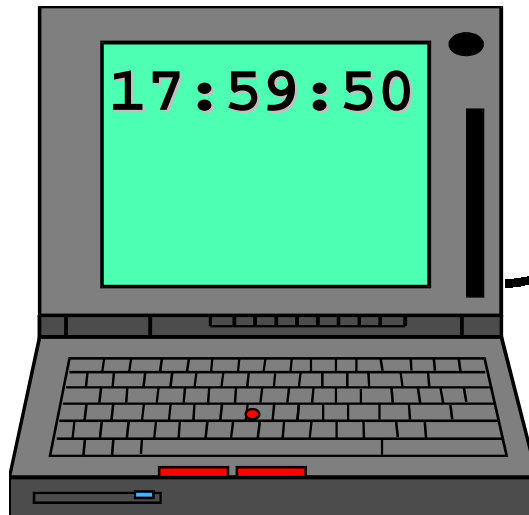


Multitasking Programming

task1=Display clock on screen

task2=onboard LED blinks

**task3=set hour/min while clock
running**

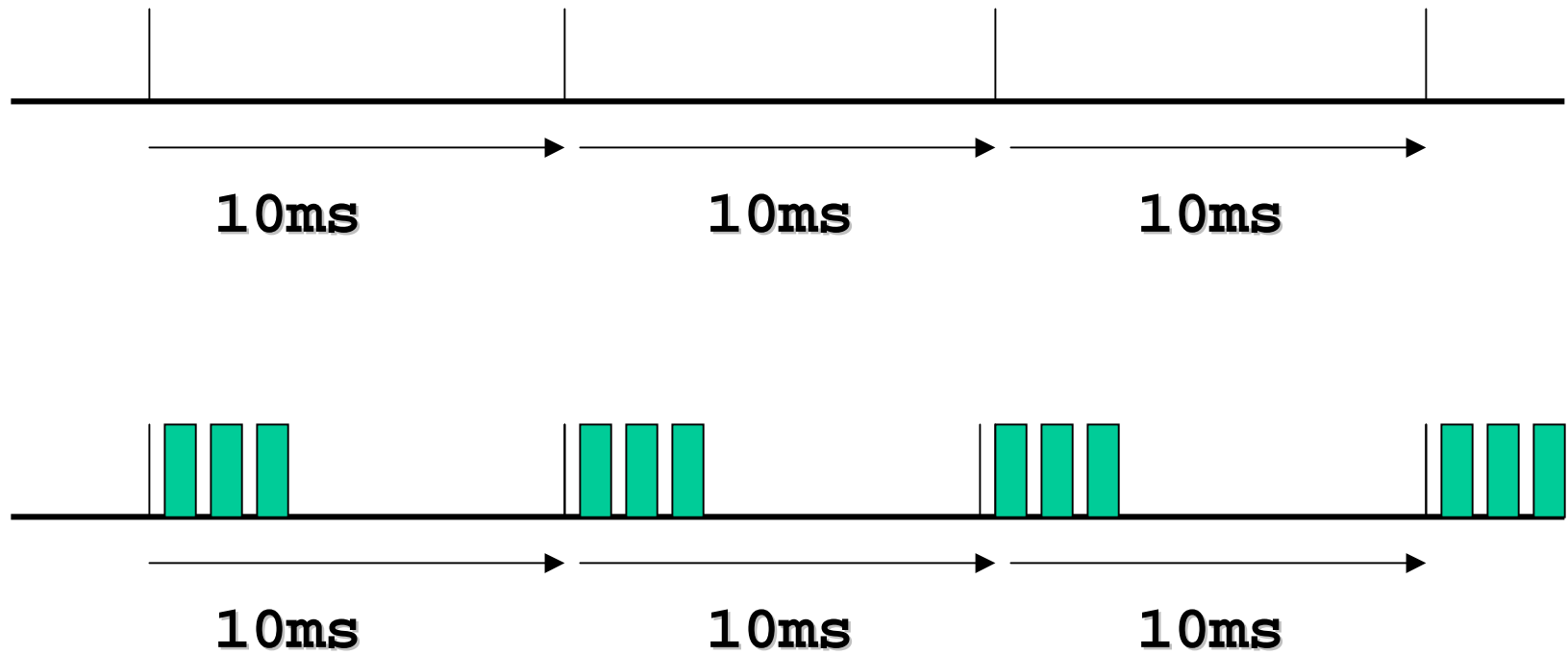


RS232C Cable

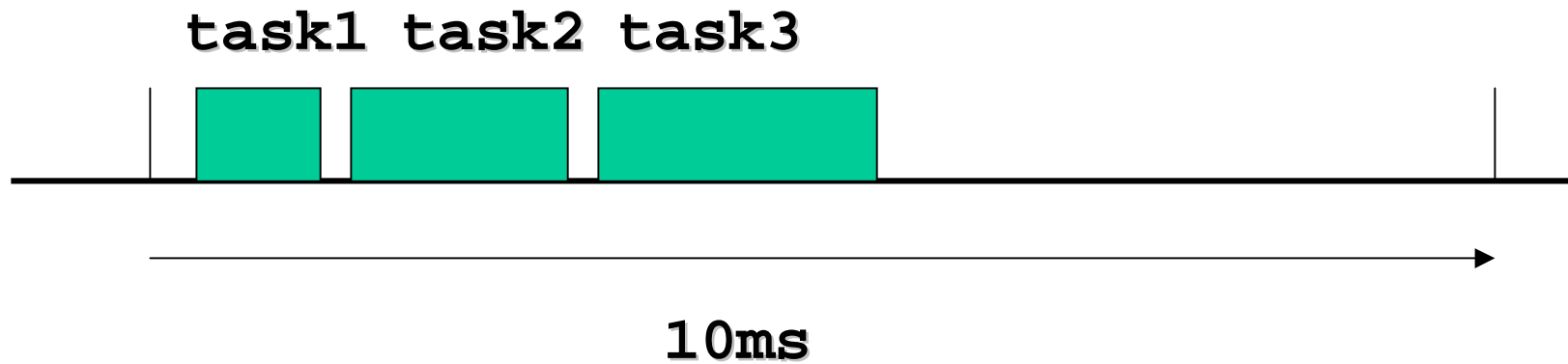


8051SBC

One CPU: Time slice



One CPU: 10ms Time slice



MAIN Program: 10ms loop

```
; main 10ms loop
```

```
LOOP:    CALL WAIT_TICK  
        CALL TICK_LED  
        CALL UPDATE_CLOCK  
        CALL GETCHAR  
        CALL PRINT_TIME  
        CALL HOUR_KEY  
        CALL MIN_KEY  
        JMP LOOP
```

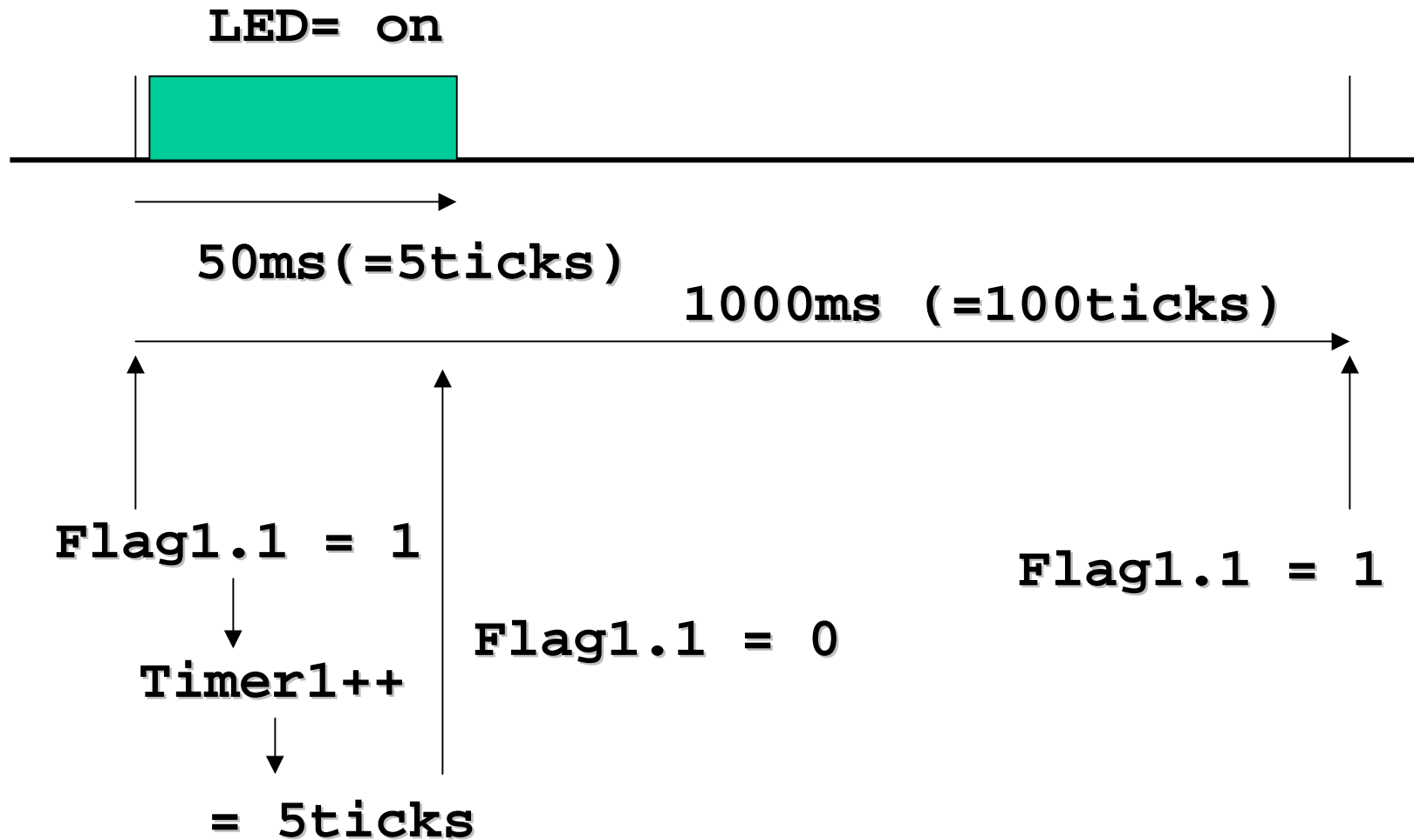
MAIN Program: WAIT_TICK

i _____|_____|_____|_____|_____

WAIT_TICK:

```
        JNB  TF0,$  
        CLR  TF0  
        ORL  TH0,#0DCH  
        INC  TICK  
        RET
```

MAIN Program: TICK_LED



Print time to terminal

; PRINT TIME EVERY SECOND

PRINT_TIME:

JNB FLAG1.0,EXIT_PRINT_TIME

CLR FLAG1.0

PRINT_TIME1:

MOV A,#CR

CALL COUT

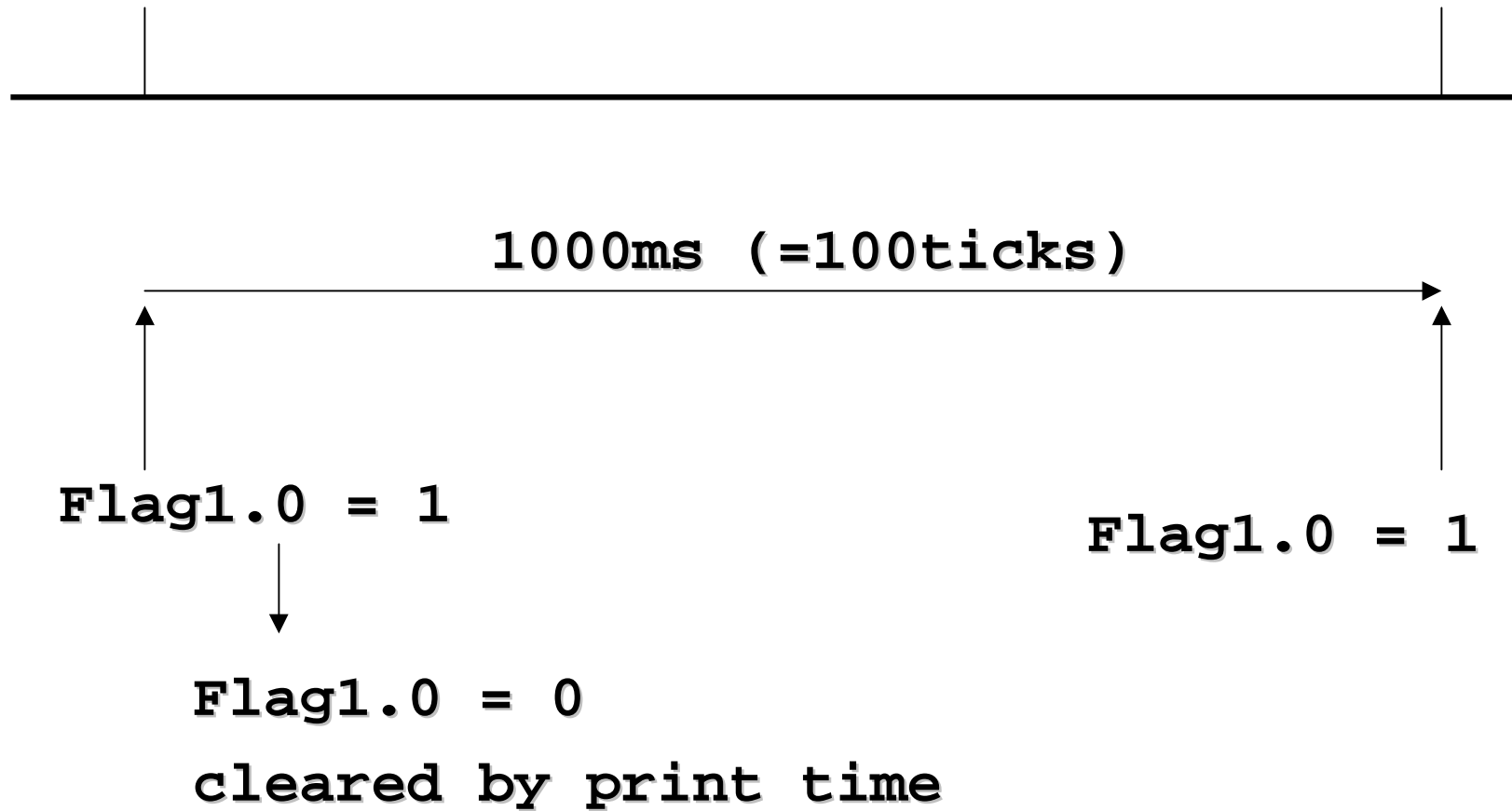
MOV A,HOUR

CALL PHEX

Print time to terminal

```
    MOV A,#':'  
    CALL COUT  
    MOV A,MIN  
    CALL PHEX  
    MOV A,#':'  
    CALL COUT  
    MOV A,SEC  
    CALL PHEX  
  
EXIT_PRINT_TIME:  
    RET
```

FLAG1.0 set every second



Check serial port every 10ms

```
; exit: COMMAND == -1 NO CHARACTER  
; COMMAND != -1 ASCII CODE  
GETCHAR: JNB RI, EXIT2  
CLR RI  
MOV A, SBUF  
MOV COMMAND, A  
RET  
  
EXIT2: MOV COMMAND, #-1  
RET
```

Command 'h' to set HOUR

```
HOUR_KEY: MOV A,COMMAND
           CJNE A,#'h',EXIT_HOUR_KEY
           MOV A,HOUR
           ADD A,#1
           DA  A
           MOV HOUR,A
           CJNE A,#24H,SKIP_CLEAR_HOUR
```

Command 'h' to set HOUR

```
MOV HOUR,#0
```

```
SKIP_CLEAR_HOUR: CALL PRINT_TIME1
```

```
EXIT_HOUR_KEY: RET
```

Command 'm' to set MIN

```
MIN_KEY:MOV  A,COMMAND
          CJNE A,#'m',EXIT_MIN_KEY
          MOV  A,MIN
          ADD  A,#1
          DA   A
          MOV  MIN,A
          CJNE A,#60H,SKIP_CLEAR_MIN
          MOV  MIN,#0
```

Command 'm' to set MIN

SKIP_CLEAR_MIN:

CALL PRINT_TIME1

EXIT_MIN_KEY:

RET

Summary of Multitasking concept

1. No infinite loop for each task!
 2. Task communication was made with FLAG signaling!
 3. Total tasks execution must be less than 10ms or one tick.
 4. Round-robin scheduler.
- Demonstrates code running...