Appendix E: Error Messages

This following sections list the iWarp-specific FORTRAN compiler error messages.

Error Messages for SEND and RECEIVE

Error on line *nn* of *foo*.f: argument out of range for _receive?

The *Channel* specified was not *GATE0* or *GATE1*. To correct, make sure you include ‘‘f77/gates.h’’ in the program (before the _RECEIVE?) and change the _RECEIVE? to specify either *GATE0* or *GATE1*.

Error on line *nn* of *foo*.f: argument out of range for _send?

The *Channel* specified was not *GATE0* or *GATE1*. To correct, make sure you include ‘‘f77/gates.h’’ in the program (before the _SEND?) and change the _SEND? to specify either *GATE0* or *GATE1* as the first parameter.

Error on line *nn* of *foo*.f: bad argument type for _receive?

The *Channel* is not specified as a constant *GATE0* or *GATE1*; variables cannot be used. To correct, make sure you include ‘‘f77/gates.h’’ in the program (before the _RECEIVE?) and change the _RECEIVE? to specify either *GATE0* or *GATE1*.

Error on line *nn* of *foo*.f: bad argument type for _send

The *Channel* is not specified as a constant *GATE0* or *GATE1*; variables cannot be used. To correct, make sure you include ‘‘f77/gates.h’’ in the program (before the _SEND?) and change the _SEND? to specify either *GATE0* or *GATE1*.

Error on line *nn* of *foo*.f: _send argument has incompatible type

The second argument of the _SEND? does not match the type specified by the type designation ‘*?*’. The compiler coerces the parameter to the type expected. Apply an explicit type cast to expressions to coerce them silently to the type desired or use the correct type of send pseudo-function.

Error on line *nn* of *foo*.f: bad number of arguments for _receive

RECEIVE? takes exactly one argument, none were supplied.

Error on line *nn* of *foo*.f: bad number of arguments for _receive

RECEIVE? takes exactly one argument, too many were specified.

Error messages for COND and CONDCALL

Error on line *nn* of *foo*.f: argument out of range for _cond

The *condition-code* specified in the _COND was not a legal value. Only *condition-code* names defined by include ‘‘f77/iwarp_cc.h’’ or include ‘‘f77/iwarp_math_cc.h’’ should be used.

Error on line *nn* of *foo*.f: argument out of range for _condcall

The *condition-code* specified in the _CONDCALL was not a legal value. Only *condition-code* names defined by include ‘‘f77/iwarp_cc.h’’ or include ‘‘f77/iwarp_math_cc.h’’ should be used.

Error on line *nn* of *foo*.f: bad number of arguments to intrinsic _cond

Condition testing with _COND takes exactly one argument, none were supplied.

Error on line *nn* of *foo*.f: bad number of arguments to intrinsic _condcall

Condition testing with _CONDCALL takes exactly one argument, none were supplied.

Error on line *nn* of *foo*.f: bad number of arguments for _cond

Condition testing with _COND takes exactly one argument, too many were supplied.

Error on line *nn* of *foo*.f: bad number of arguments for _condcall

Condition testing with _CONDCALL takes exactly one argument, too many were supplied.

Error on line *nn* of *foo*.f: bad argument type for _cond

The *condition-code* specified in the _COND was not a valid type. Only *condition-code* names defined by include ‘‘f77/iwarp_cc.h’’ or include ‘‘f77/iwarp_math_cc.h’’ should be used.

Error on line *nn* of *foo*.f: bad argument type for _condcall

The *condition-code* specified in the _CONDCALL was not a valid type. Only *condition-code* names defined by include ‘‘f77/iwarp_cc.h’’ or include ‘‘f77/iwarp_math_cc.h’’ should be used.