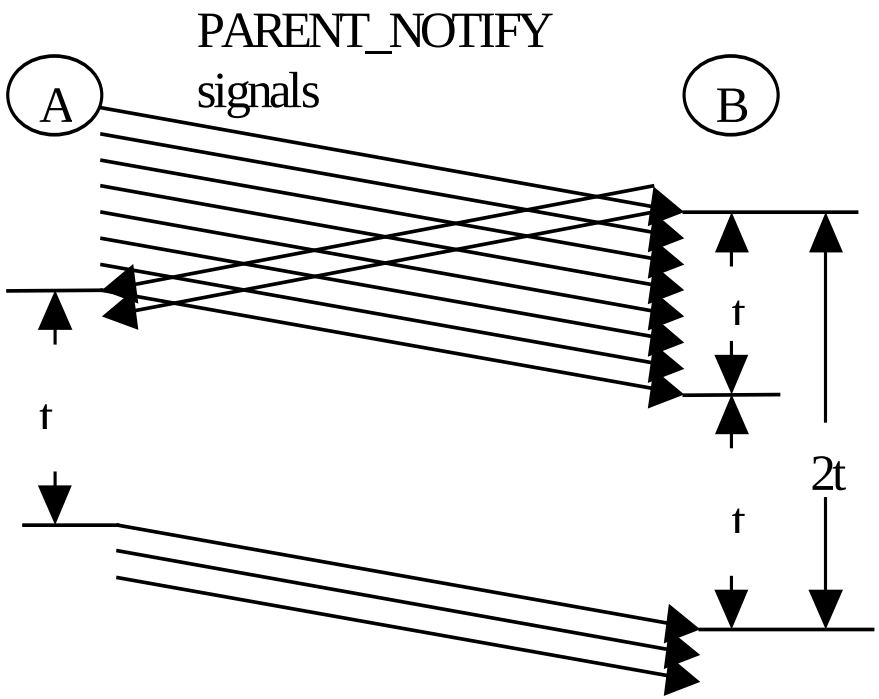
SubPhy Root Contention

Longer cables => Longer root contention constants



- To ensure that node B can see IDLE from node A, ROOT_CONTEND_FAST must be at least t
- To ensure that node B can see next PARENT_NOTIFY from node A, ROOT_CONTEND_SLOW must be at least 2t
- t = 2(cable_delay + Phy_delay)
- cable_delay = propagation_speed * length

Contention times

- Mr. Taka H. Fujimori provided the following cable propagation speed data on the p1394b reflector

UTP Category 5	5.5 ns/m
STP	5.56 ns/m
(UTP Category 4	5.7 ns/m)
(UTP Category 3	5.7 ns/m)
MM Glass Fiber	5.0 ns/m
GI/SI-POF	5.5 ns/m(expected)
1394 STP cable	5.05 ns/m

- Max 1394B cable length and Phy_Delay will determine the times needed for ROOT_CONTEND_FAST and ROOT_CONTEND_SLOW
- On the order of 1.5 and 3.0uS for 100m

  or 6 and 12uS for 500m

Alternative Proposals

- Alistair Coles/Eric Deliot
  - » 1394B Phy/subPhy negotiates potential contention results during Beta start-up
- Takayuki Nyu
  - » Nodes measure duration of PARENT_NOTIFY signals to resolve contention

Avoiding SubPhy Becoming Root

- The SubPhy could simply forward parent and child notify signals
  - » The root contention constants would have to be really big
    - – big enough to allow for 3 cables and 2 intermediate Phy delays
  - » Wouldn't work between 1394-1995 or 1394A nodes which don't have these really big constants
- The SubPhy can participate in the Tree-ID process using a modified state diagram
  - » must be compatible on the short side with regular nodes
  - » must determine relationship on long side before committing to anything on the short side
  - » must complete handshake with children before completing handshake with parent
  - » must not block progress when contention does not occur across the long side

SubPhy Tree-ID

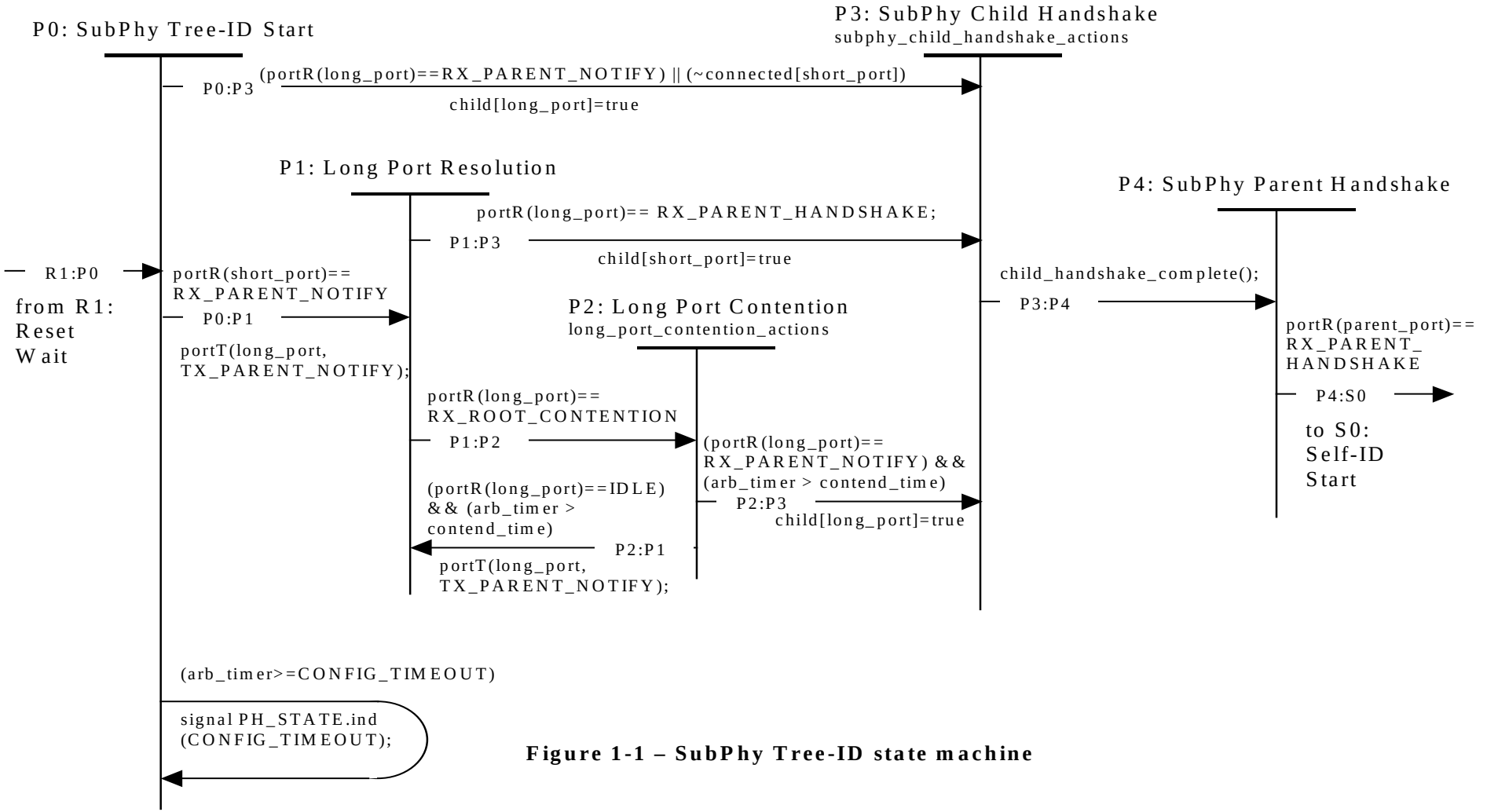
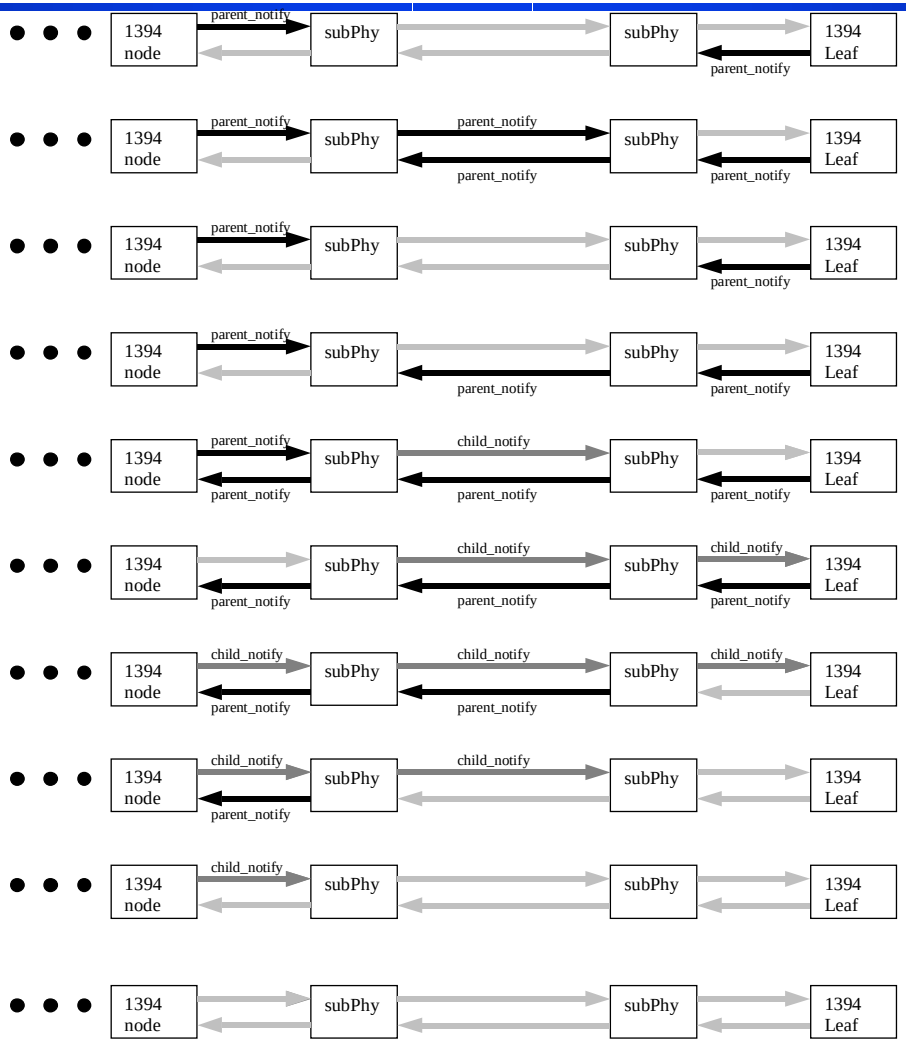


Figure 1-1 – SubPhy Tree-ID state machine

Example with Contention



Example with No Contention