Module: SV for Verification
Section: Typedef/Enum **Task:** Mailboxes

Task - Typedef/Enum
Typedef-Enum

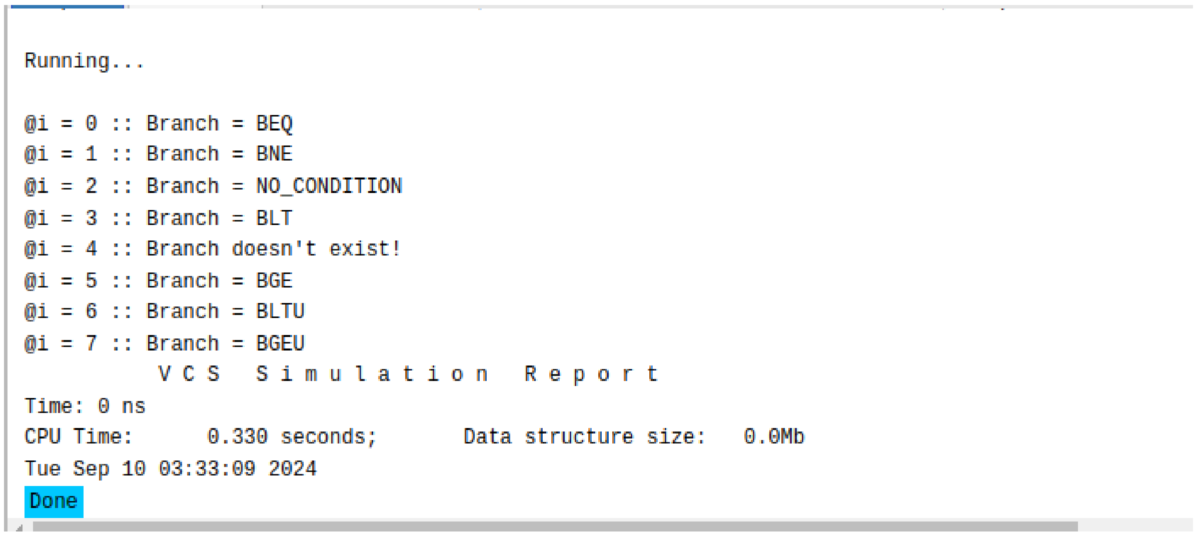
➢ Code:
```
// Author: Noman Rafiq
// Dated: Sep 10, 2024

typedef bit[15:0] data_bus_t; // custom 16-bit data-type of 'data_bus_t' type
  typedef enum bit[2:0] {BEQ, BNE, NO_CONDITION, BLT, BGE=3'b101, BLTU, BGEU}
br_cond_e;     // custom enumerated variable for conditional branch types

module tb;
  data_bus_t data;     // 16-bit variable data of 'data_bus_t' type
  br_cond_e branch;    // branch variable of enum 'br_cond_e' type

  initial begin
    branch = 0;
    $display("\nRunning...\n");
    for (int i = 0; i < 8; i++) begin
    case(branch)
      3'b000 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b001 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b010 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b011 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b101 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b110 : $display("@i = %0d :: Branch = %s", i, branch.name());
      3'b111 : $display("@i = %0d :: Branch = %s", i, branch.name());
      default: $display("@i = %0d :: Branch doesn't exist!", i);
      endcase
      branch=branch + 1;
    end
  end
endmodule
```

➢ Difference b/w bit[15:0] and data_bus_t:
The difference between **bit [15:0] data** and **data_bus_t data** is largely syntactical.

1. **bit [15:0] data:**
   - This explicitly declares a 16-bit bit vector.
2. **data_bus_t data:**
   - This uses a **typedef** alias for **bit [15:0]**, which is essentially the same type.
   - **data_bus_t** is a more abstract way to represent the data. Using the typedef improves code readability and maintainability, especially if the bit-width or type of the data bus needs to change later.

In practice, both declarations refer to a 16-bit **bit** vector, but **data_bus_t** provides more flexibility by allowing you to change the underlying type in one place if necessary.

Output:



```
Running...

@i = 0 :: Branch = BEQ
@i = 1 :: Branch = BNE
@i = 2 :: Branch = NO_CONDITION
@i = 3 :: Branch = BLT
@i = 4 :: Branch doesn't exist!
@i = 5 :: Branch = BGE
@i = 6 :: Branch = BLTU
@i = 7 :: Branch = BGEU
           V C S   S i m u l a t i o n   R e p o r t
Time: 0 ns
CPU Time:      0.330 seconds;       Data structure size:   0.0Mb
Tue Sep 10 03:33:09 2024
Done
```