/* Example code that runs on an external host and obtains packets from an */
/* APP550. The external host is only used for the slow path. */

#define BUF_SIZE 2048
#include <agere_np5.h>

int main(int argc, char *argv[]) {

    ag_st_t rc;
    ag_np5_dev_hdl_t devHandle;
    unsigned char pdu_buf[BUF_SIZE];
    ag_uint32_t pdu_buf_size = BUF_SIZE, pdu_size, devNum;

    if (argc < 2) {
        fprintf(stderr, "Usage: %s <device number>\n", argv[0]);
        return(-1);
    }

    /* get device number from command line */
    devNum = atoi(argv[1]);

    /* Open NP5 device */
    rc=ag_np5_dev_open(devNum, 0, &devHandle);
    if (rc != AG_ST_SUCCESS) {
        fprintf(stderr, "Error: Cannot open device number %i.\n", devNum);
        return(-1);
    }

    /* read packets sent from the APP550 */

    while(1) { /* do forever */

        /* read packet from ASI receive queue (block if queue is empty) */
        rc = ag_np5_dev_pdu_read(devHandle, pdu_buf, pdu_buf_size, &pdu_size);

        /* use return code to determine processing */
        switch (rc) {

            case AG_ST_DEV_INVALID_HANDLE:
                fprintf(stderr, "Error: Invalid device handle! Exiting.\n");
                return(-1);
            
            case AG_ST_DEV_INVALID_BUFFER:
                fprintf(stderr, "Error: Invalid PDU buffer! Exiting.\n");
                return(-1);
            
            case AG_ST_SUCCESS:
                /* At this point, a packet has been loaded into pdu_buf. Additional */
                /* code should be inserted here to handle the packet. */
                break;
            
            default:
                break;
        }
    }
}
fprintf(stderr, "\nUnknown return code: %u. Exiting.\n", rc);
return(-1);
}