XsAndOs Network Protocol

If you do not wish to use the C++ player interface provided in the SDK, you will need to implement your own compatible communication protocol. This is how it operates:

Connections are through network sockets, using the TCP/IP protocol. The default port is 8080, but this needs to be configurable. The command line of your program should take the network address (DNS name or IP address, including port number). If nothing is specified, “localhost:8080” should be used.

Operation:

Host listens for connections on the specified port. Connect to it. All messages are null-terminated text strings, capitalized.

I have color-coded the messages to clarify the direction: Player-to-Host or Host-to-Player

The player should immediately send the protocol message to the host:  
 CONTEST API VERSION 1

(Note that this may be updated should any incompatible protocol changes are made during the contest.)

The player should then immediately send a message with the name of the player executable file or team name. Examples:

brian/build/player

or

brian.exe

The host will respond with the contents of the board file that will be used. Each line is prepended with an extra six characters “BOARD:“ For example:

BOARD:XXOO

BOARD:OOXX

BOARD:frames 17

The host will then wait for all players to connect. When the game is ready to commence, the host sends a single message assigning the player as X or O:

PLAYER:X

or

PLAYER:O

Gameplay:

During each frame, the player may send a move, or several moves. Each move contains a frame number so the host knows the context. Frames that do not match the current frame will be ignored by the host. Numbering starts from 0.

MOVE <frameNum> cell\_x cell\_y

For example, on frame 11 the player wishes to flip the cell at location (5, 8):

MOVE 11 5 8

It then continues processing and discovers a better move:

MOVE 11 22 40

After the frame simulation time has expired, the host will send the same notification to all players with the moves that were committed, including the frame number. Notes that there can be 0, 1 or 2 moves depending on what the players did.

MOVES <frameNum> [move1\_x move1\_y] [move2\_x move2\_y]

For example, on frame 11 both players submitted moves:

MOVES 11 38 91 22 40

or only one player submitted a move:

MOVES 11 22 40

or both players chose not to toggle a cell:

MOVES 11

The next frame begins immediately after the host transmits this message.

After the last frame is complete, the host will break the connection.