Contents

1 Assignment 3 & 4 Handin Details 1
2 Robot Behaviour 1
3 C Problem Set Solutions 2

1 Assignment 3 & 4 Handin Details

Your assignment handin should follow the given guidelines below:

• be in a non-sealed envelope with a cover page stating the assignment number, and associated student numbers/names.

• include 2 sets (1 set for single person teams) of independent external documentation (max 4 pages) detailing how to assemble, upload, and use your program, and what your program does.

• a printout of your demonstrated program’s LST file.

• a printout of your demonstrated program’s A51 file (if LST file wraps/is not clearly readable).

2 Robot Behaviour

Programming a robot for autonomous (i.e. not remote controlled) operation is not a simple task, but behavioural programming makes the task much simpler. The idea is to think of the robot as an insect controlled by a set of basic behaviours with associated priorities. These priorities are simply the order of tests in your main program’s loop.

The set of events and behaviours you would expect your robot to have in the last assignment include

1. whisker contact - deal with walls/obstructions

2. light sensor readings - find direction of maximum light increase

3. aimed at light - ramming speed!
4. light switched off - turn around and repeat

Finding the light sources can be tricky, the robot has 4 light sensors pointing forward, back, right and left. Although one might be tempted to use the forward light sensor only, it is actually far better to use the side sensors! (think of the problem as finding the light gradient (i.e. direction of maximal increase).

3 C Problem Set Solutions

1. /*
   * Assume that a switch is connected to pin P1.0 and a LED is connected
   * to P1.1.
   * Write a C program for the 8051 which uses T0 interrupts to scan the switch
   * every 15 ms..
   * If the switch is 1 it uses a timer interrupt to blink the LED once per second
   */
#include <stdio.h>
#include <reg83C51.h>

sbit P1_0 = 0x90;
sbit P1_1 = 0x91;
data unsigned long TIME = 0;
data unsigned char SCAN_COUNT = 0;
data unsigned char BLINK_COUNT = 0;
bit BLINK=0;

main ( ) {
    P1_0=1; /*initialize P1.0 for input */
P1_1=0; /*Turn off LED */
TMOD=0x01; /*16 bit timer mode*/
TF0=1; /* trigger first interrupt */
IE=0x84;
while (1) {
}
}

void T0_INT ( void ) interrupt 1 using 1 {
    TR0 = 0;
    TH0 = -(5000/256);
    TL0 = -(5000%256);
    TR0 = 1;
    if (((++BLINK_COUNT==200) && BLINK) ^ P1_1);
    if ( ++SCAN_COUNT == 3) {
        SCAN_COUNT = 0;
        if ( P1_0==1 ) {
            BLINK=1;
void main ( void ) {
    /* XDATA buffer for 100 characters */
    xdata unsigned char buffer[100];
    /* buffer counter */
    data unsigned char number;
    /* setup serial port to use timer 1 auto-reload for baudrate */
    TMOD=0x20;
    /* load overflow period of 253us (arbitrary in this case since not specified)
    TH1=0xfd;
    /* timer 1 start (set TR1) */
    TCON=0x40;
    /* serial port mode 1 (8-bit UART, timer), receive enable */
    SCON=0x50;

    for (number=0; number<100; number++) {
        /* see STDIO.h for getkey, putchar */
        buffer[number] = getkey();
        putchar(buffer[number]);
    }
}