# Computer Parallel Interface Program 3 – Course Tracing Program

This program is used with the PC computer parallel interface.

This program allows you to enter a letter corresponding to the command to be given to the robot. (EXAMPLE: The letter F = FORWARD). The computer will then trace the intended route for the robot on the computer's CRT screen, by placing the instruction letter at each point as the command is entered.

To keep track of the robot's course, the program stores the robot's heading in the variable DIR (Direction). There are 8 possible headings, 1 through 8. As the robot makes a turn, DIR is either incremented (Right Turn) or decremented (Left Turn), thereby allowing the computer to keep track of the robot's intended navigation (lines 260-280). The computer then executes the proper cursor control statements based upon the heading stored in DIR. Lines 300 through 310 correct DIR if the calculated value is less than 1 or greater than 8.

```
10     REM ****ROBOT COURSE TRACING PROGRAM****
20     REM
90     CLS
100     PRINT "F = FORWARD"
110     PRINT "R = TURN RIGHT"
120     PRINT "L = TURN LEFT"
130     PRINT "B = SOUND BEEPER"
140     PRINT "X = LIGHT LED"
150     FOR S = 1 TO 16
160     PRINT
170     NEXT S
180     PRINT TAB(40); "X";
190     '
200     DIR = 1
210     '
220     '
230     A$ = INKEY$
240     IF A$ = "" THEN 230
250     '
260     IF A$ = "F" THEN DIR = DIR
270     IF A$ = "R" THEN DIR = DIR + 1
280     IF A$ = "L" THEN DIR = DIR - 1
290     '
300     IF DIR > 8 THEN DIR = DIR - 8
310     IF DIR < 1 THEN DIR = 8 - DIR
320     '
```

```
400    IF DIR = 1 THEN P$ = CHR$(29) + CHR$(30): REM CURSOR UP
410    IF DIR = 2 THEN P$ = CHR$(30) + CHR$(28): REM CURSOR UP,
       RIGHT
420    IF DIR = 3 THEN P$ = CHR$(28): REM CURSOR RIGHT
430    IF DIR = 4 THEN P$ = CHR$(31) + CHR$(28): REM CURSOR DOWN,
       RIGHT
440    IF DIR = 5 THEN P$ = CHR$(29) + CHR$(31): REM CURSOR DOWN

450    IF DIR = 6 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29) + CHR$(31):
       REM CURSOR DOWN, LEFT
460    IF DIR = 7 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29): REM
       CURSOR LEFT
470    IF DIR = 8 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29) + CHR$(30):
       REM CURSOR UP, LEFT
480    IF A$ = "B" THEN P$ = CHR$(29)
490    IF A$ = "X" THEN P$ = CHR$(29)
500    '
510    PRINT P$; A$;
520    '
530    '
550    IF A$ = "F" THEN Z = 12: REM BINARY 1100 (BOTH MOTORS)
560    IF A$ = "R" THEN Z = 4:  REM BINARY 1000 (LEFT MOTOR)
570    IF A$ = "L" THEN Z = 8:  REM BINARY 0100 (RIGHT MOTOR)
580    IF A$ = "B" THEN Z = 2:  REM BINARY 0010 (SOUND BEEPER)
590    IF A$ = "X" THEN Z = 1:  REM BINARY 0001 (LIGHT LED)
600    '
610    W = Z + 64
620    LPRINT CHR$(W); : REM SEND TO ROBOT
630    '
640    GOTO 230
```

## Computer Serial Interface Program – Course Tracing Program

This program is used with the PC computer serial interface.

This program allows you to enter a letter corresponding to the command to be given to the robot. (EXAMPLE: The letter F = FORWARD). The computer will then trace the intended route for the robot on the computer's CRT screen, by placing the instruction letter at each point as the command is entered.

To keep track of the robot's course, the program stores the robot's heading in the variable DIR (Direction). There are 8 possible headings, 1 through 8. As the robot makes a turn, DIR is either incremented (Right Turn) or decremented (Left Turn), thereby allowing the computer to keep track of the robot's intended navigation (lines 260-280). The computer then executes the proper cursor control statements based upon the heading stored in DIR. Lines 300 through 310 correct DIR if the calculated value is less than 1 or greater than 8.

```
10     REM ***ROBOT COURSE TRACING PROGRAM***
20     REM ***FOR IBM COMPATIBLE ONLY***
30     REM
40     OPEN "COM1:300,N,8" FOR RANDOM AS #1
90     CLS
100     PRINT "F = FORWARD"
110     PRINT "R = TURN RIGHT"
120     PRINT "L = TURN LEFT"
130     PRINT "B = SOUND BEEPER"
140     PRINT "X = LIGHT LED"
150     FOR S = 1 TO 16
160     PRINT
170     NEXT S
180     PRINT TAB(40); "X";
190     '
200     DIR = 1
210     '
220     '
230     A$ = INKEY$
240     IF A$ = "" THEN 230
250     '
260     IF A$ = "F" THEN DIR = DIR
270     IF A$ = "R" THEN DIR = DIR + 1
280     IF A$ = "L" THEN DIR = DIR - 1
290     '
300     IF DIR > 8 THEN DIR = DIR - 8
310     IF DIR < 1 THEN DIR = 8 - DIR
320     '
400     IF DIR = 1 THEN P$ = CHR$(29) + CHR$(30): REM CURSOR UP
410     IF DIR = 2 THEN P$ = CHR$(30) + CHR$(28): REM CURSOR UP,
        RIGHT
420     IF DIR = 3 THEN P$ = CHR$(28): REM CURSOR RIGHT
430     IF DIR = 4 THEN P$ = CHR$(31) + CHR$(28): REM CURSOR DOWN,
        RIGHT
440     IF DIR = 5 THEN P$ = CHR$(29) + CHR$(31): REM CURSOR DOWN
450     IF DIR = 6 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29) + CHR$(31):
        REM CURSOR DOWN, LEFT
460     IF DIR = 7 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29): REM
        CURSOR LEFT
470     IF DIR = 8 THEN P$ = CHR$(29) + CHR$(29) + CHR$(29) + CHR$(30):
        REM CURSOR UP, LEFT
480     IF A$ = "B" THEN P$ = CHR$(29)
490     IF A$ = "X" THEN P$ = CHR$(29)
500     '
```

```
510    PRINT P$; A$;
520    '
530    '
550    IF A$ = "F" THEN Z = 12: REM BINARY 1100 (BOTH MOTORS)
560    IF A$ = "R" THEN Z = 4: REM BINARY 1000 (LEFT MOTOR)
570    IF A$ = "L" THEN Z = 8: REM BINARY 0100 (RIGHT MOTOR)
580    IF A$ = "B" THEN Z = 2: REM BINARY 0010 (SOUND BEEPER)
590    IF A$ = "X" THEN Z = 1: REM BINARY 0001 (LIGHT LED)
600    '
610    Z = Z + 64
620    PRINT #1, CHR$(Z); : REM SEND TO ROBOT
630    '
640    GOTO 230
```

### Serial Interface BAUD Rate:

The BAUD rate is the speed at which the interface communicates with the computer. The interface has been designed so that you can select one of the five most popular BAUD rates.

To set the BAUD rate the appropriate switch must be set to the ON position. The important thing is that the BAUD rate of the interface must match that of the computer.

| BAUD RATE | Switch Section ON |
|---|---|
| 300 | 1 |
| 600 | 2 |
| 1200 | 3 |
| 2400 | 4 |
| 4800 | 5 |

### Parity:

Along with the data bits, an extra PARITY BIT is often sent to provide for ERROR checking.

| | |
|---|---|
| Parity | Switch 6 ON |
| No Parity | Switch 6 OFF |

### Number of STOP bits:

After sending the data bits, a serial interface sends out either one or two STOP bits, which informs the receiving equipment that a complete set of bits have been sent.

| | |
|---|---|
| One Stop Bit | Switch 7 ON |
| Two Stop Bits | Switch 7 OFF |

### Number of Bits:

Some computers send out 7 bits at a time, others send out 8.  The computer and interface must match.

| | |
|---|---|
| 7 Bits | Switch 8 ON |
| 8 Bits | Switch 8 OFF |