Peripheral Hardware
Ch. 5

ECE331

rev: Spring 2013

Outline

- Preview 689HC12 peripheral blocks
- I/O addressing schemes
- Configuration registers
- I/O ports
- "Modes of operation"
- External memory interface
- Memory cells, types, arrays

68HC12 Peripheral Blocks

peripheral = "on the side"
peripheral hardware = outside of the core (CPU)

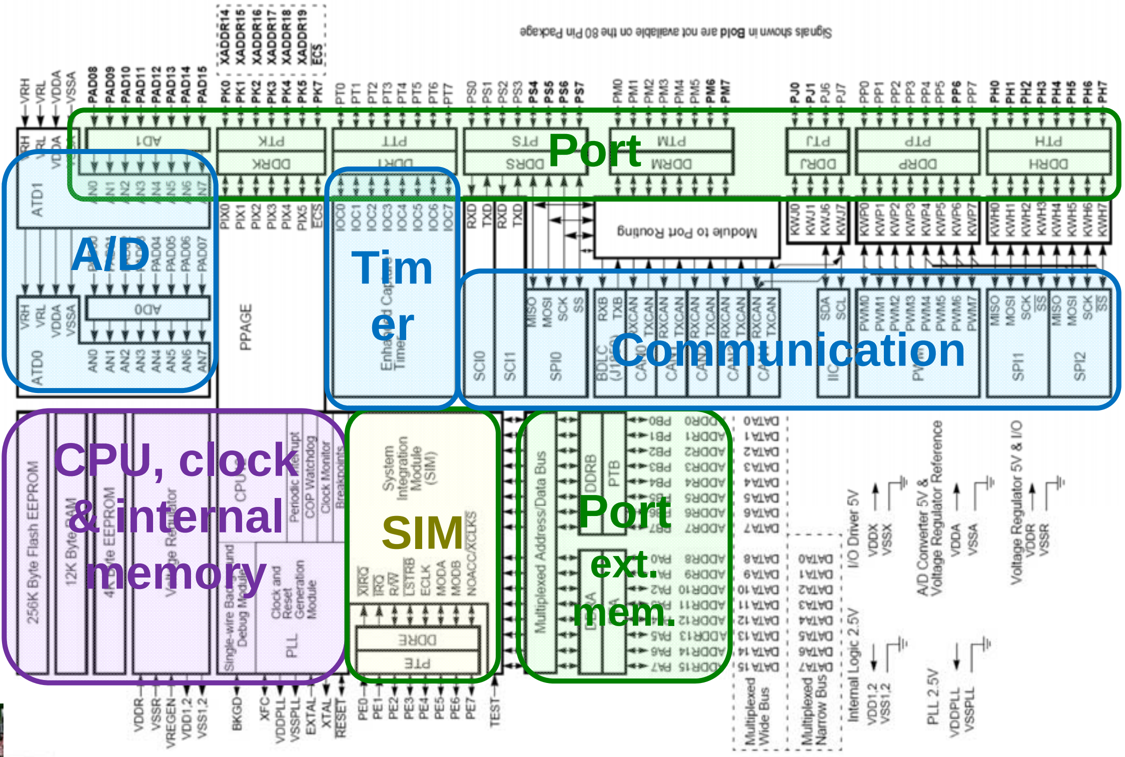
- Preview of 68HC12 Peripherals
  - see MC9S12DP256 diagram in HO_7 (and next slide)

  observe:

  - CPU12, clock module & internal memory
    - small part of the microcontroller
  - System integration module
    - mode setup, interrupts, external mem. interface
  - Data ports & data direction registers
    - generic digital I/O & external ports for peripheral functions
  - Timer
  - Communication ports (SPI, CAN, I2C)
  - Analog to digital converter, A/D (ATD)

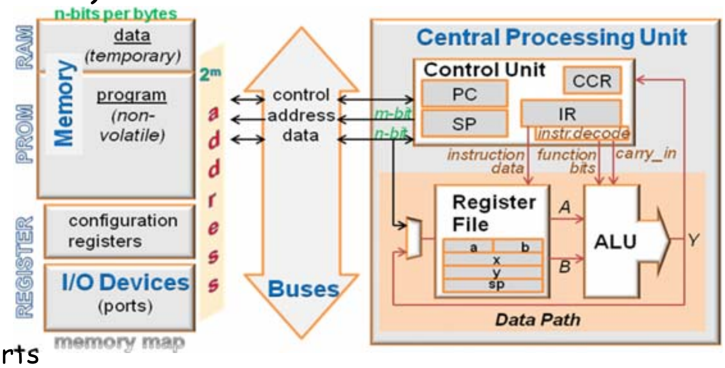
} future topics

MC9S12DP256 Block Diagram


Port
A/D
Timer
Communication
CPU, clock & internal memory
SIM
Port
ext. mem.
Signals shown in Bold are not available on the 80 Pin Package

Memory Map Architecture

- Detailed Architecture (68HC12)
  - CPU w/ HC12 register file
    - connection details added
  - Buses
    - n=8 data lines
    - m=16 address lines
  - Memory & I/O
    - data RAM
    - program PROM
    - configuration registers & ports
    - all “mapped” to 2^m addresses
    - → memory map = assignment of addresses to different memory blocks



- Configuration Registers
  - microcontroller memory registers containing function control signals
    - EX: enable/disable interrupts, set timer speed, etc.
  - port control and port data registers
    - Port = external input/output signals; interface to outside of microcontroller

I/O Addressing Schemes

How does software talk to peripherals?

- Two common “addressing schemes”
  - vary with µC brand
- <u>Memory Mapped I/O</u>
  - address bus size determines max size of memory
  - memory partitioned into 2 functional blocks
    - user memory (for program/data bytes)
      - SRAM, EEPROM, Flash EEPROM
    - I/O register memory (for control of I/O peripheral functions)
      - SRAM or FF registers (volatile: lose value when no power)
  - HC12 is memory mapped scheme

I/O Addressing Schemes

- Isolated I/O
  - separate set of addresses for a) memory and b) I/O registers
  - requires additional control signal to select from a) & b)
    - M/IO: 0=I/O, 1=Memory
  - Intel (and others) use isolated I/O scheme

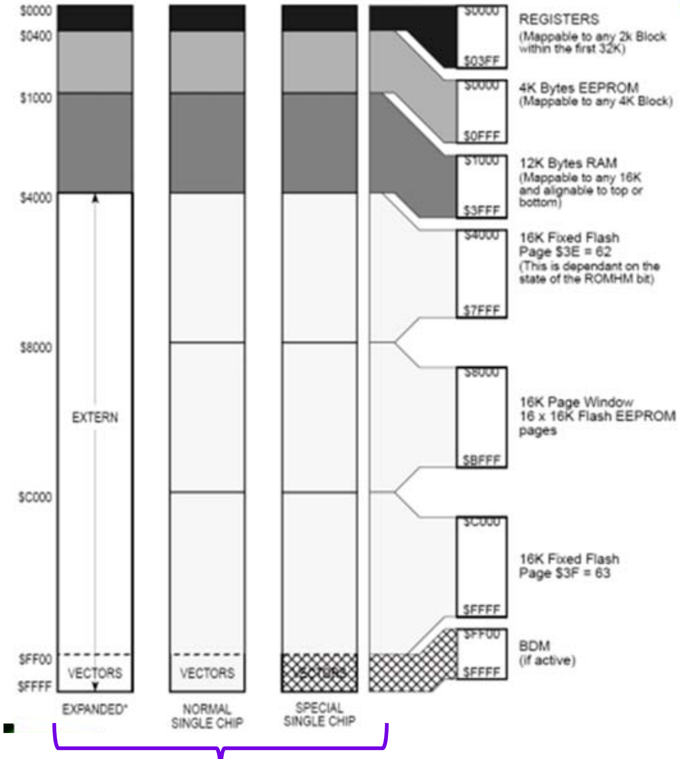
- Pros/Cons of Isolated I/O

☺ – more addresses available for I/O devices

☺ – no memory space lost to I/O devices

☹ – require extra control signal (pin) in addition to addr. bus

MC9S12DP256 Memory Map



"modes of operation"

(HO_7 page 3)

in Normal Single Chip mode

- "Configuration" Registers
  - $0000 - $03FF
- RAM
  - $1000 - $3FFF
  - volatile: temp. data storage
- Flash EEPROM
  - $4000 - $7FFF
  - non-volatile: program storage
  - you will use this memory
- More Flash EEPROM
  - $8000 - $FF00
- Interrupt Vectors
  - $FF00 - $FFFF

Configuration Registers

- Configuration registers (AKA "register set")
  - data registers for I/O control and access
  - architecturally part of Memory; not CPU Register File
  - addressable: can be read/written by software
    - HC12 maps to addresses $0000 - $02FF (767 bytes!)
    - some listed on HO_7 pg. 2, full list in textbook Appendix C

      EX: STAA $0033; configure Port K

      LDAA $0240; reads data on Port T
- Some configuration register functions
  - control uC configurable functions
    - EX: power save mode, watchdog timer
  - configure settings of I/O devices
    - EX: direction (I of O) of I/O port
  - enable/disable peripheral functions
    - EX: turn on A/D converter
  - read/write I/O port data

Config. registers are critical to uC operation
- will be introduced a few at a time in labs

Function/address vary with version of controller
- App B & C show 2 diff. sets

I/O Ports

- Port = set of external pins (typically 8) with input/output signals that are addressable by software
  - most uC ports are digital (except A/D or DAC)
- Bidirectional ports have a "data direction register" (DDR) that sets the direction of each bit/pin
  - bit x of DDR sets direction of bit x on data port
  - generally, port can be set with both in and out pins
- DDR settings (HC12)

  0 = input

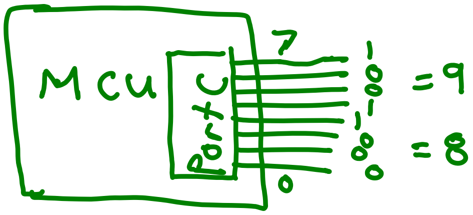
  1 = output
  - EX: store $F0 into addr. <$0033> Port K DDR

    → sets Port K: upper nibble = **output**, lower nibble = **input**

I/O Ports

- Multi-function ports
  - Ports generally wired to multiple functions
  - HC12 ports have 2 functions
    - 1) general purpose I/O,
    - 2) I/O for peripheral functions like Timer, A/D
    - varies with version of HC12
- HC12 Port C: general purpose bidirectional I/O
  - Port C Data Register <$0004>
  - Port C DDR (DDRC)   <$0006>
  - EX: Configure Port C as output & write $98

```
LDAA   #$FF
STAA   $0006; set DDRC to output
LDAA   #$98
STAA   $0004; write data to Port C
```



I/O Ports

- EX: Configure Port C so odd bits are OUTput & even bits are INput

```
DDRC   EQU    $06
       LDAA   #$AA
       STAA   DDRC
```

DDRC	7	6	5	4	3	2	1	0
	o	i	o	i	o	i	o	i
	1	0	1	0	1	0	1	0

= $AA

  - many config. registers @ $00-FF; can use Direct Addressing
- Port Latching
  - Output ports are latched (held in flip flops)
    - will drive output values until new value is set
  - Input ports constantly monitor input pins
    - must be latched externally or sampled by software & saved to memory
    - otherwise might capture wrong data if not synchronized correctly

Modes of Operation

- HC12/S12 has 8 operating modes
  - selected by pins MODC, MODB, MODA

MODC	MODB	MODA	Mode	Port A	Port B
0	0	0	special single chip	G.P. I/O	G.P. I/O
0	0	1	special expanded narrow	Addr/Data	Addr
0	1	0	special peripheral	Addr/Data	Addr/Data
0	1	1	special expanded wide	Addr/Data	Addr/Data
1	0	0	normal single chip	G.P. I/O	G.P. I/O
1	0	1	normal expanded narrow	Addr/Data	Addr
1	1	0	reserved	--	--
1	1	1	normal expanded wide	Addr/Data	Addr/Data
G.P. = general purpose					

  - 3 “normal” modes
    - some config. registers are protected (can’t read/write)
  - 4 “special” modes
    - can access all config. registers
    - used to test uC; not for typical use

Normal Operating Modes

HCS12 (what you use in lab)

- Single Chip: all I/O ports are available for general use
  - mode we will use in lab
- Expanded Wide: Ports A&B used for Addr & Data bus
  - wide = 16 bit data bus
  - multiplexed: address one cycle, data next cycle
- Expanded Narrow: Port A & B are address bus (16b)

HC12

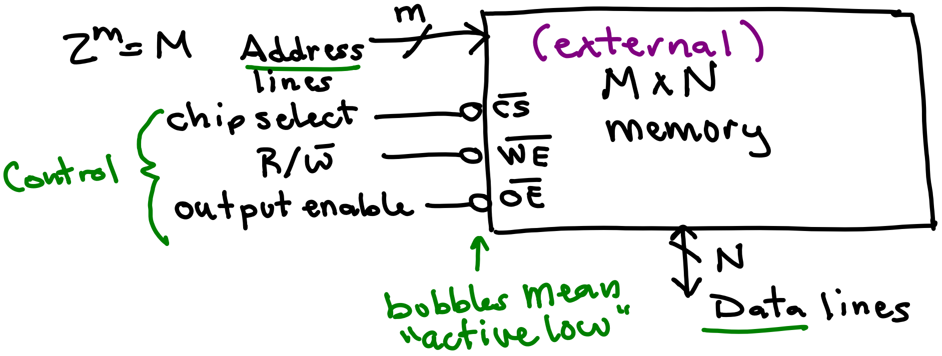
- Uses Ports A, B, C, D rather than multiplexing A&B

Expanded Modes

- Used for interfacing to **external** memory-mapped components (memory or I/O devices)
- Requires Address, Data & Control buses

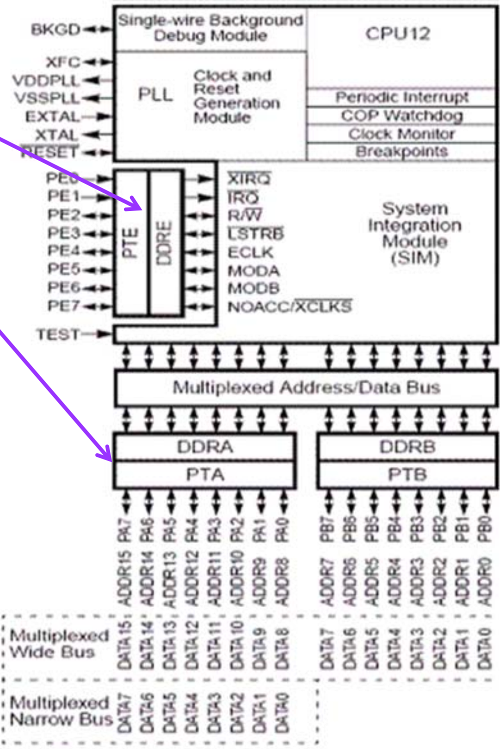
control bus

- $\overline{CS}$ = active low chip select; selects from multiple external devices
- $\overline{WE}$ = write enable
- $\overline{OE}$ = output enable; read/write handshaking



HCS12 Buses

- Port E, control port
- Port A&B, addr/data


BKGD
Single-wire Background Debug Module
CPU12
XFC
VDDPLL
VSSPLL
EXTAL
XTAL
RESET
PLL
Clock and Reset Generation Module
Periodic Interrupt
COP Watchdog
Clock Monitor
Breakpoints
PE0
PE1
PE2
PE3
PE4
PE5
PE6
PE7
PTE
DDRE
XIRQ
IRQ
R/W
LSTRB
ECLK
MODA
MODB
NOACC/XCLKS
System Integration Module (SIM)
TEST
Multiplexed Address/Data Bus
DDRA
PTA
DDRB
PTB
PA7 PA6 PA5 PA4 PA3 PA2 PA1 PA0
PB7 PB6 PB5 PB4 PB3 PB2 PB1 PB0
ADDR15 ADDR14 ADDR13 ADDR12 ADDR11 ADDR10 ADDR9 ADDR8
ADDR7 ADDR6 ADDR5 ADDR4 ADDR3 ADDR2 ADDR1 ADDR0
Multiplexed Wide Bus
DATA15 DATA14 DATA13 DATA12 DATA11 DATA10 DATA9 DATA8
DATA7 DATA6 DATA5 DATA4 DATA3 DATA2 DATA1 DATA0
Multiplexed Narrow Bus
DATA7 DATA6 DATA5 DATA4 DATA3 DATA2 DATA1 DATA0

16b address

16b data (wide)

8b data (narrow)