

ServoCenter3.1 Direct Serial I/O Examples – gcc/Linux

1. Overview

This document provides a simple C-language example program illustrating how to access the serial port and directly communicate with the ServoCenter 3.1 controller board to control a servo in the Linux environment. Note that the program is provided to illustrate simple raw serial communication using the ServoCenter 3.1 protocol and does not demonstrate the full feature set of the ServoCenter 3.1 controller. Refer to section 4.1 of the ServoCenter 3.1 manual for a description of the entire ServoCenter 3.1 protocol and feature set.

2. Sample Code

```

/*****\
* This demo program illustrates how to move servo motors *
* using raw serial communication access to the *
* Yost Engineering, Inc. ServoCenter 3.1 controller board. *
* in Linux. This code was compiled using gcc. *
* (c) 2001-2004 Yost Engineering, Inc. *
* www.YostEngineering.com *
*
\*****/

#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <fcntl.h>
#include <errno.h>
#include <termios.h>

int open_port(int portnum)
{
    int fd;
    char portfile[100]={'\0'};
    if(portnum==1)
        sprintf(portfile, "/dev/ttyS0");
    else if(portnum==2)
        sprintf(portfile, "/dev/ttyS1");
    else if(portnum==3)
        sprintf(portfile, "/dev/ttyS2");
    else if(portnum==4)
        sprintf(portfile, "/dev/ttyS3");
    else
    {
        printf("open_port: unrecognized port number\n");
        return (-1);
    }
    if((fd=open(portfile, O_RDWR | O_NOCTTY | O_NDELAY))==-1)
        perror("open_port: unable to open /dev/ttyS0 - ");
}

```

```

return (fd);
}

void init_port(int *fd, unsigned int baud)
{
    struct termios options;
    //note: the termios structure does not support a baud rate of 14400
    tcgetattr(*fd,&options);
    switch(baud)
    {
        case 9600:      cfsetispeed(&options,B9600);
                       cfsetospeed(&options,B9600);
                       break;
        case 19200:    cfsetispeed(&options,B19200);
                       cfsetospeed(&options,B19200);
                       break;
        case 38400:    cfsetispeed(&options,B38400);
                       cfsetospeed(&options,B38400);
                       break;
        default:       cfsetispeed(&options,B9600);
                       cfsetospeed(&options,B9600);
                       break;
    }
    options.c_cflag |= (CLOCAL | CREAD);
    options.c_cflag &= ~PARENB;
    options.c_cflag &= ~CSTOPB;
    options.c_cflag &= ~CSIZE;
    options.c_cflag |= CS8;
    tcsetattr(*fd,TCSANOW,&options);
}

void moveservo(int *fd, int boardnum, int servonum, int position, int speed)
{
    char buffer[6];
    int num;
    buffer[0]=boardnum%16 + 0xf0;
    buffer[1]=0x10;
    buffer[2]=servonum%16;
    buffer[3]=position%201;
    buffer[4]=speed%101;           //create packet
    buffer[5]='\0';
    num=write(*fd,buffer,5);      //send packet
}

int main()
{
    int fd,board,servo,position,speed,portnum;
    printf("                               \n");
    printf("          ServoCenter 3.1 Demonstration Program      \n");
    printf("          (c)2000-2004 Yost Engineering, Inc.         \n");
    printf("          www.YostEngineering.com                     \n");
    printf("                               \n");
    printf("Enter Port Number (1-4)\n");
    scanf("%d",&portnum);
    if((fd=open_port(portnum))===-1) //open serial port

```

```
    return (1);
init_port(&fd,9600);          //set serial port to 9600,8,n,1
while(1)
{
    printf("Enter Board Number (0-15)\n");
    scanf("%d",&board);
    printf("Enter Servo Number (0-15)\n");
    scanf("%d",&servo);
    printf("Enter Position (0-200)\n");
    scanf("%d",&position);
    printf("Enter Speed (1-100)\n");
    scanf("%d",&speed);
    printf("Sending Command...");
    moveservo(&fd,board,servo,position,speed);
    printf("done!\n");
}
return (0);
}
```

3. Additional Information

This code is available in the ServoCenter_gcc.c file in the Examples directory of the ServoCenter3.1 CD or online at www.YostEngineering.com/ServoCenter.