Team Fnatic
State Machine Test Plan

Test:	State Machine in Reset State
Section:	4.0, 5.3, 6.1, 5.2, 6.5
What to test:	1. At the same time we will test these inputs to the state machine in reset state (The status register status_reg[3:0] reads 4'h0): ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should now read 4'h0. 2. At the same time we will test these inputs to the state machine in reset state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to change to the normal state. The status register status_reg[3:0] should now read 4'h1. 3. At the same time we will test these inputs to the state machine in reset state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0. 4. At the same time we will test these inputs to the state machine in reset state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0. 5. At the same time we will test these inputs to the state machine in reset state: ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are ignored in reset. This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0.

6. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b1
- Maroon = 1'b0
- bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted)
- exp_vio = 1'b0 (export_disable not asserted, no export violation)

Result: Commands are ignored in reset. This will cause the state machine to change to the normal state. The status register status_reg[3:0] should now read 4'h1.

7. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b0
- Maroon = 1'b1
- bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted)
- exp_vio = 1'b0 (export_disable not asserted, no export violation)

Result: Commands are ignored in reset. This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0.

8. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b1
- Maroon = 1'b1
- bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted)
- exp_vio = 1'b0 (export_disable not asserted, no export violation)

Result: Commands are ignored in reset. This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0.

9. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b0
- Maroon = 1'b0
- bad_cmd = 1'b1 (apply bad command)
- exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted)

Result: This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0.

10. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b1
- Maroon = 1'b0
- bad_cmd = 1'b1 (apply bad command)
- exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted)

Result: Commands are ignored in reset. This will cause the state machine to change to the normal state. The status register status_reg[3:0] should now read 4'h1.

11. At the same time we will test these inputs to the state machine in reset state:
- Gold =1'b0
- Maroon = 1'b1
- bad_cmd = 1'b1 (apply bad command)

	● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are ignored in reset. This will cause the state machine to remain in the reset state.The status register status_reg[3:0] should read 4'h0. 12. At the same time we will test these inputs to the state machine in reset state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are ignored in reset. This will cause the state machine to remain in the reset state. The status register status_reg[3:0] should read 4'h0. 13. When in reset state, on asserting rst_b pin, the state machine should stay in the reset state, irrespective of the other inputs. The status register status_reg[3:0] should read 4'h0. **Note**: Other cases (cases of bad_cmd = 1'b0 and exp_vio = 1'b1) are not part of the test since due to the specifications, we can only have either a no command (bad_cmd = 1'b0 and exp_vio = 1'b0), a bad command (bad_cmd = 0), or an export violation command (bad_cmd = 1 and exp_vio = 1). To perform a bad command that will trigger an error state or an export_violation state if export_disable is asserted, we will write 16h '800A to the command register. When testing, we are only looking at one cause of getting into the next state, and code coverage will cover the others. We will clear INT1 and INT2 by writing 1 which is caused by bad command or overflow (INT1) and export violation(INT2).

Test:	State Machine in Normal State
Section:	4.0, 5.3, 6.2, 5.2, 6.5
What to test:	1. At the same time we will test these inputs to the state machine in Normal state (The status register status_reg[3:0] reads 4'h1): ● Maroon = 1'b0 ● Gold = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to stay in Normal state. The status register status_reg[3:0] should read 4'h1. 2. At the same time we will test these inputs to the state machine in Normal state (The status register status_reg[3:0] reads 4'h1): ● Maroon = 1'b0 ● Gold = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation)

	Result: This should cause the State Machine to stay in Normal state. The status register status_reg[3:0] should read 4'h1. 3. At the same time we will test these inputs to the state machine in Normal state (The status register status_reg[3:0] reads 4'h1): ● Maroon = 1'b1 ● Gold = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to stay in Normal state. The status register status_reg[3:0] should read 4'h1. 4. At the same time we will test these inputs to the state machine in Normal state (The status register status_reg[3:0] reads 4'h1): ● Maroon = 1'b1 ● Gold = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to stay in Normal state. The status register status_reg[3:0] should read 4'h1. 5. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b0 ● Gold = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to transition to Error state. The status register status_reg[3:0] should read 4'h2. 6. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b0 ● Gold = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to transition to Error state. The status register status_reg[3:0] should read 4'h2. 7. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b1 ● Gold = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to transition to Error state. The status register status_reg[3:0] should read 4'h2. 8. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b1

	● Gold = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the State Machine to transition to Error state. The status register status_reg[3:0] should read 4'h2. 9. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b0 ● Gold = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** This should cause the State Machine to transition to Export Violation state.The status register status_reg[3:0] should read 4'h8. 10. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b0 ● Gold = 1'b1 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1 (Set export violation command by applying a bad command to the command register with export_disable asserted) **Result:** This should cause the State Machine to transition to Export Violation state.The status register status_reg[3:0] should read 4'h8. 11. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b1 ● Gold = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1 (Set export violation command by applying a bad command to the command register with export_disable asserted) **Result:** This should cause the State Machine to transition to Export Violation state.The status register status_reg[3:0] should read 4'h8. 12. At the same time we will test these inputs to the state machine in Normal state: ● Maroon = 1'b1 ● Gold = 1'b1 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1 (Set export violation command by applying a bad command to the command register with export_disable asserted) **Result:** This should cause the State Machine to transition to Export Violation state.The status register status_reg[3:0] should read 4'h8. 13. In the normal state, on asserting rst_b pin, the state machine should transition to the reset state, irrespective of the other inputs. The status register status_reg[3:0] should now read 4'h0. **Note**: Other cases (cases of bad_cmd = 1'b0 and exp_vio = 1'b1) are not part of the test since due to the specifications, we can only have either a no command (bad_cmd = 1'b0 and exp_vio = 1'b0), a bad command (bad_cmd = 0), or an export violation

	command (bad_cmd = 1 and exp_vio = 1). To perform a bad command that will trigger an error state or an export_violation state if export_disable is asserted, we will write 16h '800A to the command register. When testing, we are only looking at one cause of getting into the next state, and code coverage will cover the others.

Test:	State Machine in Error State
Section:	4.0, 5.3, 6.4, 5.2, 6.5
What to test:	1. At the same time we will test these inputs to the state machine in error state (The status register status_reg[3:0] reads 4'h2): ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 2. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 3. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to change to the normal state. The status register should status_reg[3:0] now read 4'h1. 4. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This will cause the state machine to remain in the error state. The status register should status_reg[3:0] read 4'h2. 5. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted)

	● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 6. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 7. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the error state. This will cause the state machine to change to the normal state. The status register status_reg[3:0] should now read 4'h1. 8. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state.The status register status_reg[3:0] should read 4'h2. 9. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 10. At the same time we will test these inputs to the state machine in error state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state. The status register status_reg[3:0] should read 4'h2. 11. At the same time we will test these inputs to the state machine in error state:

	● Gold =1’b0 ● Maroon = 1’b1 ● bad_cmd = 1’b1 (apply bad command) ● exp_vio = 1’b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the error state. This will cause the state machine to change to the normal state. The status register status_reg[3:0] should now read 4’h1. 12. At the same time we will test these inputs to the state machine in error state: ● Gold =1’b1 ● Maroon = 1’b1 ● bad_cmd = 1’b1 (apply bad command) ● exp_vio = 1’b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the error state. This will cause the state machine to remain in the error state. The status register should status_reg[3:0] read 4’h2. 13. When in error state, on asserting rst_b pin, the state machine should transition to the reset state, irrespective of the other inputs. The status register status_reg[3:0] should now read 4’h0. **Note**: Other cases (cases of bad_cmd = 1’b0 and exp_vio = 1’b1) are not part of the test since due to the specifications, we can only have either a no command (bad_cmd = 1’b0 and exp_vio = 1’b0), a bad command (bad_cmd = 0), or an export violation command (bad_cmd = 1 and exp_vio = 1). To perform a bad command that will trigger an error state or an export_violation state if export_disable is asserted, we will write 16h '800A to the command register. When testing, we are only looking at one cause of getting into the next state, and code coverage will cover the others. Whenever the state transitions into an error state we will get INT1.

Test:	State Machine in Export Violation State
Section:	4.0, 5.3, 6.3, 5.2, 6.5
What to test:	1. At the same time we will test these inputs to the state machine in export violation state (The status register status_reg[3:0] reads 4’h8): ● Gold =1’b0 ● Maroon = 1’b0 ● bad_cmd = 1’b0 (Set no commands,No error) ● exp_vio = 1’b0 (export_disable not asserted, no export violation) **Result:** This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4’h8. 2. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1’b1 ● Maroon = 1’b0

	● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 3. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 4. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b0 (Set no commands,No error) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 5. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 6. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 7. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register

	command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 8. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (Set error command applying a bad command to the register command with export_disable not asserted) ● exp_vio = 1'b0 (export_disable not asserted, no export violation) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 9. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b0 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 10. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b1 ● Maroon = 1'b0 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 11. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b0 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8.

	12. At the same time we will test these inputs to the state machine in export violation state: ● Gold =1'b1 ● Maroon = 1'b1 ● bad_cmd = 1'b1 (apply bad command) ● exp_vio = 1'b1(Set export violation command applying a bad command to the register command with export_disable asserted) **Result:** Commands are disabled in the export violation state. This should cause the state machine to remain in the export violation state. The status register status_reg[3:0] should read 4'h8. 13. When in the export violation state, on asserting rst_b pin, the state machine should transition to the reset state, irrespective of the other inputs. The status register status_reg[3:0] should now read 4'h0. **Note**: Other cases (cases of bad_cmd = 1'b0 and exp_vio = 1'b1) are not part of the test since due to the specifications, we can only have either a no command (bad_cmd = 1'b0 and exp_vio = 1'b0), a bad command (bad_cmd = 0), or an export violation command (bad_cmd = 1 and exp_vio = 1). To perform a bad command that will trigger an error state or an export_violation state if export_disable is asserted, we will write 16h '800A to the command register. When testing, we are only looking at one cause of getting into the next state, and code coverage will cover the others.Whenever the state transitions into an export violation state we will get INT2.