Slide 1

ECE/CE 3720: Embedded System Design

Chris J. Myers

Lecture 24: Microcomputer-Based Control Systems

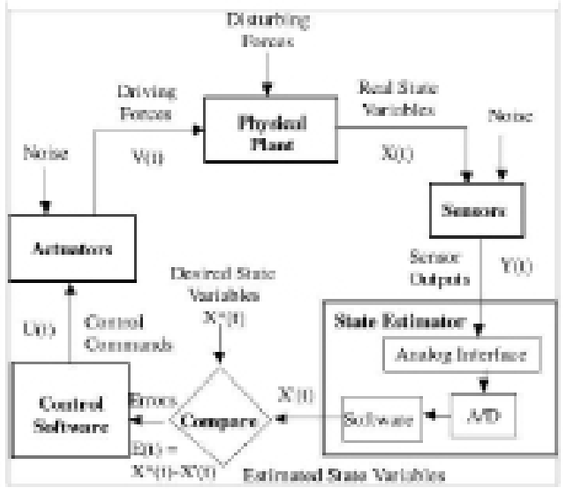
Slide 2

Introduction to Digital Control Systems

- Learned about systems that collect information from the environment.
- Now, we use this information to control the environment.
- A *control system* is a collection of mechanical and electrical devices connected to regulate a *physical plant*
- Many embedded applications are control systems.
- Control theory is a richly developed discipline covered by other courses, so we emphasize implementation issues.

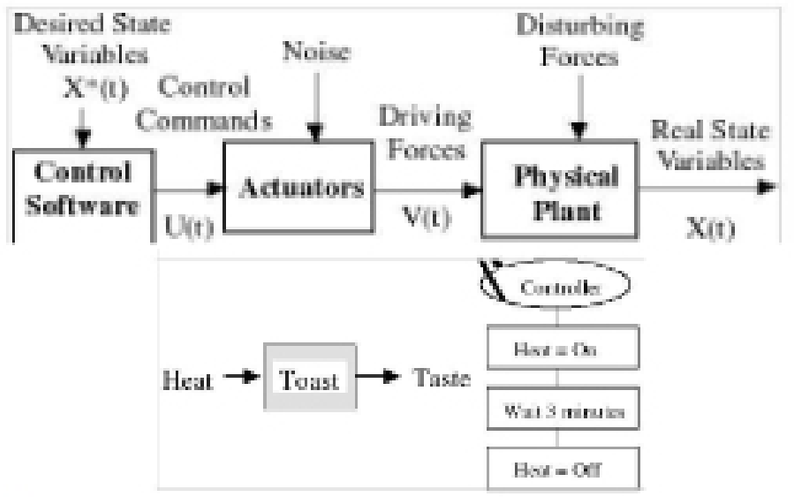
Slide 3

Block Diagram



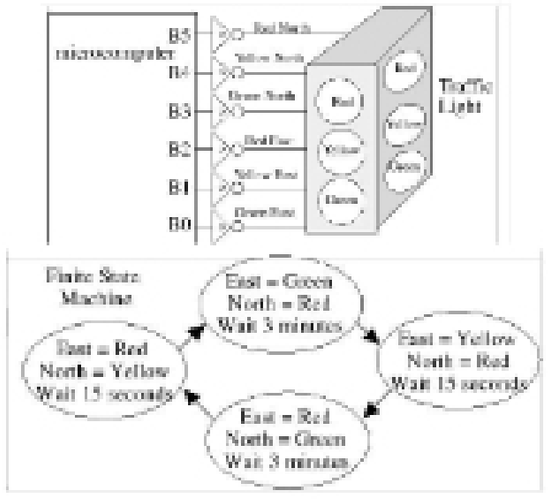
Slide 4

Open-Loop Control Systems



Slide 5

Traffic Light Control System



Slide 6

Traffic Light Controller

```
const struct State {
  unsigned char Out;  /* Output to Port B */
  unsigned short Time;/* Time in sec to wait */
  const struct State *Next;}; /* Next state */
typedef const struct State StateType;
#define NorthRed_EastGreen  &fsm[0]
#define NorthRed_EastYellow &fsm[1]
#define NorthGreen_EastRed  &fsm[2]
#define NorthYellow_EastRed &fsm[3]
StateType fsm[4]={
  {0x21, 180, NorthRed_EastYellow},
  {0x22, 15, NorthGreen_EastRed},
  {0x0C, 180, NorthYellow_EastRed},
  {0x14, 15, NorthRed_EastGreen}};
```

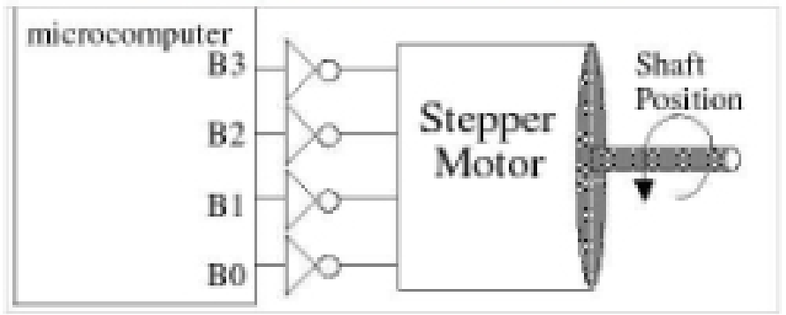

Slide 7

Traffic Light Controller (cont)

```
void main(void){ StatePtr *Pt;  /* Current State */
  Pt=NorthRed_EastGreen; /* Initial State */
  DDRB=0xFF;             /* Make Port B outputs  */
  while(1){
    PORTB=Pt->Out; /* Perform output */
    Wait(Pt->Time);/* Time to wait in this state */
    Pt=Pt->Next;   /* Move to next state */
  }
};
```

Slide 8

Open-Loop Stepper Controller



Slide 9

Circular List for Stepper Motor

```
const struct State {
  unsigned char Out;          /* Output to Port B */
  const struct State *Next;};  /* Next state */
typedef const struct State StateType;
#define S6  &fsm[0]
#define S5  &fsm[1]
#define S9  &fsm[2]
#define S10 &fsm[3]
StateType fsm[4]={{0x06, S5},{0x05, S9},
                  {0x09, S10},{0x0A, S6}};
StatePtr *Pt;  /* Current State */
unsigned short Speed;
```

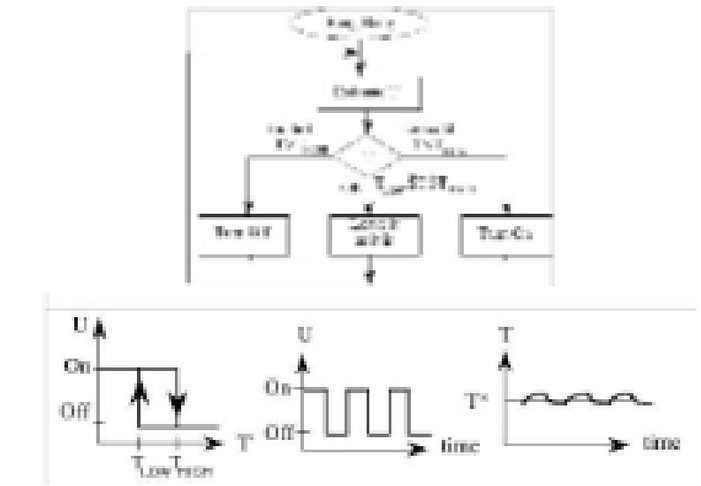
Slide 10

Spin Stepper Motor at Constant Speed

```
#define OC5  0x08
#pragma interrupt_handler TOC5handler()
void TOC5handler(void){
  PORTB=Pt->Out; // output for this state
  Pt=Pt->Next;   // Move to next state
  TFLG1=OC5;     // Ack OC5F
  TOC5=TOC5+Speed; }// Executed every step
void ritual(void) {
asm(" sei");     // make atomic
   TMSK1|=OC5;  // Arm output compare 5
   TFLG1=OC5;   // Initially clear OC5F
   Speed=10000; // initial speed
   Pt=S6;          // initial state
  TOC5=TCNT+2000; // First one in 1 ms
asm(" cli"); }
```


Slide 11

Bang-Bang Temperature Controller



Slide 12

Bang-Bang Temperature Controller