

ssn

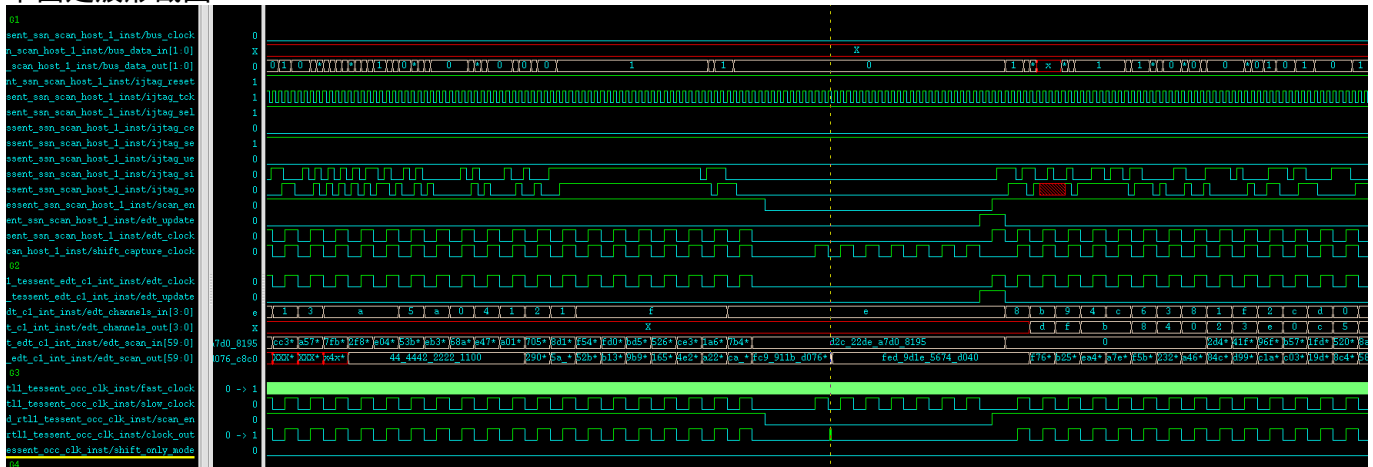
1. ijtag streaming interface

设置streaming interface为ijtag, 即可以通过JTAG接口来输入scan test pattern

此时edt 和 occ slow clock都来自于ssn host, 由tck分频而来。scan_en=0时, occ切换到fast clock进行正常的at-speed capture操作。

```
set_ssn_options -streaming_interface ijtag
```

下面是波形截图



2. retarget 2 tile

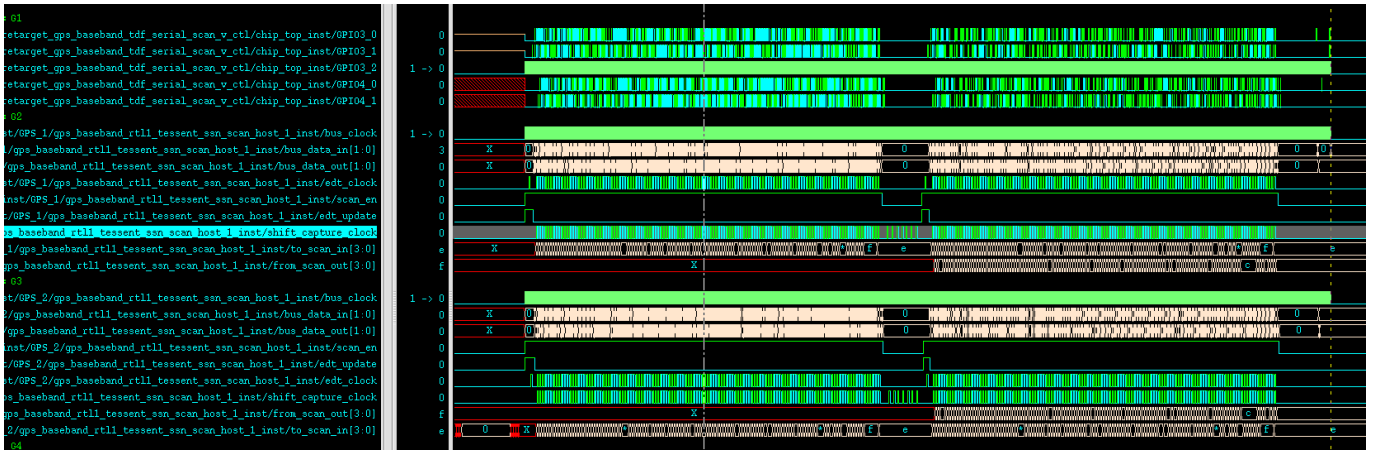
retarget时, 建议总是打开on_chip_compare_enable。这样会在ssn host里面保存scan sticky结果, 方便jtag读取结果。

```
set_core_instance_parameters -module
gps_baseband_rttl1_tessent_ssn_scan_host_1
    -parameter_values { on_chip_compare_enable on on_chip_compare_group 1
}

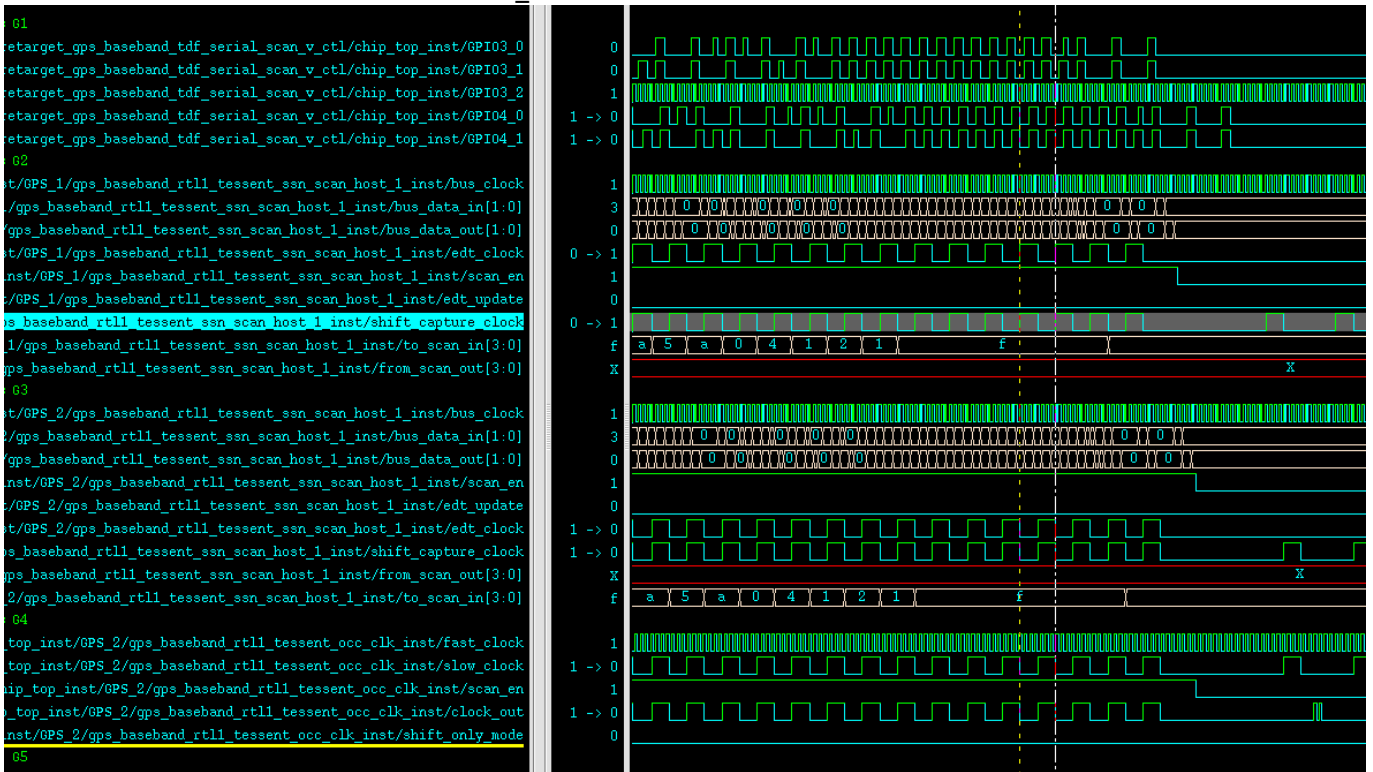
```

2.1 use ssn_bus streaming interface

下图是同时retarget 2 个gps模块的波形图, 其内部edt同时工作, 并且几乎是同时停, 没有带宽浪费。

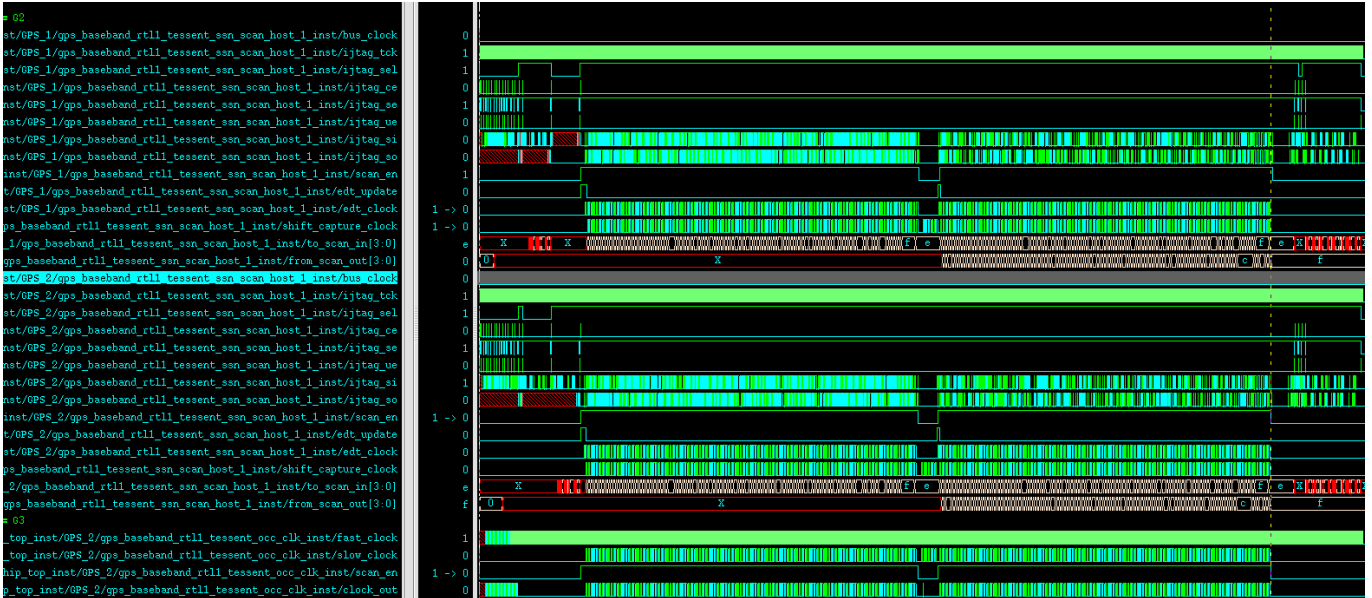


下图是说edt clk & shift clk这些是从bus_clk分频而来的，



2.2 use ijtag streaming interface

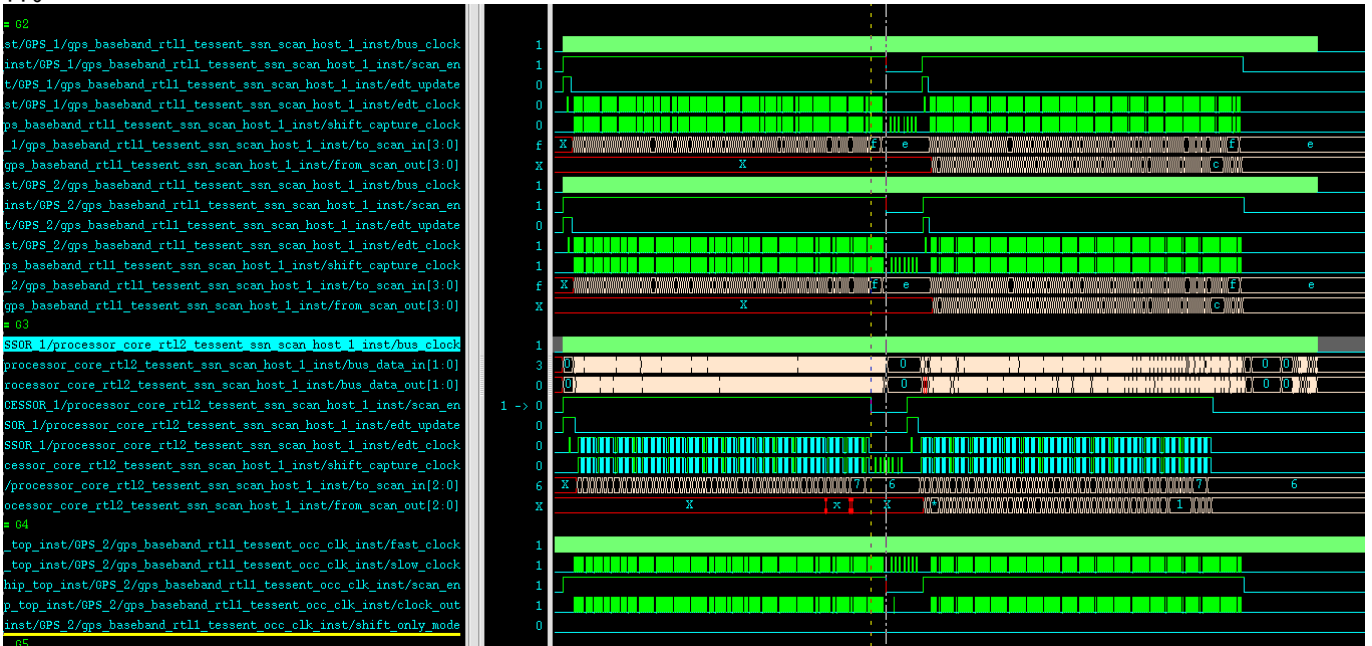
下图是仅使用ijtag接口来传scan pattern从图中看出bus_clk没有时钟 regartet 2个GPS模块，2个edt同时停，没有带宽浪费。



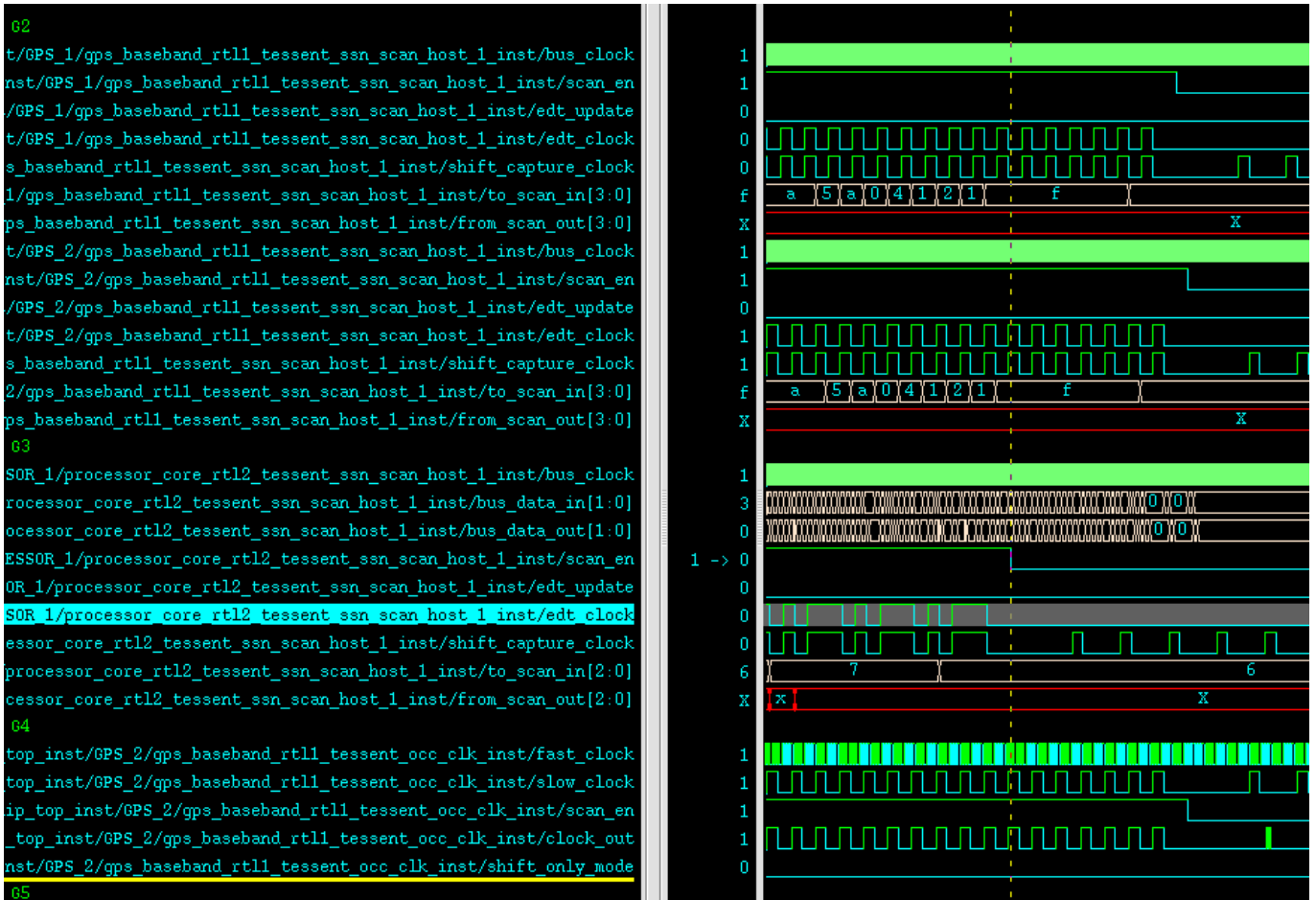
3. retarget 3 tile

3.1 ssn bus scan pattern

下图是3个tile一起retarget的情况，一个process core, 2个gps core, 从图中看出scan_en下降的时间不一样。

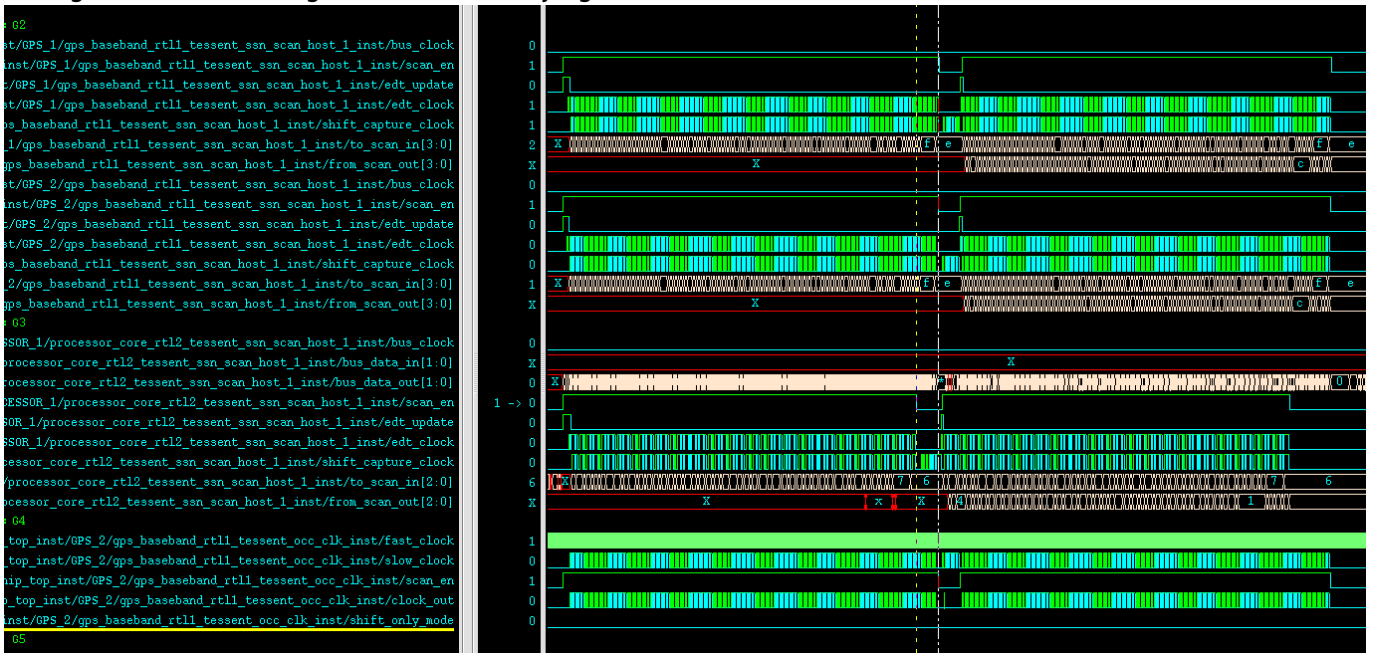


放大，看edt_clk & shift clock 细节，比较明显是process core的edt clk给得慢，gps core给得快（几乎是2个process edt clk的时间，3个gps edt clk）而且 process edt channel宽度更窄一些，说明算法已经在尽量平衡带宽，又保证ssn bus带宽不浪费。

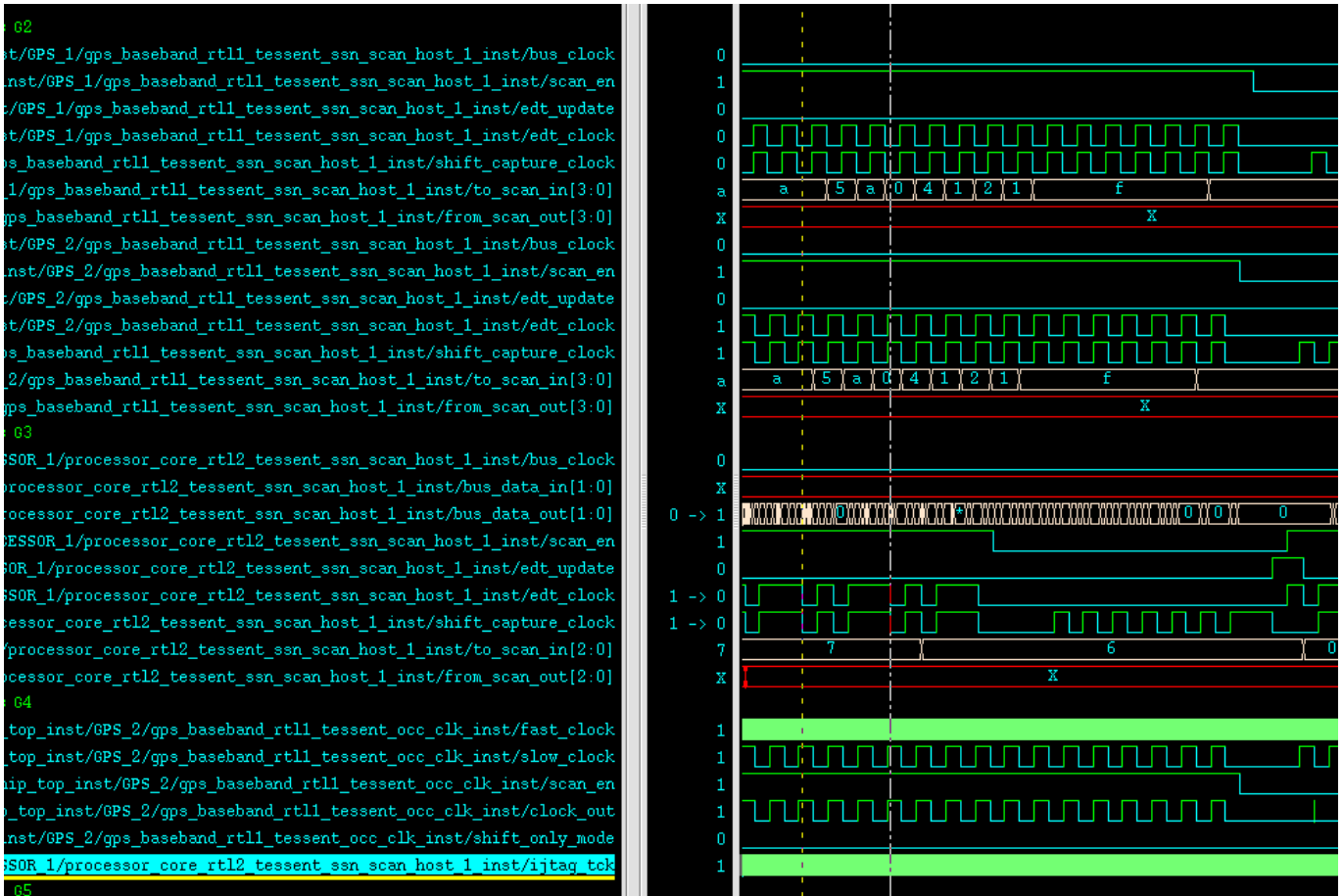


3.2 ijttag scan pattern

retarget 3 tile, streaming interface只使用ijtag



retarget 3 tile, streaming interface只使用ijtag, 放大



4. DefaultChildConfiguration

```

set spec [create_dft_specification -sri_sib_list ssn]
read_config_data -in $spec -from_string {
  SSN {
    DefaultChildConfiguration {
      bus_register_reset :off;
    }
    Datapath(1) {
      output_bus_width : 16;
      Pipeline(1) { //operates at 1/4 rate
      }
      BusFrequencyDivider(1) {
        frequency_ratio : 4;
      }
      Pipeline(2) { //operates at full rate
      }
    }
  }
}

```

5. extrace_icl

如果抽icl时遇到说ssn path没有定义，可以使用如下方式定义一下再抽取icl

```
SETUP> add_icl_ssn_datapaths {1}  
SETUP> set_icl_ssn_datapath_ports -name 1 -clock_inputs ssn_bus_clock \  
-bus_data_inputs ssn_bus_data_in[15:0] -bus_data_outputs  
ssn_bus_data_out[15:0]  
SETUP> extract_icl
```

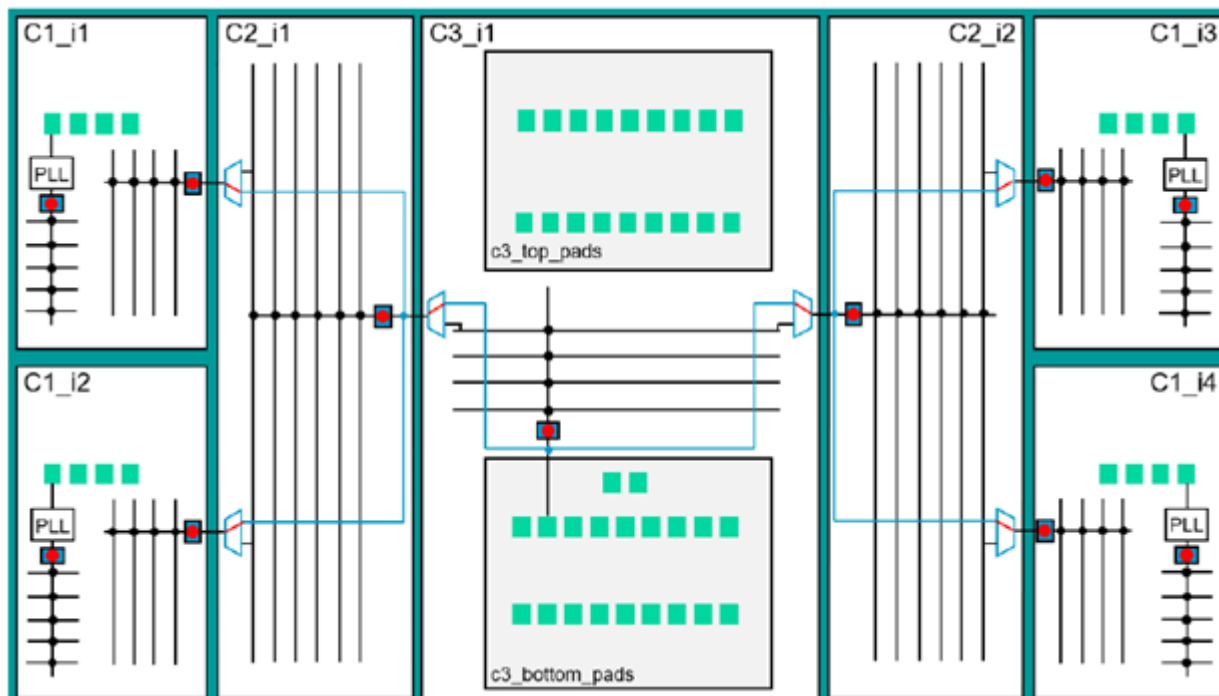
5.1 不同SSN DATAPATH宽度集成

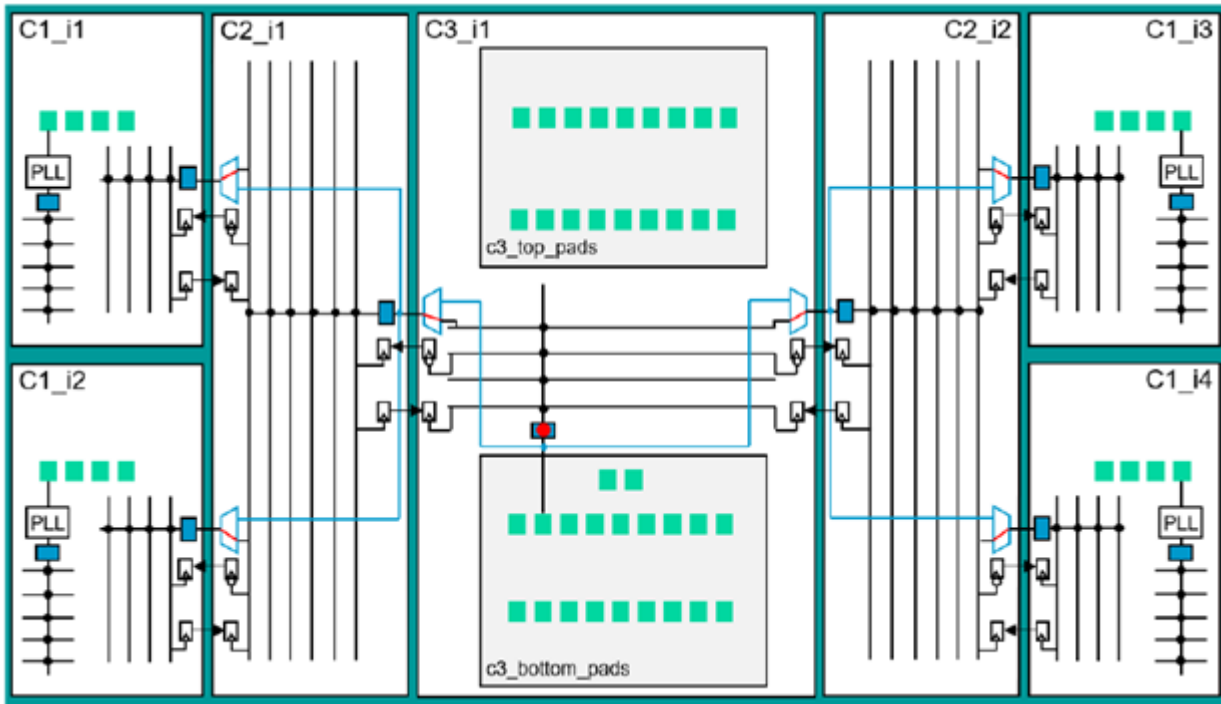
当在一个SSN BUS上有不同宽度的SSH需要集成时，要注意用ssn mux隔开，

比如SSN BUS主宽度为10，某个SSH宽度为4，不能直接将SSH接入BUS中一定需要SSH先接一个宽度为10的ssn mux, 只是ssn mux与SSH之间的宽度为4而ssn mux的主路径宽度依然是10

这样操作的话，自动抽取ICL也不会有问题。

6. clock arch?





7. reducing clock skew

