

Instantaneous Changing of IPm Serial Ports**Abstract:**

It is possible to write a C program that reconfigures IPm serial ports for protocol (SIXNET, Modbus, User, Unassigned), baud rate, mode, etc... By default, any mode changes will take up to one minute to take effect. This document explains how to make the mode changes nearly instantaneous. It is assumed that the developer already has written C code that successfully reconfigures the serial port(s) but the changes don't take effect right away.

Functional Description

An IPm station has a separate configuration file for each of its serial ports (in the /etc/stacfg sub-directory). There is also one empty file, /tmp/update_ports. Periodically (about once a minute) the IPm firmware checks the date and timestamp of this empty file. If the date/timestamp has changed since the last check, the serial port configuration files are then checked to see if any of them have changed since bootup or since they were last written to by the I/O Tool Kit program. If any configuration file has changed, the corresponding serial port will be updated immediately from its respective configuration file. The I/O Tool Kit writes to the empty file whenever a "load" operation is performed, thereby assuring that all serial port configuration changes take effect immediately. This same technique can be used in a C program.

Sample C Code

```
include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <errno.h>
extern int errno;
int main()
{
int fd;
if ((fd = open("/tmp/update_ports", O_WRONLY | O_TRUNC)) == -1) {
fprintf(stderr, "open() failed: %s\n", strerror(errno));
exit (1);
}
if (close(fd) == -1) {
fprintf(stderr, "close() failed: %s\n", strerror(errno));
exit (1);
}
exit (0);
}
```