsyn.com

Related Information

M68HC11RM/AD
M68HC11
Reference Manual
Rev. 3.0
Contact:
Motorola Technical Information Center (TIC)
Phone +1-800-521-6274,
URL : http://mot-sps.com/support/technical/helpline.html.
http://www.mcu.motsps.com/contact.html