EE4800-03
Embedded Systems Design

Lessons 28-30

Design Case Studies

Systems Design Approach

- First reaction with new systems design - PANIC!!!
- No! Apply systematic design approach
  - Requirements
  - 68HC12 systems employed
  - Background theory
  - Hardware construction
  - Structure Chart and UML Activity Diagrams
  - Code
  - Testing

Design Case Study I - Wall-following Robot System

- Autonomous - self-contained
- Navigate through unknown maze
- Detect maze walls with IR emitter-detector pairs
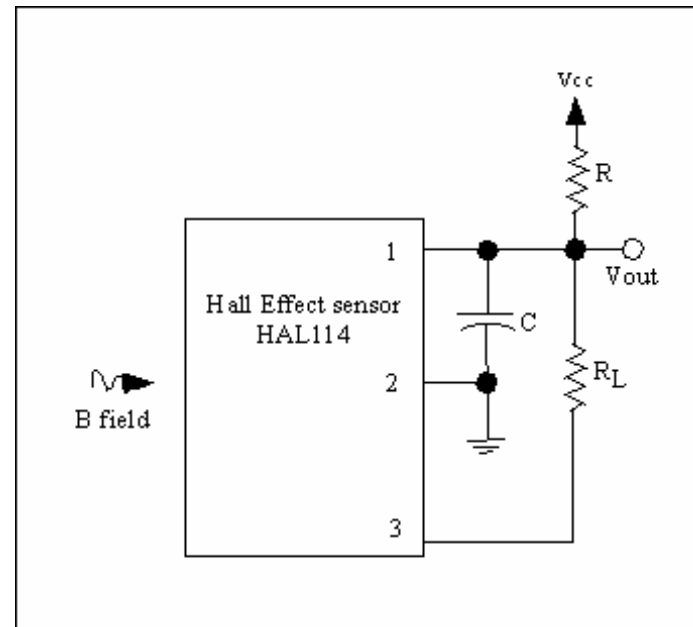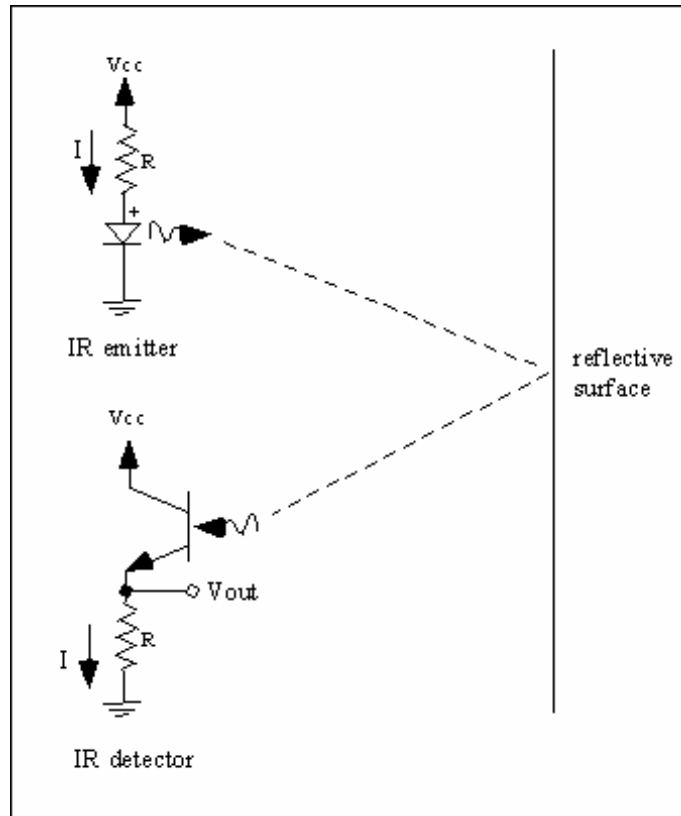- Avoid "land mines" (magnets) in maze floor
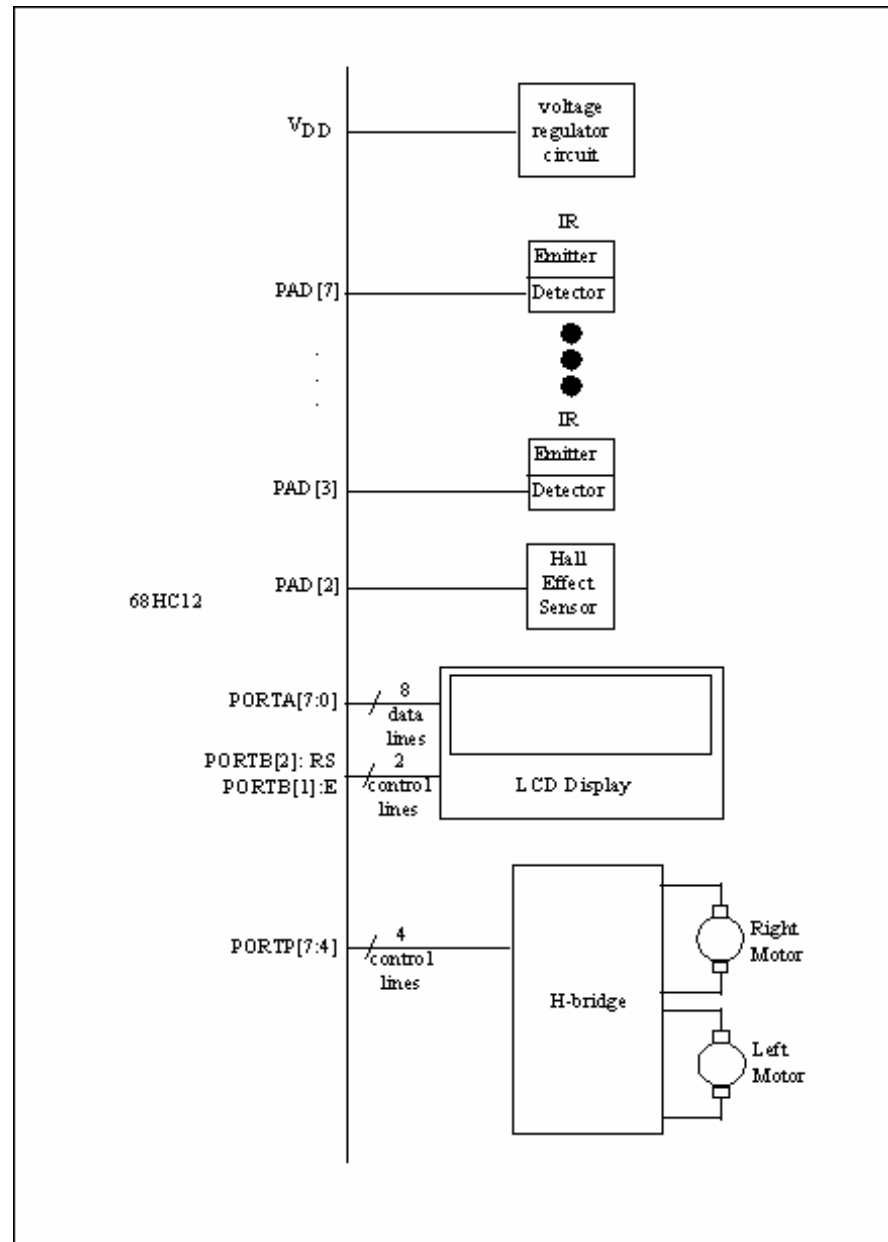
Wall-following Robot System
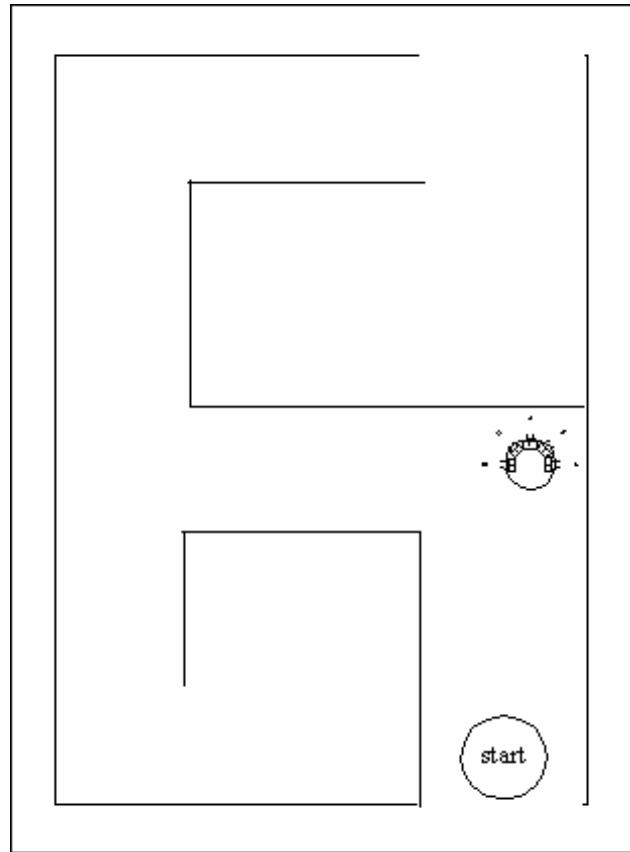
Required Robot Functions

- ATD conversion for IR sensors
- Turn decision algorithm
- Turn control function
- Hall Effect sensor processing
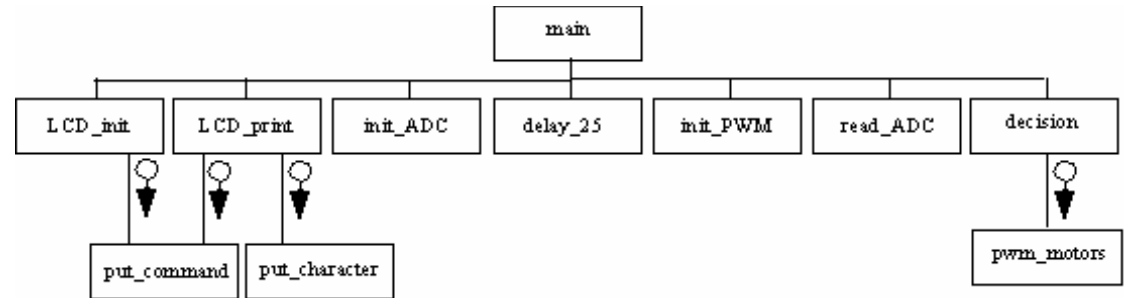- Land mine avoidance algortihm
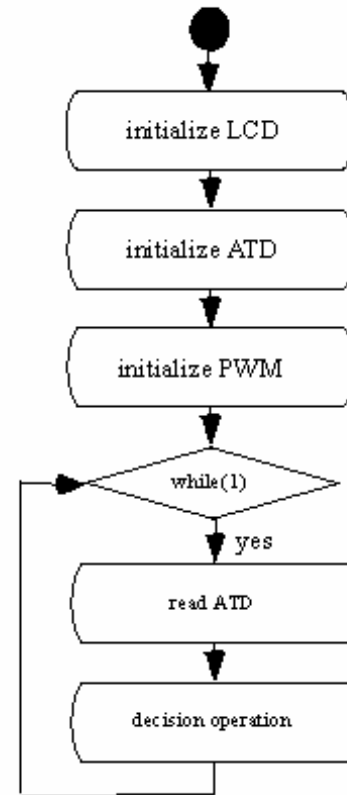- LCD display

Background Theory

Hardware Interface

Software Interface
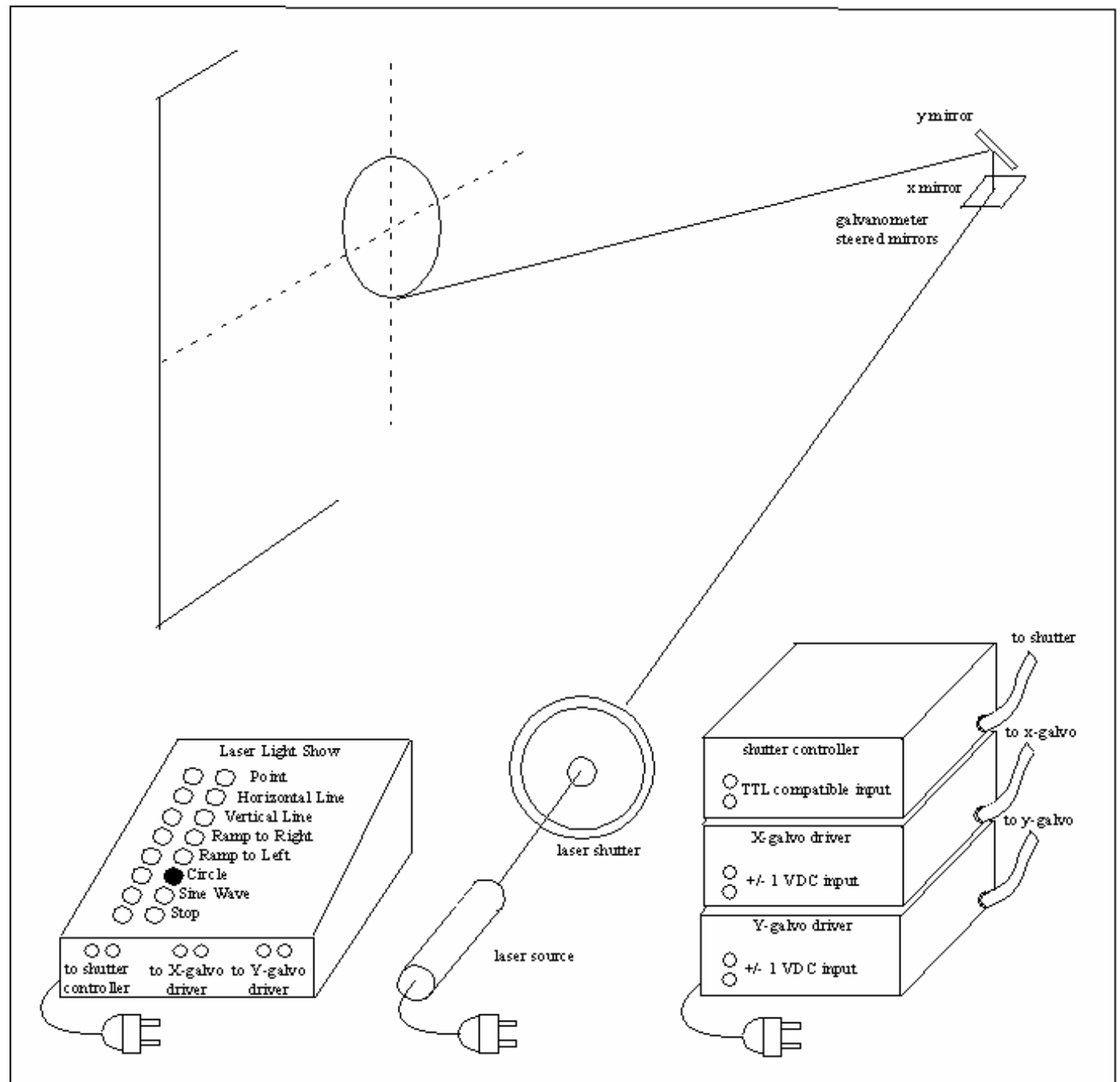

a) Structure Chart

b) UML activity diagram

Design Case Study II - Laser Light Show

- Seven pre-coded patterns

- traced by laser

- LED illuminates on control panel to indicate selected pattern

- Control system traces selected pattern

Laser Light Show

 y mirror

x mirror

galvanometer
steered mirrors

to shutter

to x-galvo

to y-galvo

Laser Light Show

- ○ ○ Point
- ○ ○ Horizontal Line
- ○ ○ Vertical Line
- ○ ○ Ramp to Right
- ○ ○ Ramp to Left
- ○ ● Circle
- ○ ○ Sine Wave
- ○ ○ Stop

to shutter   to X-galvo   to Y-galvo
controller   driver   driver

laser shutter

laser source

shutter controller

○ TTL compatible input
○

X-galvo driver

○ +/- 1 VDC input
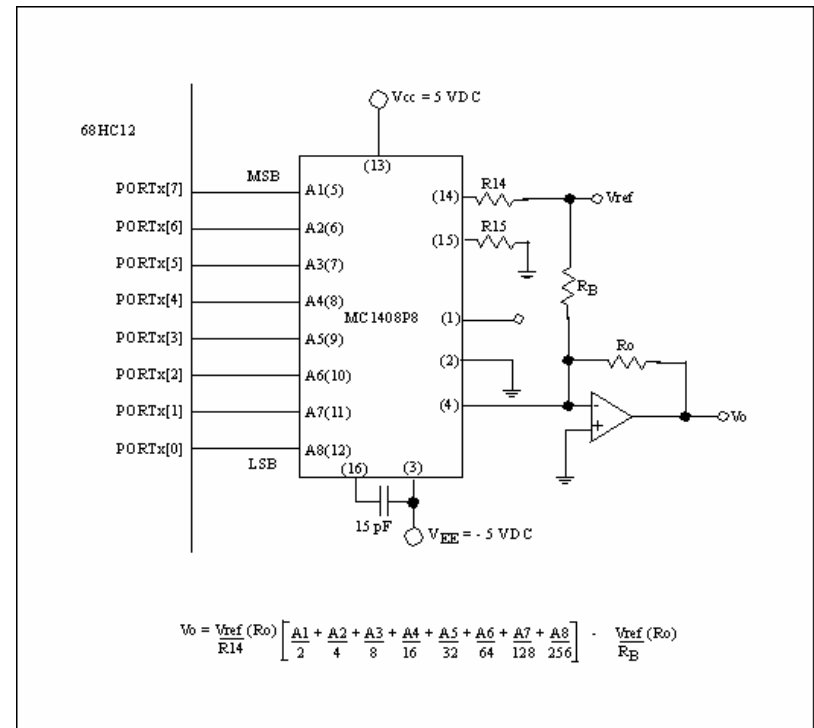○

Y-galvo driver

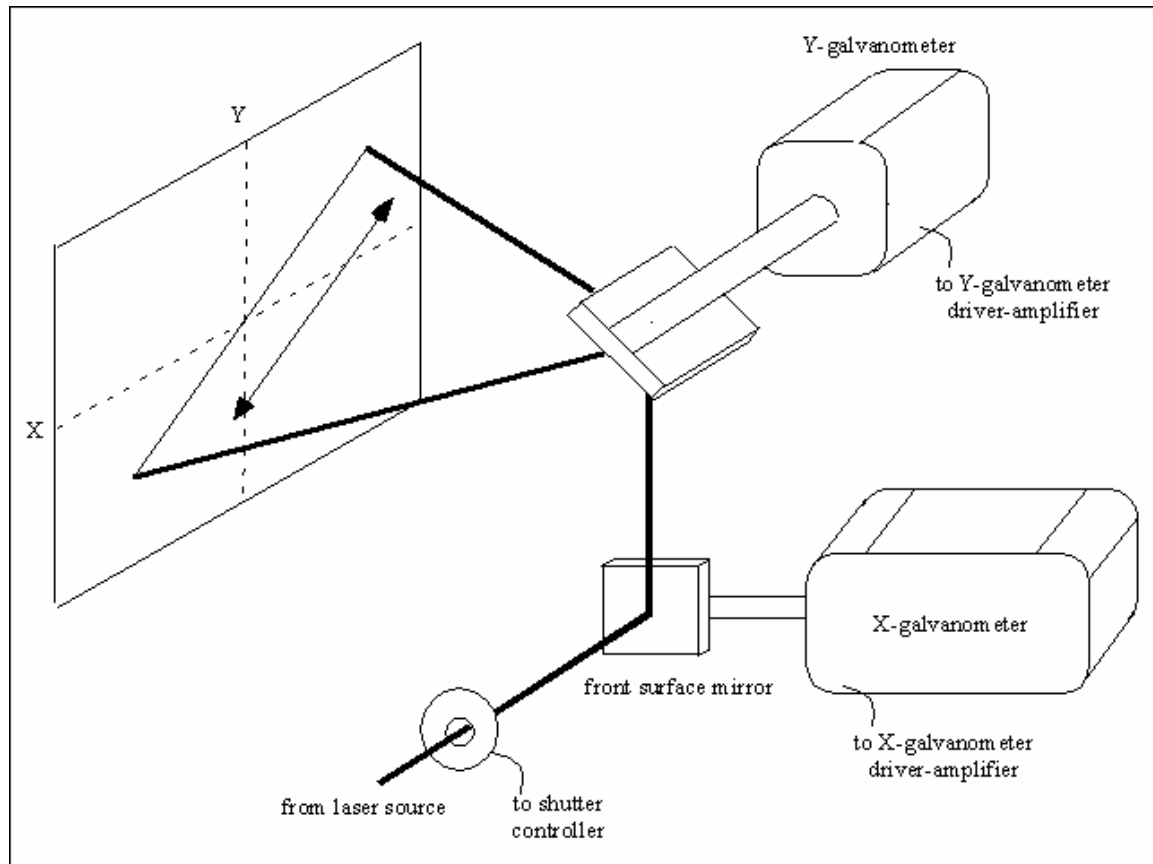○ +/- 1 VDC input
○

68HC12 Systems Employed

- Debounded eight switch bank
- Eight position LED display
- Two-channel DAC
- Laser source
- Shutter and shutter controller
- Two galvanometer steered mirrors
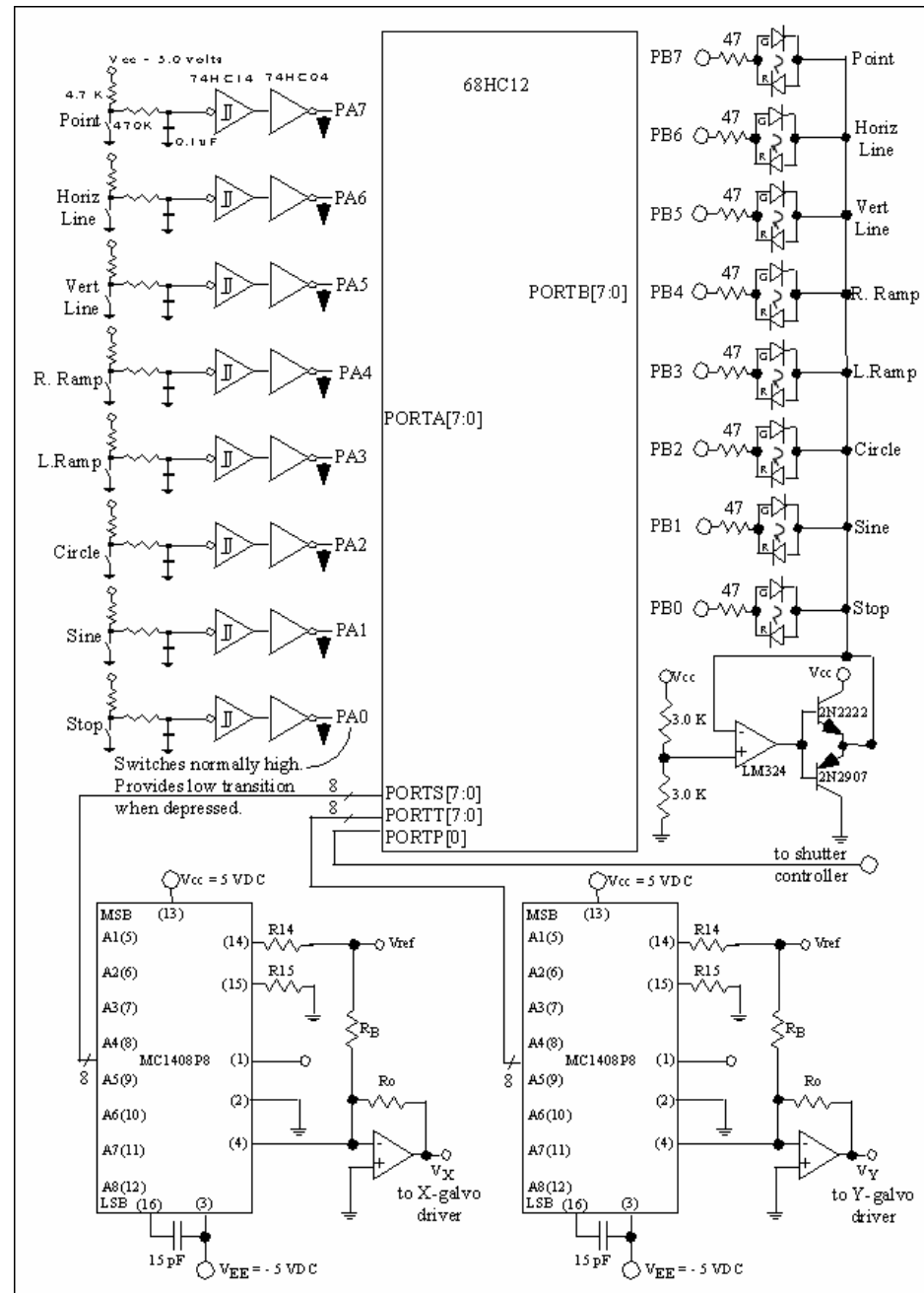
Background Theory

- DAC
- Lasers
- Laser Safety
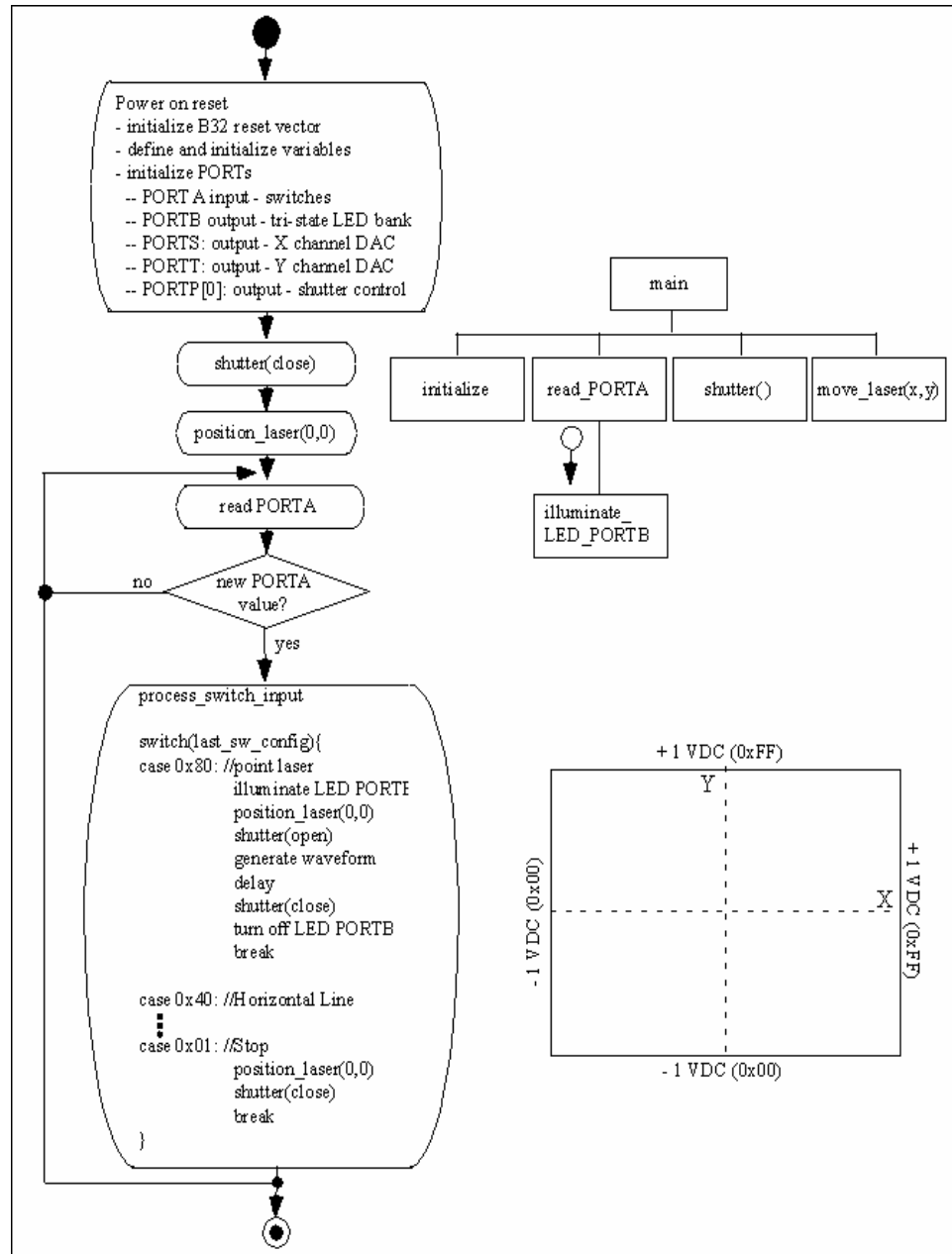- Laser Control HW
  - mirrors
  - shutters
  - galvanometers

X-Y Scanning System

Hardware Interface

Software Design

Testing

- Use bottom-up implementation approach
- Test each subsystem separately and exhaustively
- Combine subsystems one at a time until system fully functional
- Test! Test! Test!