Embedded Computer Systems
Chapter 4:
Bus Based Computer System

Eng. Husam Y. Alzaq
Islamic University of Gaza
Bus-Based Computer Systems

- Busses.
- Memory devices.
- I/O devices:
  - serial links
  - timers and counters
  - keyboards
  - displays
  - analog I/O
The CPU bus

- Bus allows CPU, memory, devices to communicate.
  - Shared communication medium.

- A bus is:
  - A set of wires.
  - A communications protocol.
Bus protocols

- Bus protocol determines how devices communicate.
- Devices on the bus go through sequences of states.
  - Protocols are specified by state machines, one state machine per actor in the protocol.
- May contain asynchronous logic behavior.
Four-cycle handshake

device 1

device 2

enq

ack

device 1

device 2

1 2 3 4

5

time
Four-cycle handshake, cont’d.

1. Device 1 raises enq.
2. Device 2 responds with ack.
3. Device 2 lowers ack once it has finished.
4. Device 1 lowers enq.
Microprocessor busses

- Clock provides synchronization.
- R/W is true when reading (R/W' is false when reading).
- Address is a-bit bundle of address lines.
- Data is n-bit bundle of data lines.
- Data ready signals when n-bit data is ready.

© 2010 Husam Alzaq
Computers as Components
Timing diagrams

[Diagram showing timing diagrams for signals A, B, and C, with labels for high, low, 10 ns, changing, stable, timing constraint, rising, and falling.]
Bus read

[Diagram showing waveforms for Clock, R/W, Address, Address enable, Data ready, and Data over time, indicating read and write operations.]
State diagrams for bus read

CPU

Get data

Done

Send data

Release ack

See ack

Adrs

Ack

Adrs

Wait

start

device
Bus wait state

- Clock
- R/W
- Address enable
- Address
- Data ready
- Data

Time

Wait state
Bus burst read

- Clock
- R/W
- Burst
- Address enable
- Address
- Data ready
- Data: Data 1, Data 2, Data 3, Data 4

Time
Bus multiplexing
DMA

- Direct memory access (DMA) performs data transfers without executing instructions.
  - CPU sets up transfer.
  - DMA engine fetches, writes.
- DMA controller is a separate unit.
Bus mastership

- By default, CPU is bus master and initiates transfers.
- DMA must become bus master to perform its work.
  - CPU can’t use bus while DMA operates.
- Bus mastership protocol:
  - Bus request.
  - Bus grant.
DMA operation

- CPU sets DMA registers for start address, length.
- DMA status register controls the unit.
- Once DMA is bus master, it transfers automatically.
  - May run continuously until complete.
  - May use every n\textsuperscript{th} bus cycle.

© 2010 Husam Alzaq

Computers as Components
Bus transfer sequence diagram
System bus configurations

- Multiple busses allow parallelism:
  - Slow devices on one bus.
  - Fast devices on separate bus.
- A bridge connects two busses.
Bridge state diagram
ARM AMBA bus

- Two varieties:
  - AHB is high-performance.
  - APB is lower-speed, lower cost.
- AHB supports pipelining, burst transfers, split transactions, multiple bus masters.
- All devices are slaves on APB.
Memory components

- Several different types of memory:
  - DRAM.
  - SRAM.
  - Flash.
- Each type of memory comes in varying:
  - Capacities.
  - Widths.
Random-access memory

- Dynamic RAM is dense, requires refresh.
  - Synchronous DRAM is dominant type.
  - SDRAM uses clock to improve performance, pipeline memory accesses.
- Static RAM is faster, less dense, consumes more power.
SDRAM operation

- CLK
- CS'
- RAS'
- CAS'
- WE'
- ADRS

adrs
Read-only memory

- ROM may be programmed at factory.
- Flash is dominant form of field-programmable ROM.
  - Electrically erasable, must be block erased.
  - Random access, but write/erase is much slower than read.
Flash memory

- Non-volatile memory.
  - Flash can be programmed in-circuit.

- Random access for read.

- To write:
  - Erase a block to 1.
  - Write bits to 0.
Flash writing

- Write is much slower than read.
  - 1.6 $\mu$s write, 70 ns read.
- Blocks are large (approx. 1 Mb).
- Writing causes wear that eventually destroys the device.
  - Modern lifetime approx. 1 million writes.
Types of flash

**NOR:**
- Word-accessible read.
- Erase by blocks.

**NAND:**
- Read by pages (512-4K bytes).
- Erase by blocks.

**NAND is cheaper, has faster erase, sequential access times.**
Timers and counters

*Very similar:*

- a timer is incremented by a periodic signal;
- a counter is incremented by an asynchronous, occasional signal.

*Rollover causes interrupt.*
Watchdog timer

- Watchdog timer is periodically reset by the system timer.
- If watchdog is not reset, it generates an interrupt to reset the host.
Switch debouncing

- A switch must be debounced to multiple contacts caused by eliminate mechanical bouncing:
Encoded keyboard
Encoded keyboard

- An array of switches is read by an encoder.
- **N-key rollover** remembers multiple key depressions.
LED

Must use resistor to limit current:
7-segment LCD display

- May use parallel or multiplexed input.
Types of high-resolution display

- Liquid crystal display (LCD) is dominant form.
- Plasma, OLED, etc.
- Frame buffer holds current display contents.
  - Written by processor.
  - Read by video.
Touchscreen

- Includes input and output device.
- Input device is a two-dimensional voltmeter:
Touchscreen position sensing

ADC

voltage
Digital-to-analog conversion

Use resistor tree:

\[ V_{\text{out}} \]

\[ b_n \]

\[ 2R \]

\[ b_{n-1} \]

\[ 4R \]

\[ b_{n-2} \]

\[ 8R \]

\[ b_{n-3} \]
Flash A/D conversion

- N-bit result requires $2^n$ comparators:

\[ V_{in} \rightarrow \text{encoders} \rightarrow \text{encoder} \]
Dual-slope conversion

- Use counter to time required to charge/discharge capacitor.
- Charging, then discharging eliminates non-linearities.
Sample-and-hold

• Samples data:

\[ V_{\text{in}} \]

converter
Bus-Based Computer Systems

- Designing with microprocessors.
- Development and debugging.
- System-level performance analysis.
System architectures

- Architectures and components:
  - software;
  - hardware.

- Some software is very hardware-dependent.
Hardware platform architecture

Contains several elements:

- CPU;
- bus;
- memory;
- I/O devices: networking, sensors, actuators, etc.

How big/fast much each one be?
Software architecture

Functional description must be broken into pieces:
- division among people;
- conceptual organization;
- performance;
- testability;
- maintenance.
Hardware and software architectures

Hardware and software are intimately related:

- software doesn’t run without hardware;
- how much hardware you need is determined by the software requirements:
  - speed;
  - memory.