Chapter 16

Motorola MC68HC11 Family
MCU Architecture
Lesson 1

68HC11 MCU Architecture overview
CPU Registers

- ACCA and ACCB
- 16-bit Program Counter PC
- 16-bit Stack Pointer S
- 16-bit Index Registers IX and IY
- 8-bit CCR (Condition Code Register)
Microcontroller 68HC11

- 8-bit
- 8 MHz XTAL
- ACCA and ACCB
- Fully MOS-FET Based
- 16-bit Program Counter PC
- 16-bit Index Registers IX and IY
- 16-bit Stack Pointer S
- Clock Reducible to 0
- Change after push, increase before pop

Internal RAM and ROM, RAM, Registers, EEPROM

Internal Bus

8-bit XTAL

ACC, ACCB

Fully MOS-FET Based

16-bit Program Counter

16-bit Index Registers IX and IY

16-bit Stack Pointer S

Decrease after push, increase before pop

Accumulators

Stop Instruction

Internal RAM and ROM, RAM, Registers, EEPROM
MCU Architecture overview

• Fully static operation- MCU clock can be reduced to 0 – being fully MOSFETs based

Stop Instruction
Performance

As per 8MHz XTL 2MHz E-Clock
• 68HC11 common 8-bit internal bus for the 16-bit addressing and 8-bit data
• Princeton architecture bus
• Bus interface for 8-bit data and instructions.
• Two interrupts – Maskable Interrupt request (IRQ) and Initialization option as unmaskable (XIRQ)
• PC initialization using reset vector FFFEH-FFEFH
68HC11 Architecture Overview
64 kB of linear address space

Address

68HC11/12

IO and internal devices, System control Registers

Internal RAM and ROM, RAM, EEPROM

Internal devices Registers Data and Program, constants, stored tables Common Memory

Memory Architecture
68HC11 Family Programming Model

- CPU Registers
- IO and internal devices Registers
- System Function Control Registers
- Internal RAM and ROM, RAM, EEPROM
- OPTION, COPRS, IPRO, INIT, CONFIG, TESTI

Address Space
68HC11 Family Programming Model

- Instructions have 8 data type
- Few use 16-bit data types
- Processor operation of 16-bit data type is as per word alignment at memory in big endian [least significant byte stored as higher bits (address 1) of a word]
16-bit word alignment

A 16bit word in memory

<table>
<thead>
<tr>
<th>Big Endian</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte1 (MSB)</td>
<td>Address0</td>
</tr>
<tr>
<td>Byte0 (LSB)</td>
<td>Address1</td>
</tr>
</tbody>
</table>
68HC11 Address and Data Buses
Internal and External Buses

- ACCA or ACCB
- IY
- IX
- CCR
- Internal Devices
- S
- PC

Internal Bus  8-bit

- MDR
- MAR
- AD0 – AD7
- A8 - A15

Bus Interface Unit
Execution Unit- ALU

• ACCA or ACCB used as 8-bit ACC or as 16-bit ACCD (double accumulator)
• 8-bit ALU includes multiplier and divide
## Execution Unit

- **IR**
- **ID**
- **Control and Sequencer Circuits**
  - ACCA or ACCB
  - IX
  - IY
  - Internal RAM and ROM, RAM, Registers, EEPROM
- **Temp 1**
- **Temp 2**
- **ALU**
  - OR, AND, XOR
  - Rotate/Shift
  - CMP, TST
  - +, -, x or ÷
Instruction Execution

STAGE 1

Instruction Fetch

STAGE 2

Instruction Decode

STAGE 3 to n

Instruction Execute

Time

clock cycle (s)
Internal bus

Fetch

Decode

Execution

IR

ID

Control and Sequencer Circuits

Control memory micro codes based - Implementation
Port PB

Address – x’004H

PB7  PB6  PB5  PB4  PB3  PB2  PB1  PB0

0  0  0  0  0  0  0

Option 1

A15  A14  A13  A12  A11  A10  A9  A8

Option 2

004H  004H
Port PC

Address – x’007H

- DDRC7
- DDRC6
- DDRC5
- DDRC4
- DDRC3
- DDRC2
- DDRC1
- DDRC0

PC7
PC6
PC5
PC4
PC3
PC2
PC1
PC0

IO

A7
A6
A5
A4
A3
A2
A1
A0

Handshake signals for Port

Option 1

Option 2

Option 3

D7
D6
D5
D4
D3
D2
D1
D0
Port PD

Address – x’009H – x’008H

<table>
<thead>
<tr>
<th>DDRD5</th>
<th>DDRD4</th>
<th>DDRD3</th>
<th>DDRD2</th>
<th>DDRD1</th>
<th>DDRD0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD5</td>
<td>PD4</td>
<td>PD3</td>
<td>PD2</td>
<td>PD1</td>
<td>PD0</td>
</tr>
</tbody>
</table>

Option 1

<table>
<thead>
<tr>
<th>IO</th>
<th>SS</th>
<th>SCLK</th>
<th>MOSI</th>
<th>MISO</th>
<th>TxD</th>
<th>RxD</th>
</tr>
</thead>
</table>

Option 2

Master

Slave

Option 3

Option 4

Option 5

Receiving

SCI
Port PE

Address – x’00AH

AMUX
S/H
ADC

ADC Register
ADC TL Register

PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0

I I I I

AN7 AN6 AN5 AN4 AN3 AN2 AN1 AN0

Option 1

Analog Input Option 2
Expansion Mode

- Expanded Mode used for interfacing External Memory and Ports
- Expanded Mode looses Port B and Port B
- Recovered back by using 68HC24
Expansion Mode

A15
A14
A13
A12
A11
A10
A9
A8

Port B Option 2

Mode A=0

Mode B=1

AD7
AD6
AD5
AD4
AD3
AD2
AD1
AD0

Port C Options 2/3

Latch

Address Decoder

CS0

CS1

Expansion to Ports and Memory

An-A15

A0-A15

D0-D7
Lost Ports Replacement Unit

68HC24

Handshake signals for Port

Port B

Port C

Recovered Ports B and Port C

Mode A=0

Mode B=1

Microcontrollers... 2nd Ed. Raj Kamal
Pearson Education
Summary
We learnt

- 68HC11 family 8-bit processor
- PC, S, ACCA, ACCB, IX, IY, CCR
- Princeton architecture
- 8 bit data type
- Big endian 16-bit word alignment
- Two interrupts – Maskable Interrupt request (IRQ) and Initialization option as unmaskable (XIRQ)
- PC initialization using reset vector
We learnt

**Internal and External Memory Addresses**

- IO/Devices Control and Status Registers
- System Function Control Registers
- Internal RAM
- Internal ROM
- EEPROM
We learnt

Internal Devices

• TCNT with out compare, input capture,
• SPI, SCI
• RTC
• PACNT
• Port E with option of analog inputs multi channel AMUX, S/H, ADC
We learnt

Internal Devices

• Port B /A8-A15,
• Port C with DDRC / AD0-AD7/
• Port A /IC1-IC3, OC1-OC5
• Port D with DDRD/SCI/SPI Master/Slave
End of Lesson 1 on MCU
Architecture overview