Adding Conditionality to Bundled-Data Designs

Dylan Hand[*†], Austin Katzin[*], **William Koven**[*†]

[*] University of Southern California, Los Angeles, CA, USA

[†] Reduced Energy Microsystems, San Francisco, CA, USA

Reduced Energy Microsystems

Founded 2014

Building low-power asynchronous circuits

- Encryption ciphers
- IoT Microcontrollers

Commercializing resilient asynchronous design

Application

DoD contract with Galois, USC, and REM

Hardware implementation of Simon and Speck block ciphers

- Designed for low-power → Resilient Bundled Data
- Iterative operation → Conditional Communication

Starting from a high level language

- SystemVerilogCSP

Outline

Resilient Bundled Data

SystemVerilogCSP

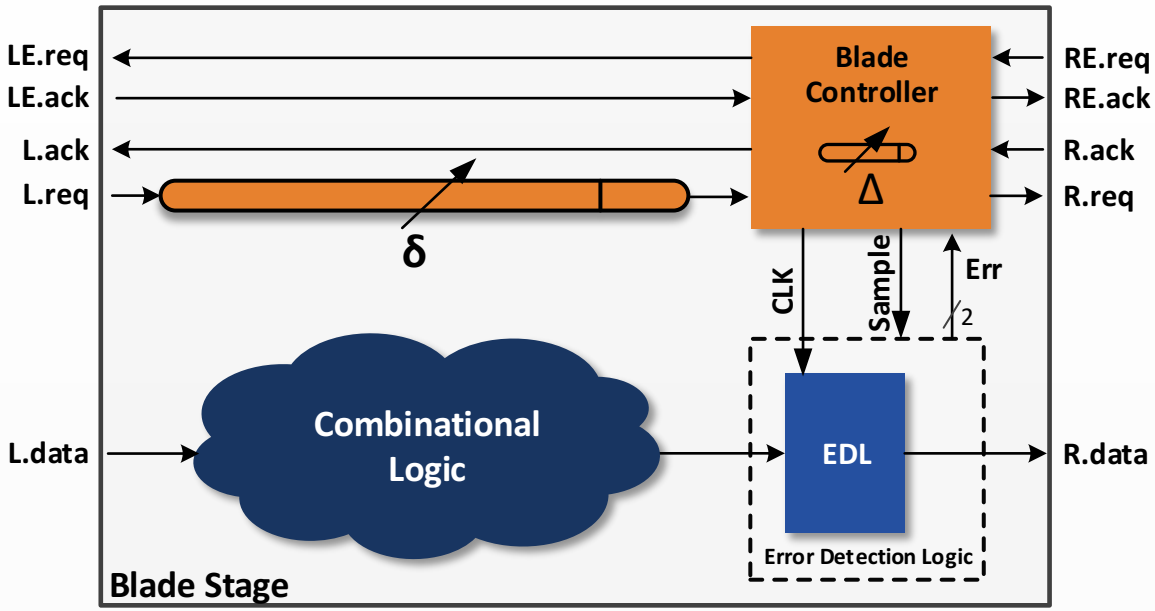
Slackless controllers

- RECV
- SEND

Testability

Conclusions

Resilient Bundled Data



Blade design template

- Error detecting latches monitor timing violations
- Control path pauses until violations are resolved

SystemVerilogCSP

```
`BUNDLED_DATA(72)
`BUNDLED_DATA(48)
module simon(
  bundled_data_72.In  key,
  bundled_data_48.In  blk,
  bundled_data_48.Out  enc );
...
always
begin
  count = 6'h0;
  forever
  begin
    if ((count == 6'h0))
    begin
        key.Receive(k);
        blk.Receive(pt);
    end
    ...
    if ((count == 6'h21))
      enc.Send(pt);
```

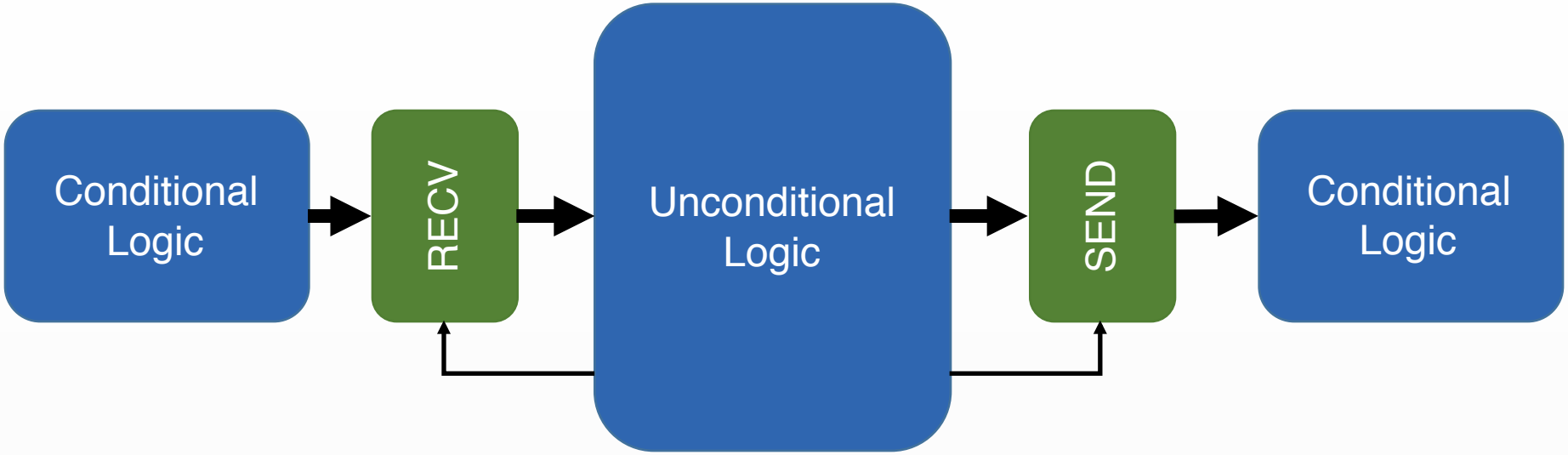
SystemVerilog Interfaces

- Abstracts channels
- Enables implicit conditional communication

SVC2RTL

- Converts SVC to synthesizable RTL
- Adds SEND/RECV blocks

SystemVerilogCSP

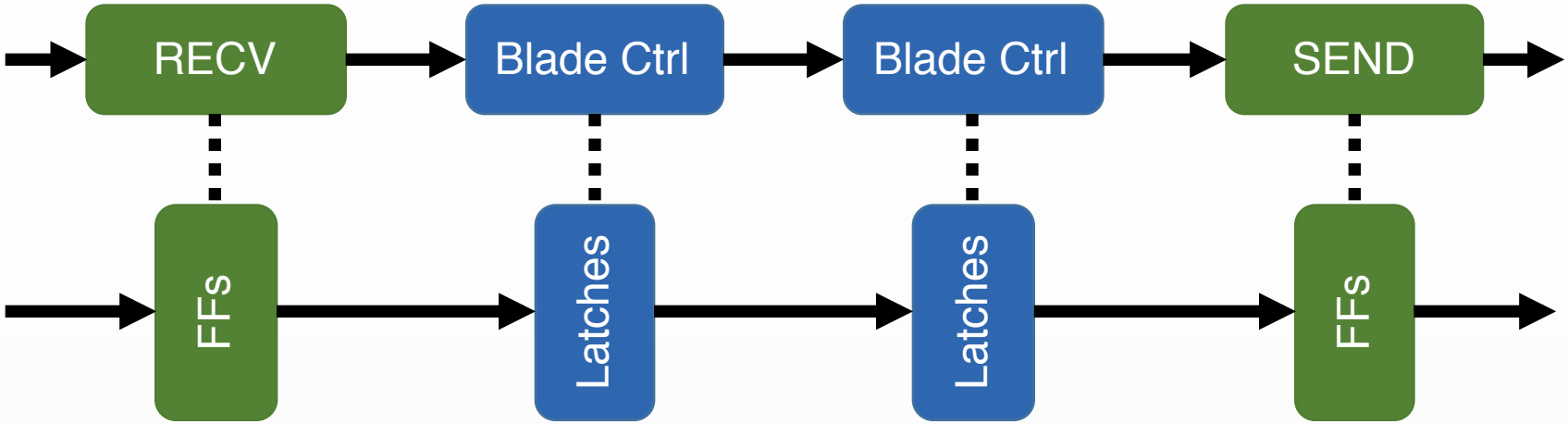


Communication on I/O channels

- Selectively *receive* a token on input channels
- Selectively *send* a token on output channels

Unconditional logic sees communication every cycle

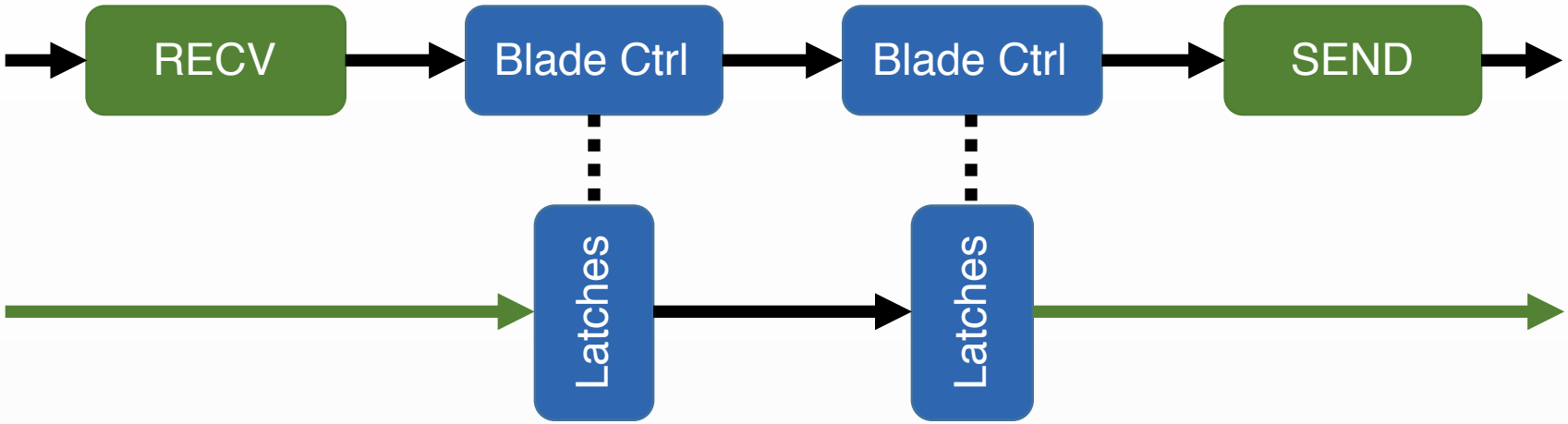
Slackless Controllers



SEND/RECV stages following BD paradigm

- Increases area substaintally for wide datapaths
- Wastes power

Slackless Controllers



Additional sequentials are unnecessary

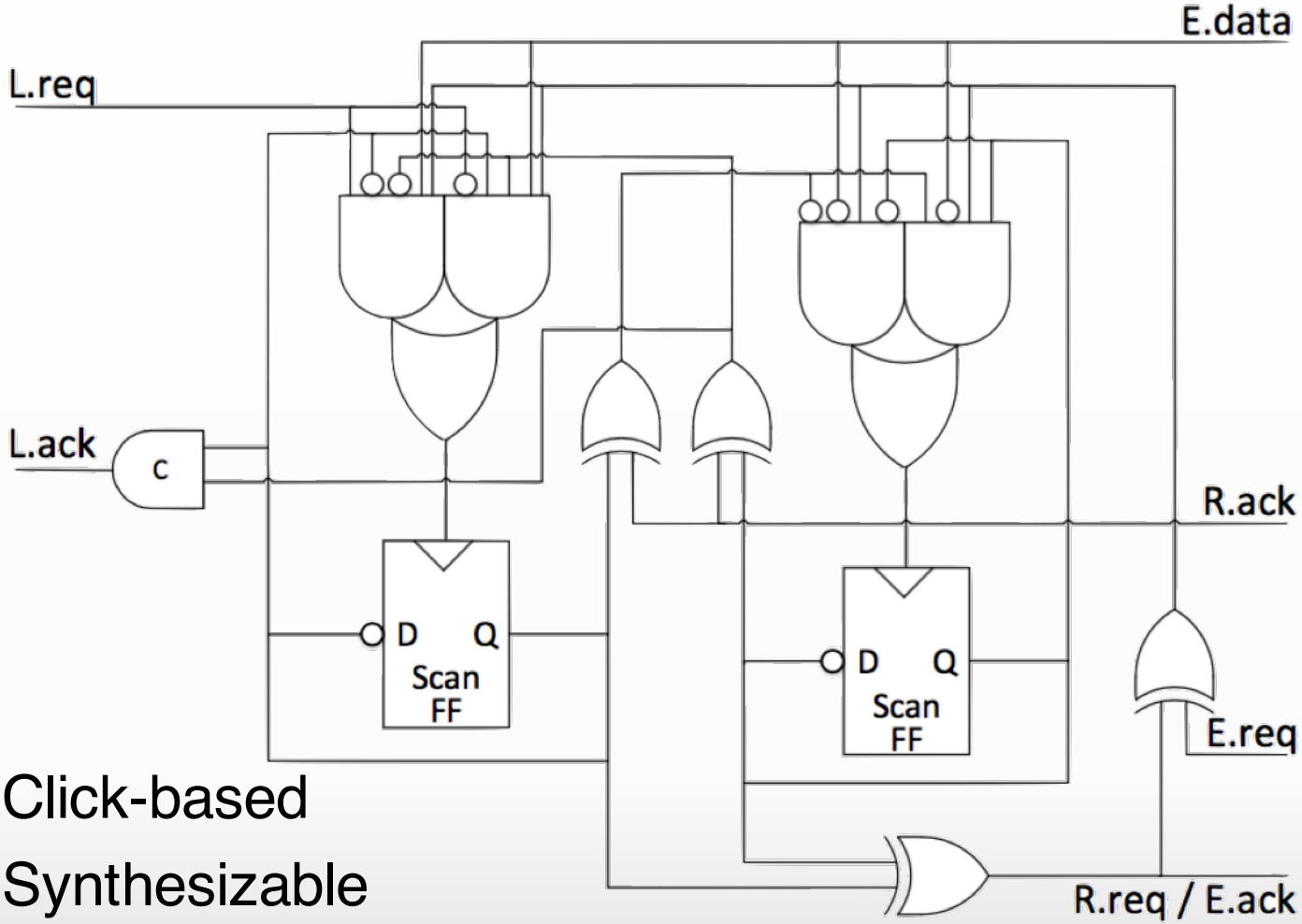
- Iterative logic ignores inputs during non-RECV cycle
- Next stage ignores changing output until *Req* is asserted

No impact on performance

- Iterative stage is the bottleneck

Externally looks like normal BD stage

RECV Controller

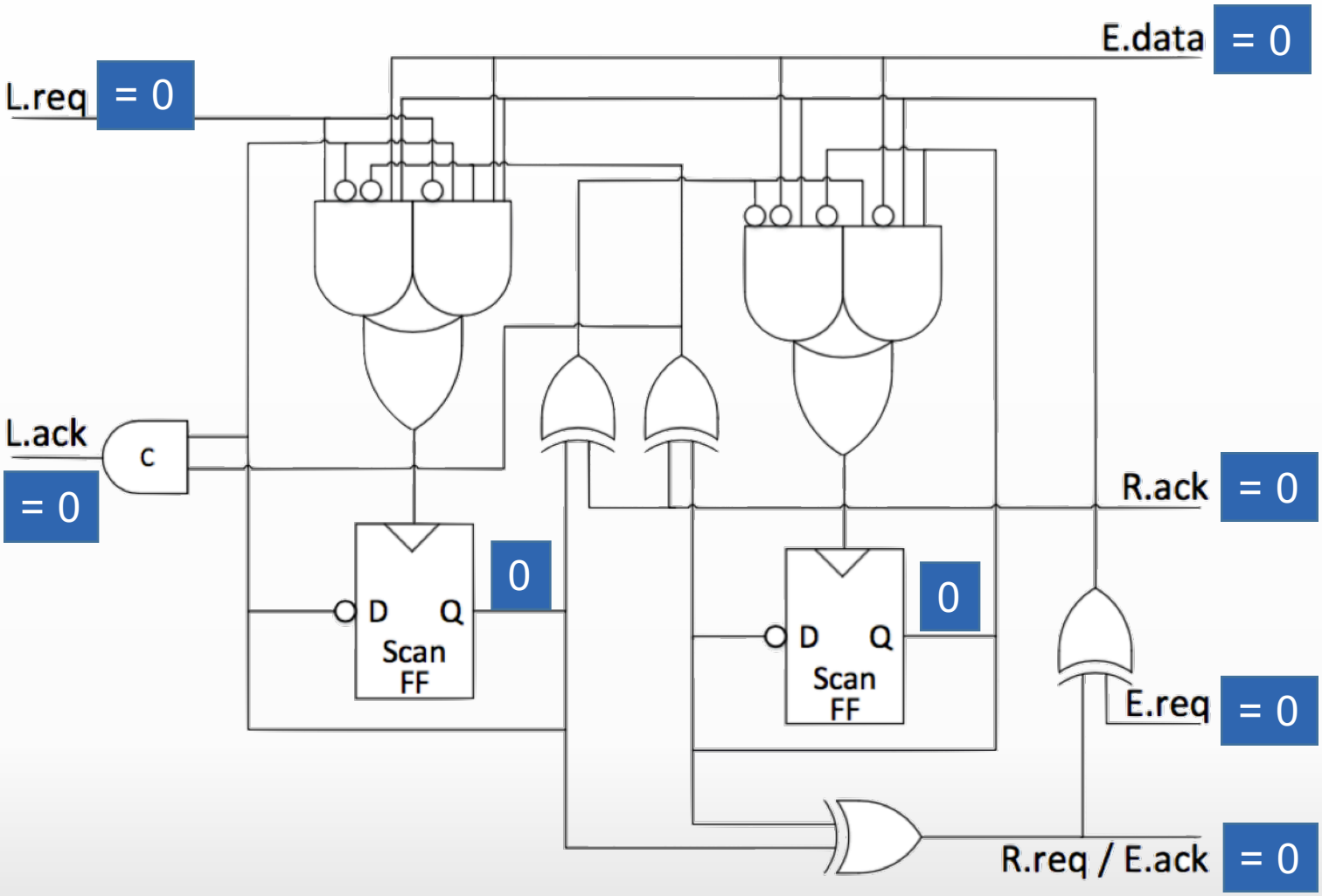


Click-based

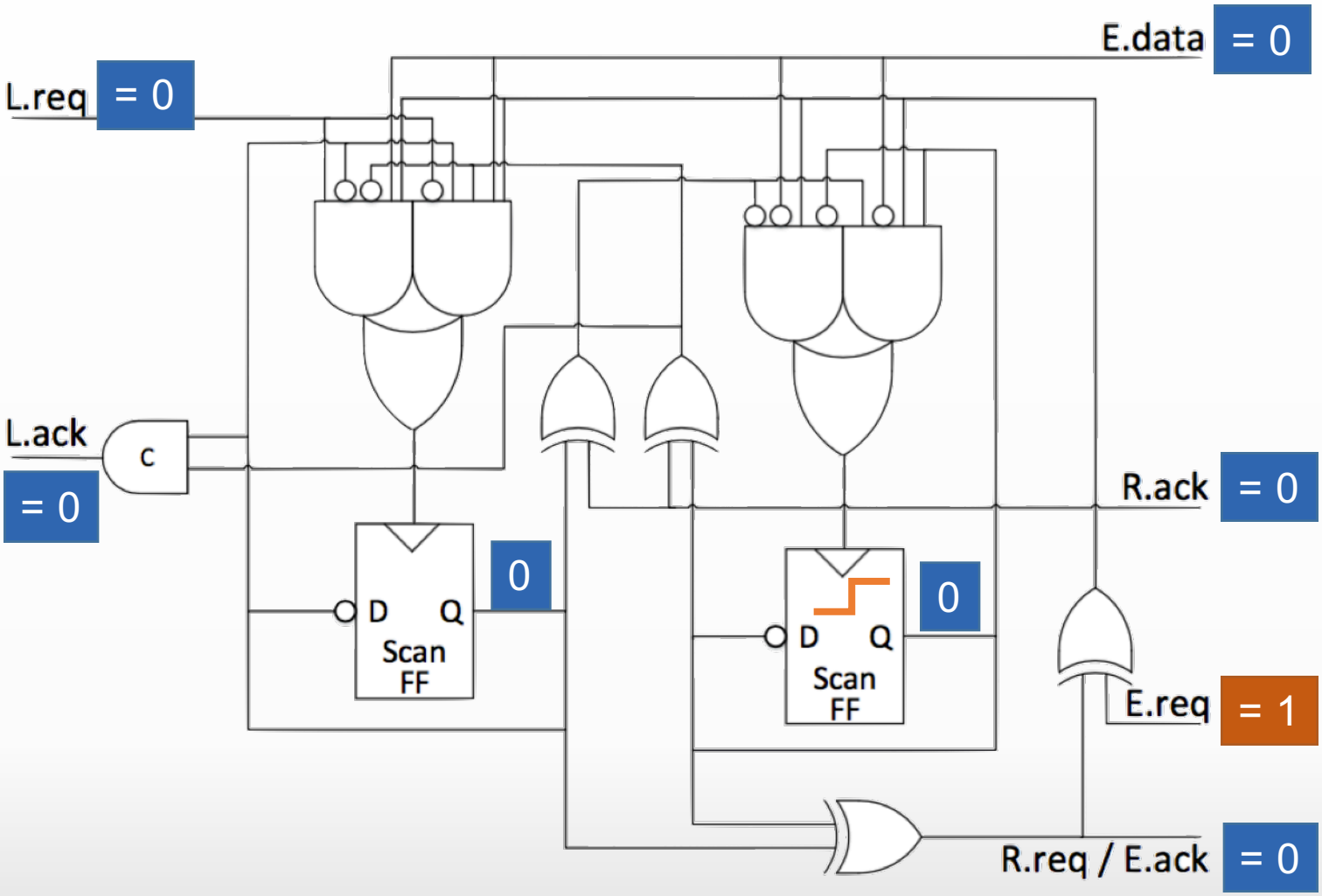
Synthesizable

Custom C-element

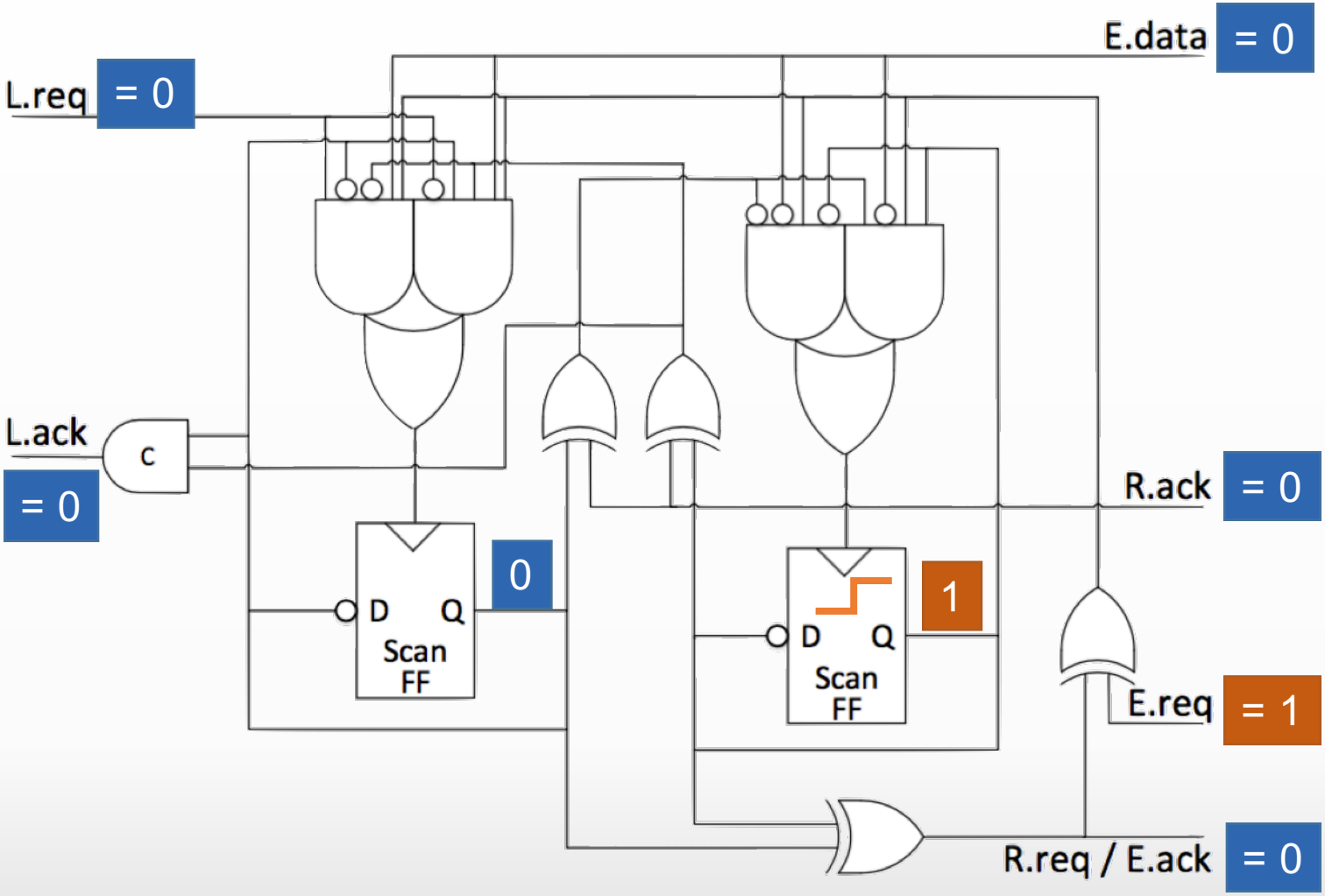
RECV Controller



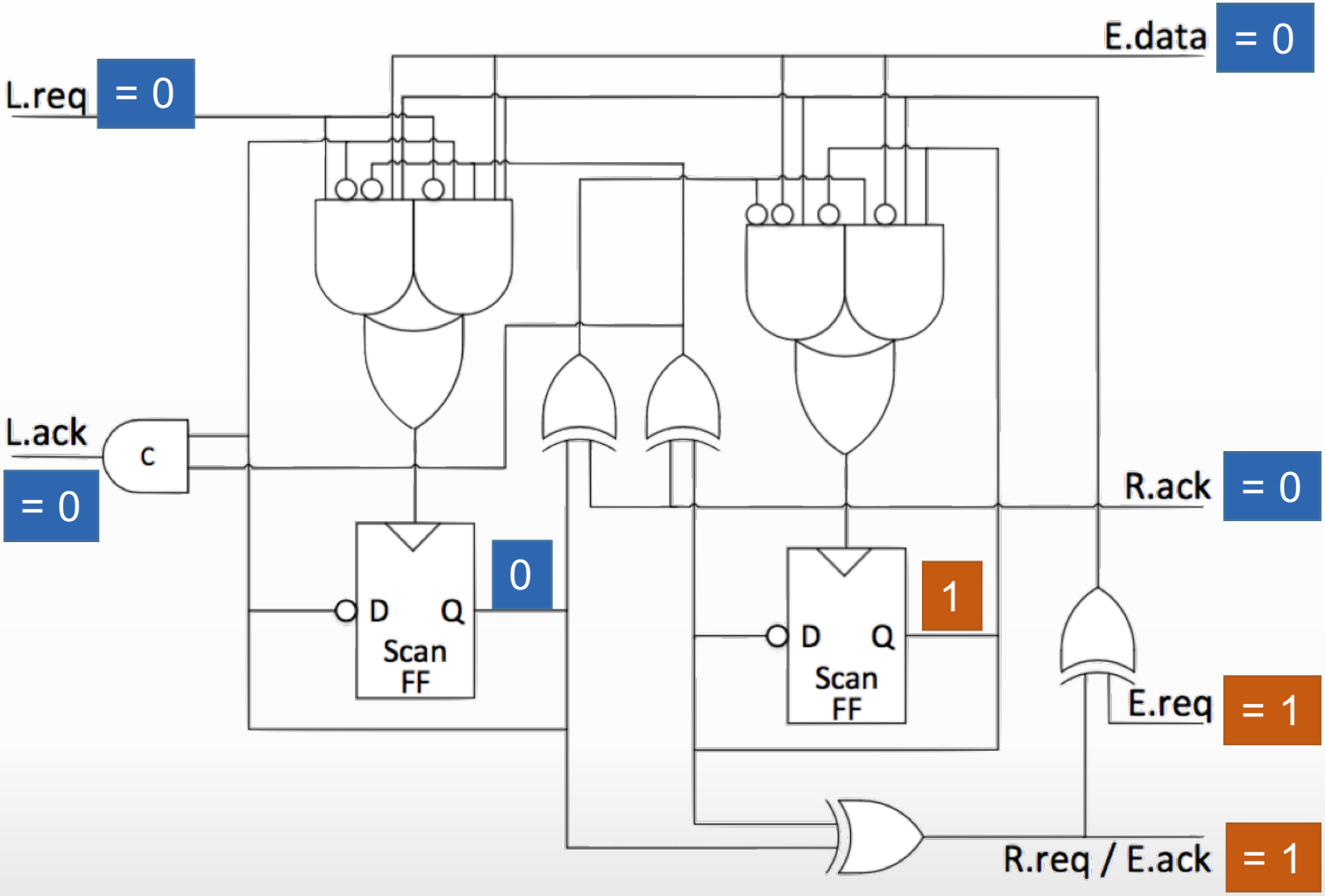
RECV Controller



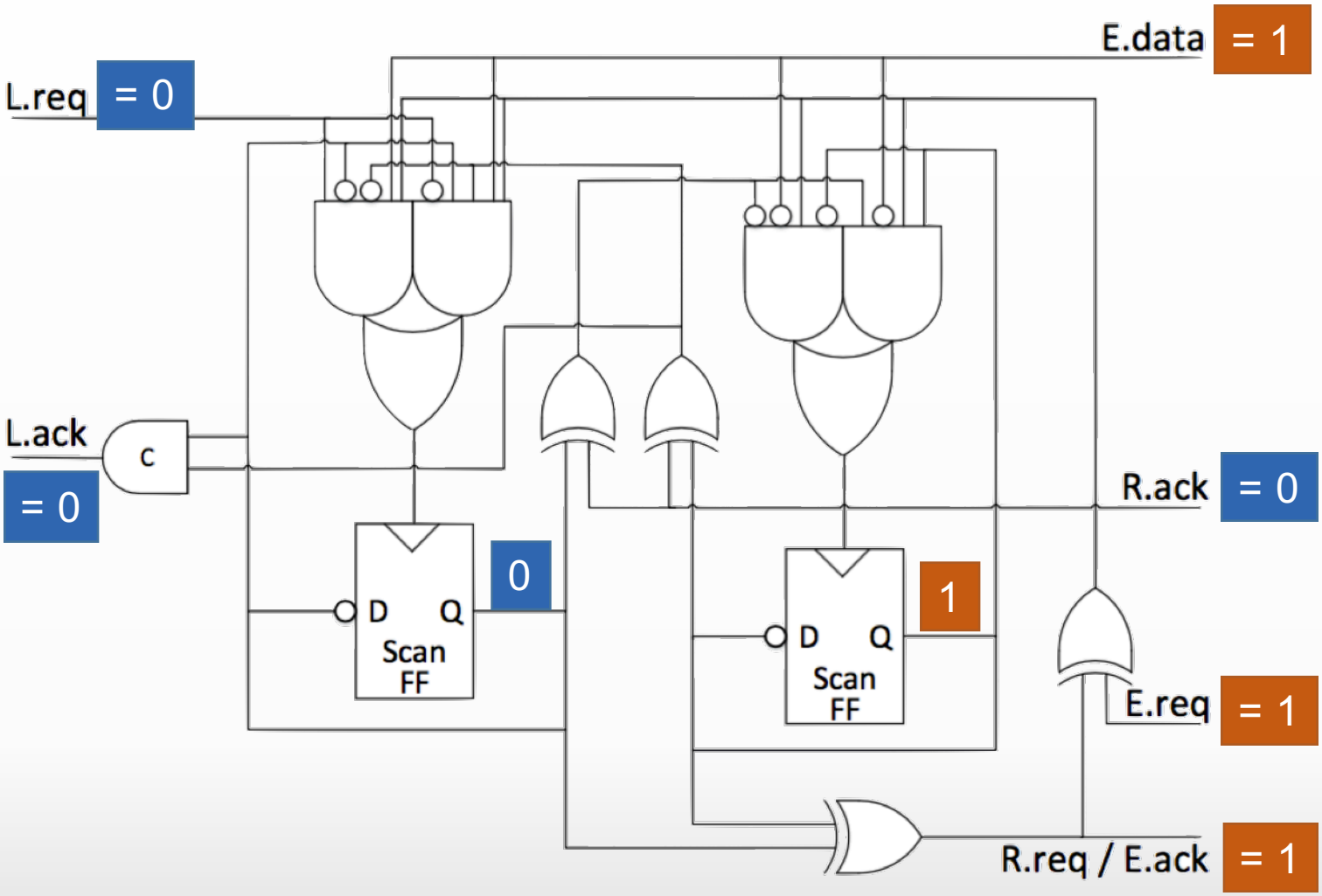
RECV Controller



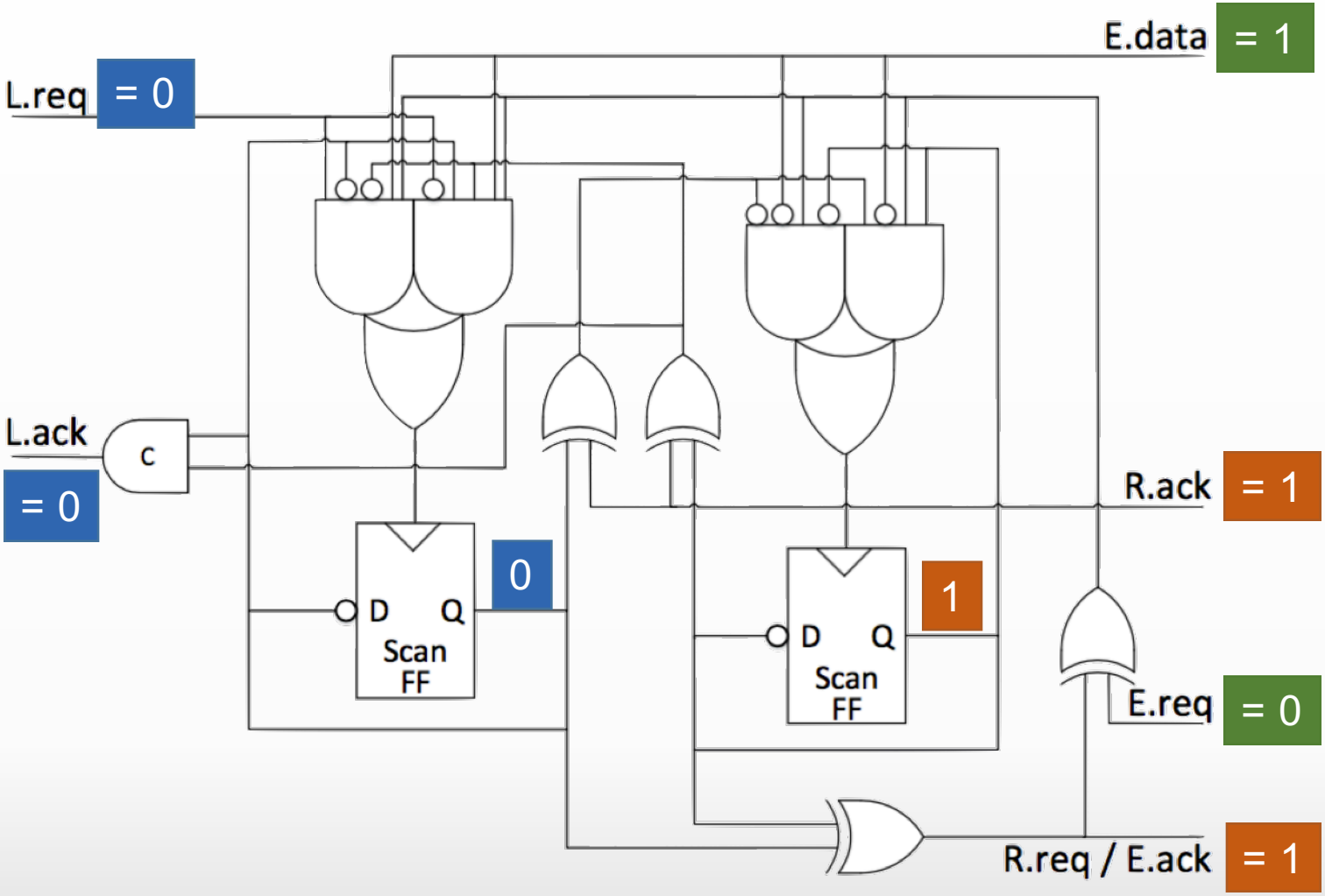
RECV Controller



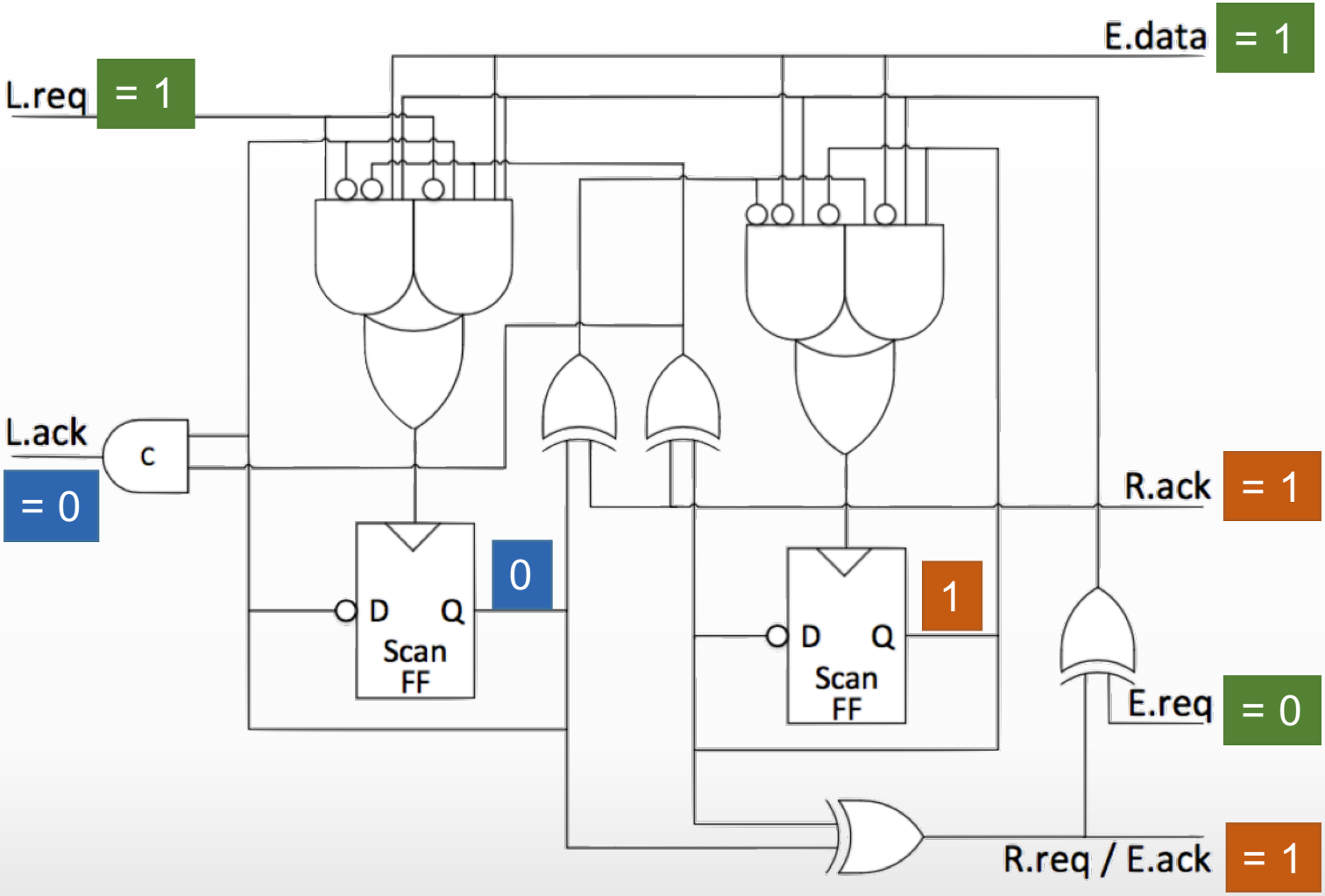
RECV Controller



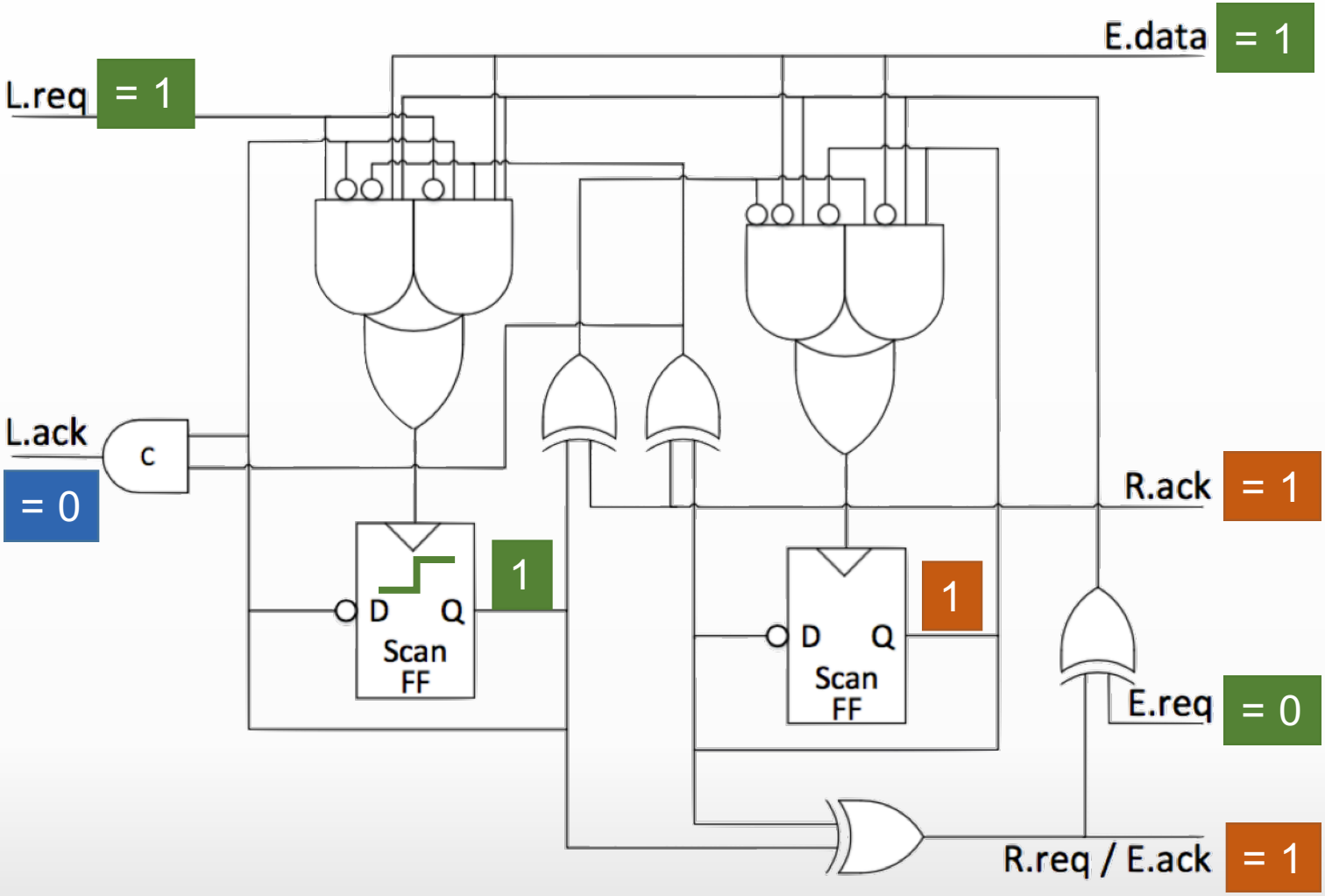
RECV Controller



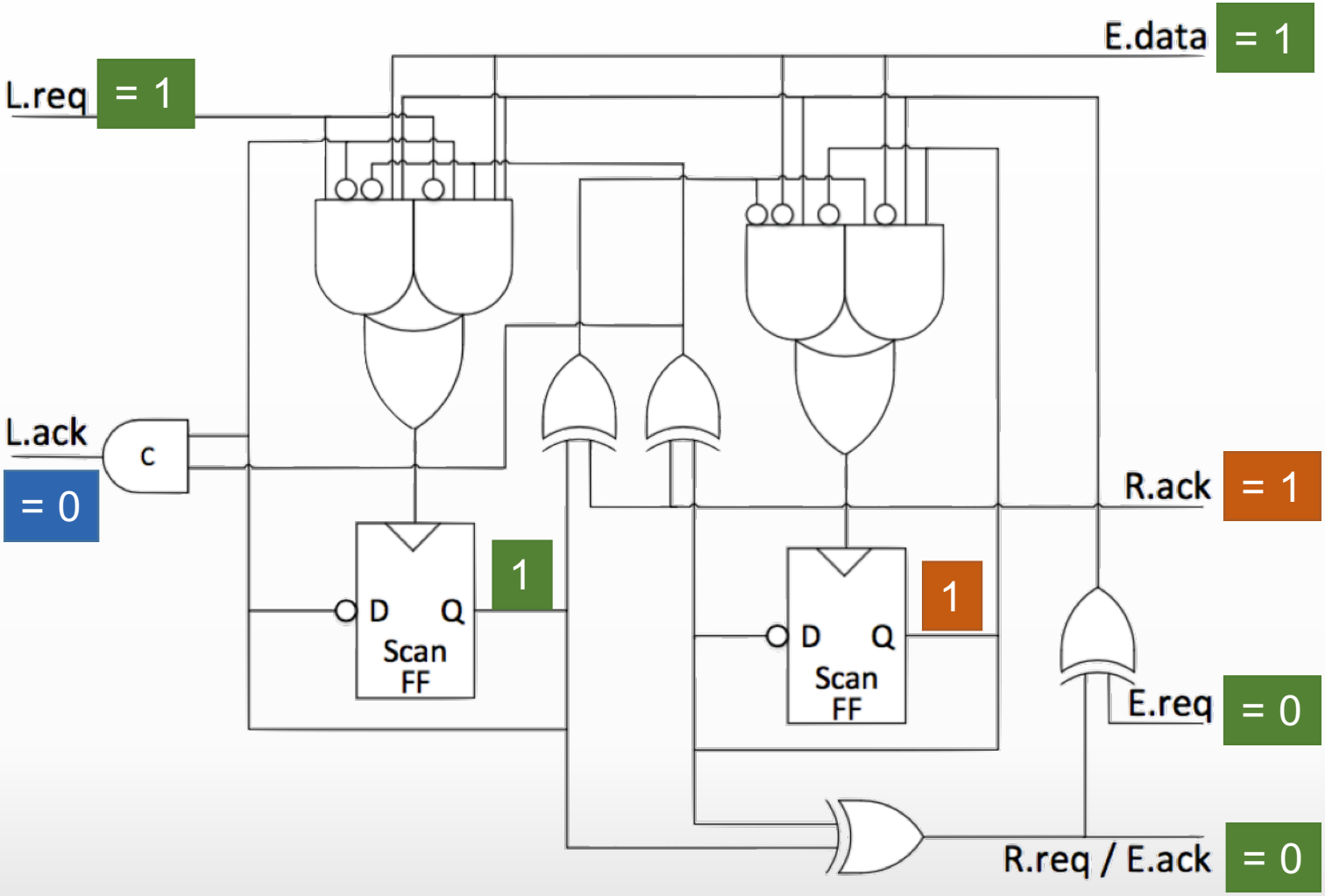
RECV Controller



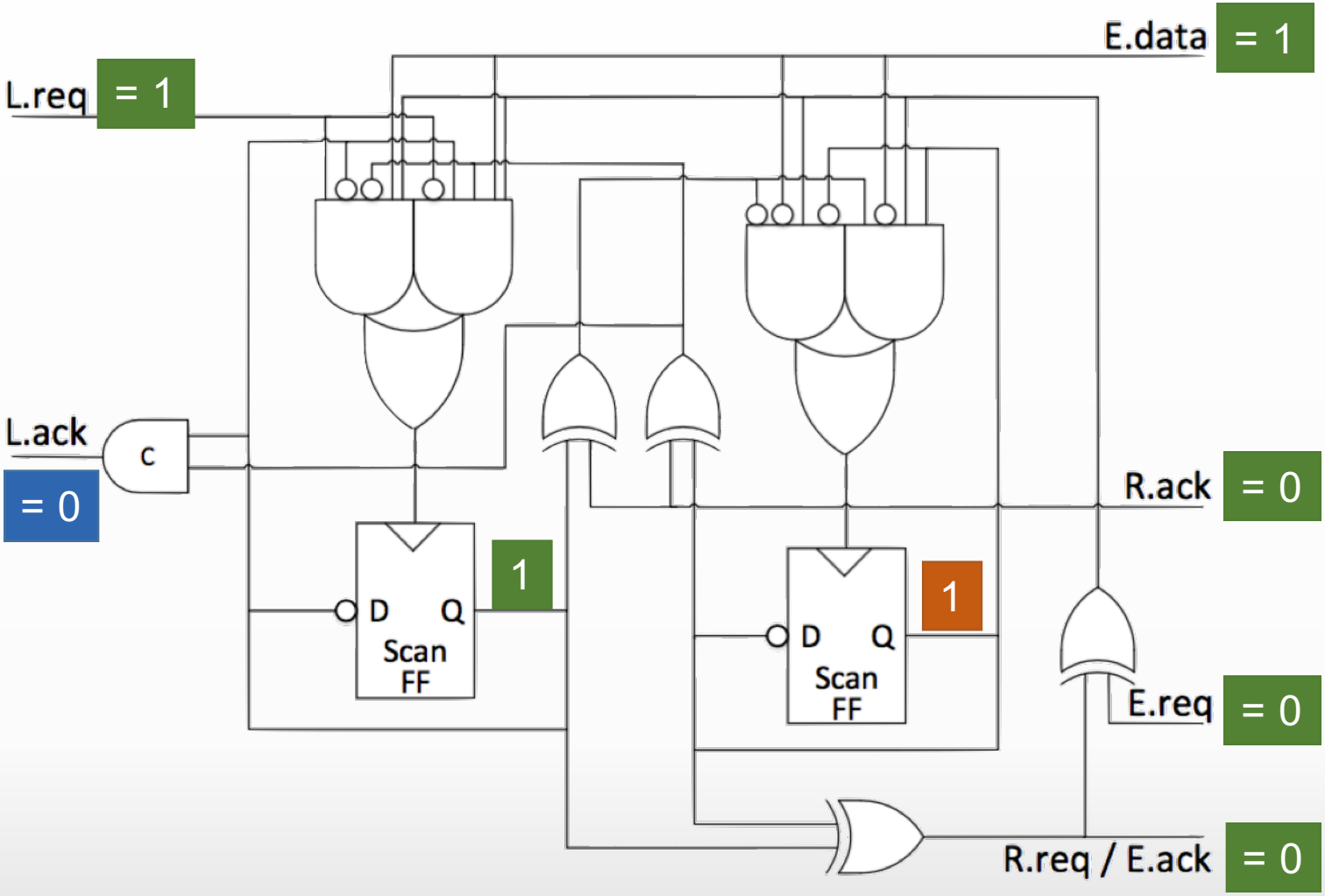
RECV Controller



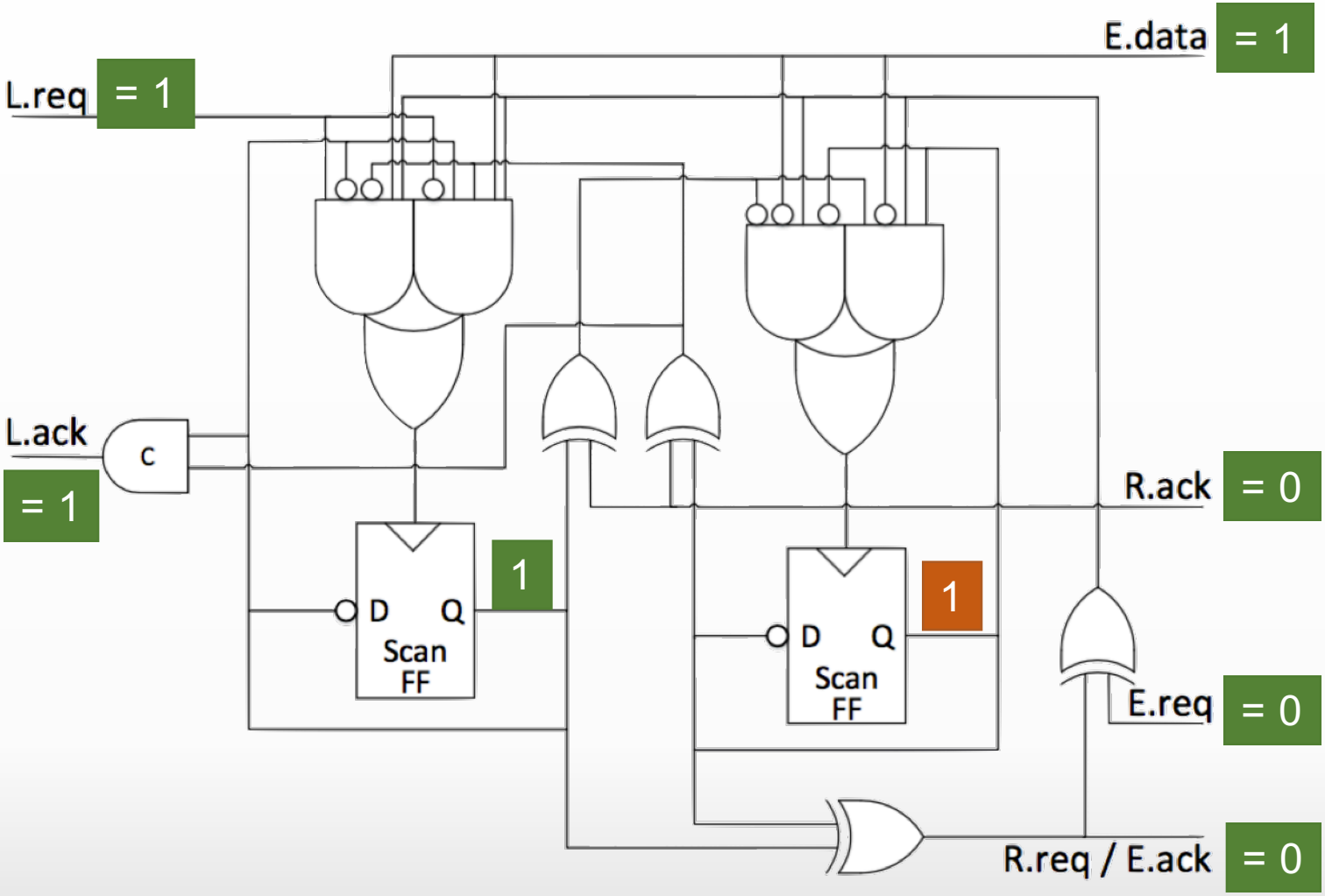
RECV Controller



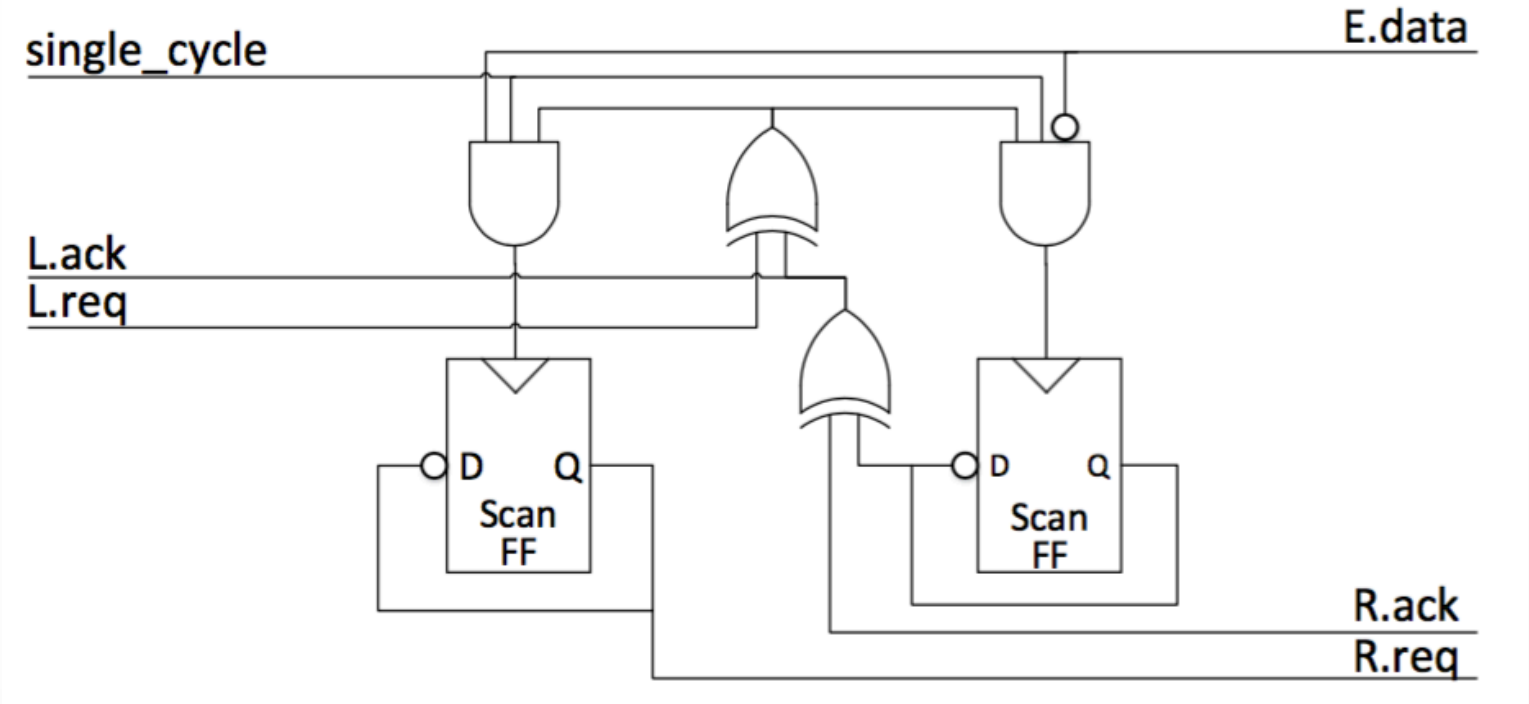
RECV Controller



RECV Controller



SEND Controller

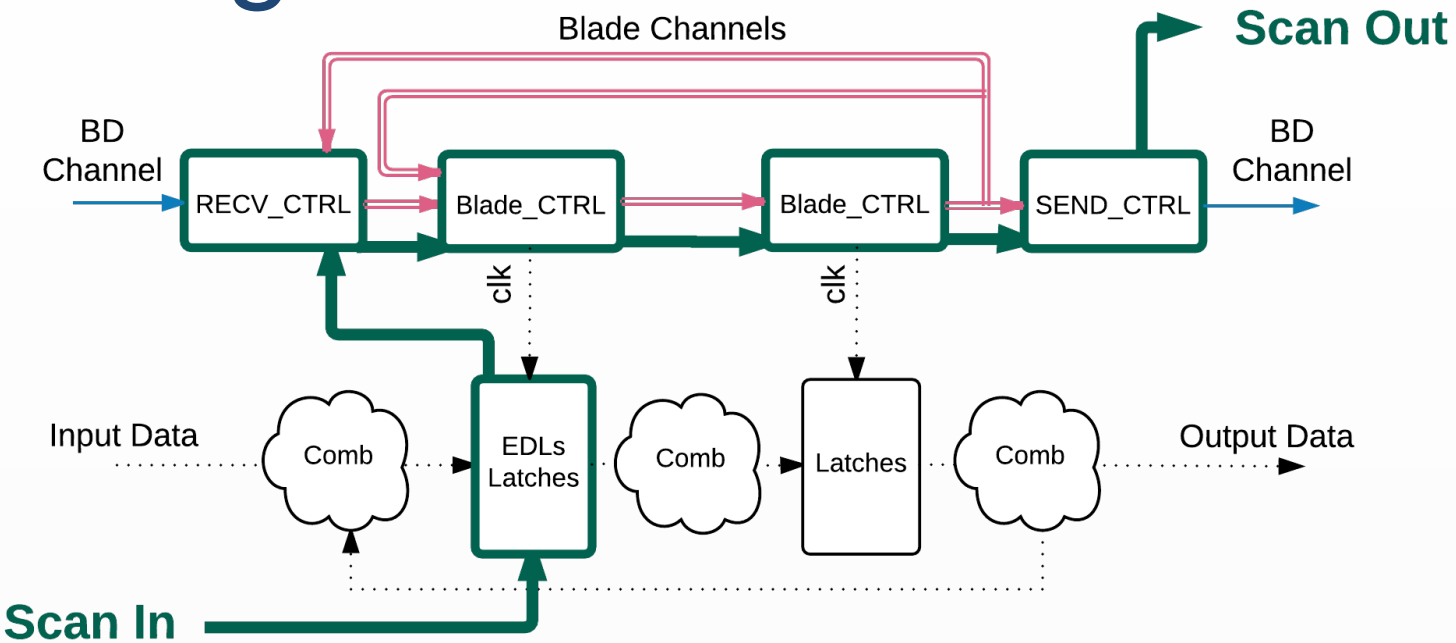


E channel's req and ack shared with *L*

single_cycle disables the SEND controller

- Prevents more than one token passing

Testing



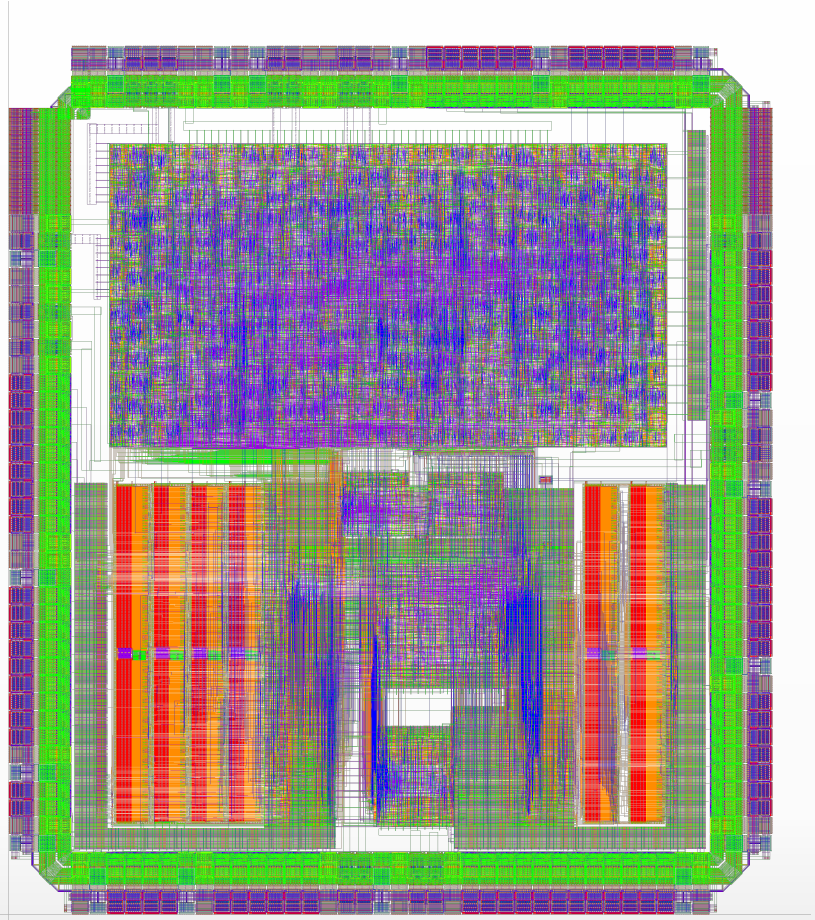
Click controllers enable use of scannable FFs

- Initialize controllers to particular state

Master stage of Blade uses scannable EDLs

Functional testing through *single_cycle* in SEND

Conclusions



3.3M transitor chip currently in fabrication

First Blade implementation in silicon

Entire design synthesized from SystemVerilogCSP

- Using Synopsys EDA tools

Saved ~14% area via slackess design

Questions?

In collaboration with:

|galois|

William Koven

koven@remicro.com