

# 2023年12月21日交换机实验-测试 副本

---

1. 交换机的连接
2. 进入外存or内存
3. 设置环境变量
4. 编译P4程序
5. 运行P4可执行文件
6. 进入pm配置端口，显示端口信息
7. 添加端口，并使能端口
8. 退出pm进入bfrt配置配置流表？

## 1. 交换机的连接

- 串口连接
  - 白色交换机：（波特率）9600
  - 黑色交换机：115200
- ssh远程连接
  - ssh root@192.168.1.209
  - 密码：onl

## 2. 进入外存or内存

```
▼ Shell |
1 root@bmc:~# sol.sh
```

## 3. 设置环境变量

```
1 source set_sde.bash
```

其中 set\_sde.bash 文件为：

```
1 root@localhost:~/bf-sde-9.2.0# cat set_sde.bash
2 export SDE=/root/bf-sde-9.2.0
3 export SDE_INSTALL=/root/bf-sde-9.2.0/install
4 export PATH=$PATH:$SDE_INSTALL/bin
```

```
1 root@localhost:~ # cd bf-sde-9.5.2/
2 root@localhost:~/bf-sde-9.5.2 # ls
3 bf_drivers.log          CMakeCache.txt          pkgsrc
4 bf_drivers.log.0        CMakeFiles              README
5 bf_drivers.log.1        CMakeLists.txt          run_bfshell.sh
6 bf_drivers.log.2        extract_all.sh           run_p4_tests.sh
7 bf_drivers.log.3        install                  run_switchd.sh
8 bf_drivers.log.4        install.bk               run_tofino_model.sh
9 bf-p4c-prefix           logs                     set_sde.bash
10 bf-p4i-prefix           p4_build.sh             simple_l3.p4
11 bf-p4o-prefix           p4runtime_update_config.py tools
12 bf-sde-9.5.2.manifest  p4studio                 zlog-cfg-cur
13 build                   p4studio_build
14 cmake                   packages
15 root@localhost:~/bf-sde-9.5.2 # . set_sde.bash
16 Using bf-sde-9.5.2 in /root/bf-sde-9.5.2
17
18
```

## 4. 编译P4程序

```
1 ./p4_build.sh <path-to-p4src.p4>
```

`${SDE}=/root/bf-sde-9.5.2`

▼ Shell

```
1 root@localhost:~ # $SDE/p4_build.sh simple_port.p4
2 Using SDE /root/bf-sde-9.5.2
3 Using SDE_INSTALL /root/bf-sde-9.5.2/install
4 Using SDE version bf-sde-9.5.2
5
6 OS Name: "Open Network Linux OS ONL-onf-ONLPv2, 2021-10-27.19
7 This system has 8GB of RAM and 8 CPU(s)
8 Parallelization: Recommended: -j4 Actual: -j4
9
10 Compiling for p4_16/tna
11 P4 compiler path: /root/bf-sde-9.5.2/install/bin/bf-p4c
12 P4 compiler version: 9.5.2 (SHA: 640ad11) (p4c-based)
13 Build Dir: /root/bf-sde-9.5.2/build/p4-build/simple_port
14 Logs Dir: /root/bf-sde-9.5.2/logs/p4-build/simple_port
15
16 Building simple_port CLEAR CONFIGURE MAKE INSTALL ... DONE
17
```

## 5. 运行P4可执行文件

▼ Shell

```
1 root@localhost:~ # $SDE/run_switchd.sh -p simple_port
```

## 6. 进入pm配置端口，显示端口信息

启动ucli

▼ Shell

```
1 bfshell> ucli
2 Starting UCLI from bf-shell
3 Cannot read termcap database;
4 using dumb terminal settings.
5 bf-sde>
6
```

进入pm表



The diagram illustrates the timing of the 100BASE-T4 PHY signals. The signals are: PORT, MAC, D\_P, P/PT, SPEED, FEC, AN, KR, RDY, ADM, OPR, LPBK, FRAMES RX, and FRAM. The signals are shown over 15 clock cycles. The signals are: PORT, MAC, D\_P, P/PT, SPEED, FEC, AN, KR, RDY, ADM, OPR, LPBK, FRAMES RX, and FRAM. The signals are shown over 15 clock cycles. The signals are: PORT, MAC, D\_P, P/PT, SPEED, FEC, AN, KR, RDY, ADM, OPR, LPBK, FRAMES RX, and FRAM. The signals are shown over 15 clock cycles.

4

5 PORT |MAC |D\_P|P/PT|SPEED |FEC |AN|KR|RDY|ADM|OPR|LPBK |FRAMES RX  
|FRAM

6

```
1 bf-sde.pm> exit
2 bfshell>
```

```
1 bfshell>
2 bfshell> bfrt_python
3 cwd : /root/bf-sde-9.2.0
4
5 We've found 1 p4 programs:
6 simple_l3_mcast
7
8 Loading the tables ...
9
10
11 Python 3.4.8+ (default, Aug 10 2022, 16:47:50)
12 Type 'copyright', 'credits' or 'license' for more information
13 IPython 6.5.0.dev -- An enhanced Interactive Python. Type '?' for help.
14
15 bfrt_root>
```

```
1  bfrt_root>bfrt
2  -----> bfrt()
3  Available symbols:
4  dump                - Command
5  info                - Command
6  mirror              - Node
7  pktgen              - Node
8  port                - Node
9  pre                 - Node
10 simple_port         - Node
11 tf1                 - Node
12
13
14 bfrt>
15
16 bfrt>
17
18 bfrt> simple_port
19 ----> simple_port()
20 Available symbols:
21 dump                - Command
22 info                - Command
23 pipe                - Node
24
25
26 bfrt.simple_port>
27
28 bfrt.simple_port> pipe
29 -----> pipe()
30 Available symbols:
31 Ingress              - Node
32 IngressParser        - Node
33 dump                - Command
34 info                - Command
35 snapshot             - Node
36
37
38 bfrt.simple_port.pipe> Ingress
39 -----> Ingress()
40 Available symbols:
41 dump                - Command
42 info                - Command
43 send_t              - Table
44
```

```

45
46 bfrt.simple_port.pipe.Ingress> send_t
47 -----> send_t()
48 BF Runtime CLI Object for pipe.Ingress.send_t table
49
50 Key fields:
51     ig_intr_md.ingress_port          type=EXACT      size=9
52
53
54 Actions, Data fields:
55     NoAction (DefaultOnly)
56         0 data fields:
57     Ingress.send
58         1 data fields:
59             port                      type=BYTE_STREAM size=9
60
61
62 Available Commands:
63 add_from_json
64 add_with_send
65 clear
66 delete
67 dump
68 entry_with_send
69 get
70 get_default
71 get_handle
72 get_key
73 info
74 mod_with_send
75 reset_default
76 set_default_with_NoAction
77 set_default_with_send
78 symmetric_mode_get
79 symmetric_mode_set
80
81 bfrt.simple_port.pipe.Ingress.send_t> add_with_send?
82 Signature: add_with_send(ingress_port=None, port=None, pipe=None, gress_d
83     ir=None,
84 Docstring:
85 Add entry to send_t table with action: Ingress.send
86
87 Parameters:
88 ingress_port          type=EXACT      size=9  default=0
89 port                  type=BYTE_STREAM size=9  default=0
90 File:                 Dynamically generated function. No source code available.
91 Type:                 method

```

```

91
92
93 bfrt.simple_port.pipe.Ingress.send_t>
94
95 bfrt.simple_port.pipe.Ingress.send_t>
96
97 bfrt.simple_port.pipe.Ingress.send_t> add_with_send(132, 140)
98
99 bfrt.simple_port.pipe.Ingress.send_t> dump
100 -----> dump()
101 ----- send_t Dump Start -----
102 Default Entry:
103 Entry data (action : NoAction):
104
105 Entry 0:
106 Entry key:
107     ig_intr_md.ingress_port      : 0x84
108 Entry data (action : Ingress.send):
109     port                        : 0x8C
110
111 ----- send_t Dump End -----
112
113
114 bfrt.simple_port.pipe.Ingress.send_t> exitf_rt cli exited normally.
115 [42D
116 bfshell>
117 bfshell>
118 bfshell>
119 bfshell> exit

```