Microcontroller based System Design Concepts

EE4380 Fall 2001

Pari vallal Kannan
Center for Integrated Circuits and Systems
University of Texas at Dallas

9/11 Bad day

-

Step – 1 : Problem Definition

- Define the system to be designed
  - External Inputs (operating conditions, constraints)
  - External Outputs required
  - Engineering time
  - Cost, size, running costs etc
  - Performance/Quality requirements

Step – 2 : Block Diagram

- Identify blocks in the system
  - Prior experience
  - Use the system spec
  - Rigorous processes are available (complicated)
- Identify major signals between blocks
  - Signals are data and control
  - Include external inputs and outputs
  - Determine the range of each signals (bit width)

Step – 3 : Realize the blocks

- Analyze the blocks for their "*realizability*"
- Blocks become
  - Well known components
    - Displays, A/D D/A converters
    - Microprocessor or a digital control system
    - Digital Circuits like decoders, muxes, latches etc
  - Unknown digital blocks
    - Usually made up of random logic
    - Implement using PLDs / FPGAs or discrete components
  - Analog blocks
- If some blocks are unrealizable, then go back to the block diagram

Step - 4 : Component Choice

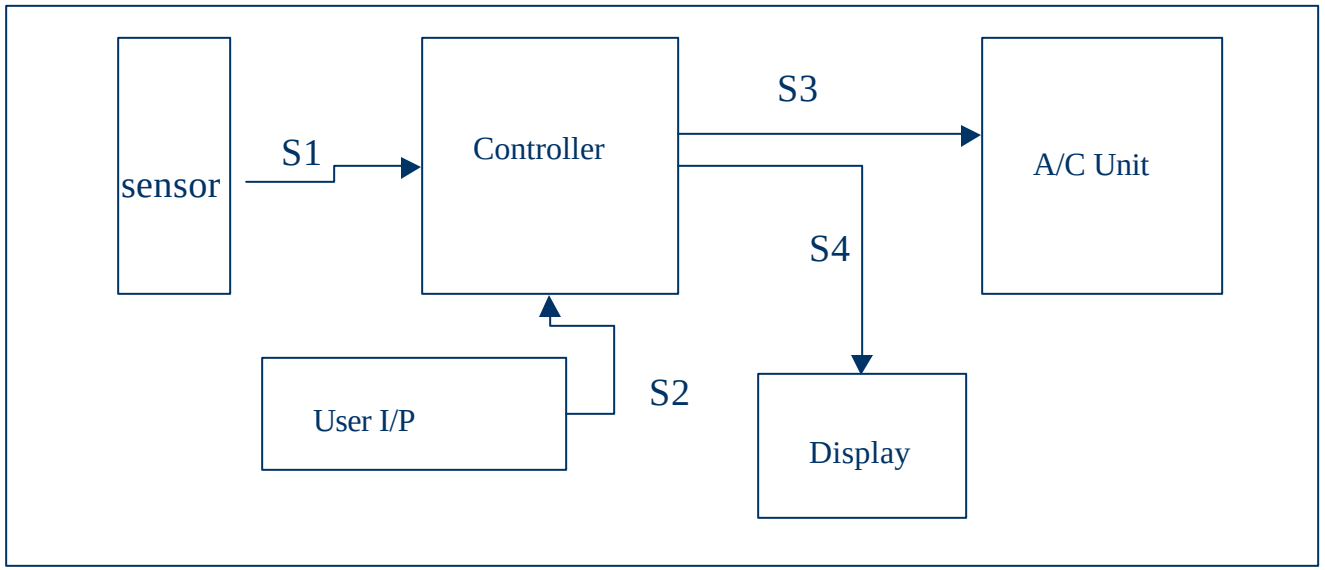
- Microprocessors
  - MIPS rating, power consumption, package size etc
    - MP3 decoding Vs temperature controller
    - Battery operated Vs mains operated
    - Hand held Vs desktop
  - Memory, Peripherals, UI
  - Software (prom code, OS, protocol compliance etc.)
- Sensors
  - Range, accuracy, speed, resolution
- Displays
  - Power, # of characters / lines, UI
- PLDs/FPGAs
  - Number of logic gates, cost, package etc

Example - 1

- Design a temperature controller. Use any temperature sensor of your choice. Control is effected by controlling the turn-on time of the room A/C
- Is spec complete ?
  - Interface to the A/C ? A/C parameters ??
  - Suitable temperature sensor availability ?
  - Cost issues ?
  - UI spec ?

Example – 1 (contd.)

- Block Diagram



Example – 1 (contd.)

- Signal Identification
  - S1 : Sensor and Controller
    - Current Temperature : Sensor → Controller
    - Controller → Sensor ??
  - S2 : User Input and Controller
    - On/Off : User → Controller
    - Temperature Setting : User → Controller
  - S3 : Controller and AC unit
    - On/Off : Controller → AC
    - Status: AC → Controller ??
  - S4: Controller and Display
    - Current Temperature : Controller → Display
    - Status : Controller → Display

Example – 1 (contd.)

- Realizability Analysis
  - Sensor is available as a package
    - Or design one using a thermistor and A/D converter
  - User I/P dip switch
  - Display two or more 7seg LEDs or LCD
  - AC unit control lines
- Controller Design
  - Design a digital controller
  - Use microprocessor

Example – 1 (contd.)

- Microprocessor based Controller
  - Decide on control algorithm
  - Low system speed (a few kHz), so microprocessor speed is not critical
  - Small number of steps in control algorithm. So a low MIPS processor is adequate
  - Choose 8051
  - Need external memory ?
    - Maybe some EPROM.
    - No external RAM
  - Peripherals ?
    - One port for sensor reading, one for user display and one for AC
    - So need an external 8255 I/O chip

Example – 1 (contd.)

- 8051 as the controller
  - Write the code for the control algorithm
  - Choose external ROM from the code size
  - Or choose 8051 clone with internal ROM (8751, DS5000 etc)
- Optionally do a board level simulation
- Design the board with the components
  - Use existing general purpose boards if possible

Board Design

- CAD tools for design entry and pcb design
  - Protel, ORCAD, etc.