

SerialGen.cpp

```

// SerialGen.cpp - GENERATE SERIAL DATA
//
// MAINTENANCE HISTORY
// DATE          PROGRAMMER AND DETAILS
// 12-09-08   JS      Original
//
//-----

#include <cmath>          // C-style mathematical functions
#include <fstream>        // C++ File stream declarations
using namespace std;     // Expand the standard namespace

//-----

// DEFINITIONS

static const size_t BLOCK_LEN = 1022;
                        // Block length
static const size_t STREAM_LEN = 1022*20 + 326;
                        // Stream length

//-----

// MAIN LINE

int
main ()
{
    size_t      bytesRem;           // Bytes remaining
    char        blockBuf[BLOCK_LEN]; // Block buffer
    size_t      blockLen;          // Block length
    fstream     blockStream;       // Block stream file
    fstream     controlStream;     // Control stream file
    size_t      i;                 // General purpose index

    // Create the output files

    blockStream.open ("SERIAL_BLOCKS", ios_base::out);
    controlStream.open ("SERIAL_CONTROL", ios_base::out);

    // Generate blocks until all blocks have been sent

    bytesRem = STREAM_LEN;
    while (bytesRem != 0) {
        if (bytesRem > BLOCK_LEN)
            blockLen = BLOCK_LEN;
        else
            blockLen = bytesRem;
        for (i = 0; i < blockLen; i++)
            blockBuf[i] = static_cast<char>(rand());
        blockStream << static_cast<char>(blockLen >> 8);
        blockStream << static_cast<char>(blockLen);
        blockStream.write (blockBuf, blockLen);
        controlStream.write (blockBuf, blockLen);
        bytesRem -= blockLen;
    }

    // Close the output files

    blockStream.close ();
    controlStream.close ();

    return 0;
}

```