-module(monitor\_con).

-export([morn/0,get\_flow\_table/0, send\_flow/0, go/0]).

switch1() -> 'linc@pc'.

switch2() -> 'linc@pc-0'.

morn() ->

Pid1 = spawn(monitor\_con, get\_flow\_table,[]),

Pid2 =spawn(monitor\_con, send\_flow, []),

Pid1 ! {self(), ok},

receive

Msg -> Msg

end,

[ {go() ->

X = rpc:call(Switch2, erlang, is\_alive, [])},

case X of

true ->

%Pid2 = spawn(monitor\_con, send\_flow,[]),

Pid2 ! Msg;

false ->

go()

end.].

get\_flow\_table() ->

Switch = switch1(),

monitor\_node(Switch, true),

receive

{From, ok} -> ok

end,

Msg= rpc:call(Switch, linc\_us4\_flow, get\_flow\_table,[0,0]),

%Pid = spawn(monitor\_con, send\_flow, []),

From ! Msg.

send\_flow() ->

Switch2 = switch2(),

monitor\_node(Switch2, true),

receive

Msg ->

io:format("~w", [Msg]),

Switch2! Msg

end.