Flash FSM State Transition Diagram

Note: This FSM diagram is for the Bonus part of the lab where audio samples are played at 44kHz & 8-bit output

reset

IDLE

```
if (restart) begin //if R key pressed, go to start of song
    current_address <= 23'h0;
    restart_start_protocol <= 1'b1;
end
```

OUTPUT4

```
if (current_direction == `BACKWARD) begin
    audio_out <= data_out[7:0]; //Output lower 16 bits of sample
end else begin
    audio_out <= data_out[31:24]; //Output upper 16 bits of sample
end

if (current_direction == `FORWARD) begin
    if (current_address < `MAX_ADDRESS) begin //Total number of samples is 23'h7FFFF
        current_address <= current_address + 1;
    end else begin
        current_address <= 23'h0; //Wrap around to start
    end
end else begin
    if (current_address > 0) begin
        current_address <= current_address - 1;
    end else begin
        current_address <= `MAX_ADDRESS; //Wrap around to end
    end
end
```

(play-enable == `PLAY_MUSIC) && clk_async

(play-enable == `PLAY_MUSIC) && clk_async

current_direction <= direction
flash-mem-read <= 1'b1
flash_mem_address <= current_address

OUTPUT3

```
if (current_direction == `BACKWARD) begin
    audio_out <= data_out[15:8]; //Output middle 16 bits of sample
end
else begin
    audio_out <= data_out[23:16]; //Output middle 16 bits of sample
end
```

ADDRESS

(play-enable == `PLAY_MUSIC) && clk_async

!flash_mem_waitrequest
flash_mem_read <= 1'b0

OUTPUT2

```
if (current_direction == `BACKWARD) begin
    audio_out <= data_out[23:16]; //Output middle 16 bits of sample
end
else begin
    audio_out <= data_out[15:8]; //Output middle 16 bits of sample
end
```

WAIT_VALID

OUTPUT1

```
if (current_direction == `BACKWARD) begin
    audio_out <= data_out[31:24]; //Output upper 16 bits of sample
end
else begin
    audio_out <= data_out[7:0]; //Output lower 16 bits of sample
end
```

(play-enable == `PLAY_MUSIC) && clk_async

flash-mem-readdata valid
data_out <= flash-mem-readdata