



9. Aufgabenblatt

16.06.2010

Aufgabe 1: Schnelles Potenzieren

Entwerfen Sie analog zur in der Vorlesung gezeigten Vorgehensweise zum systematischen Entwurf eine Schaltung für ein Verfahren zur schnellen Potenzierung ganzer Zahlen mit ganzzahligen Exponenten nach folgendem Pseudo-Code:

- Eingaben Basis *base*, Exponent *exp* vorzeichenlos zu je 8 Bit
- Ausgabe Potenz *pow* ist 512 Bit breit
- Signal *start*=1 startet Rechnung
- Signal *done*=1 zeigt Abschluss der Rechnung an

```
pow2n[511:0] := base[7:0];  
pow[511:0] := 1;  
done := 0;
```

```
WHILE (exp != 0) DO BEGIN  
  if (exp[0] == 1)  
    pow := pow * pow2n;  
    pow2n := pow2n * pow2n;  
    exp := exp >> 1;  
END
```

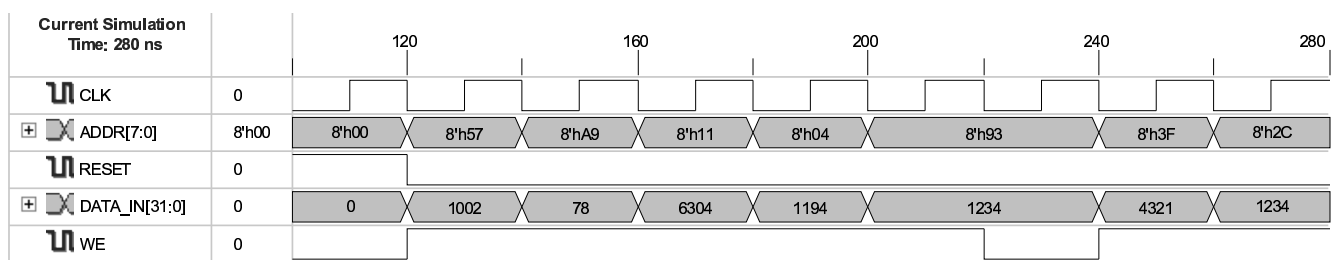
```
done := 1;
```

Implementieren Sie Steuerwerk und Datenpfad als getrennte Verilog-Module. Finden Sie dazu geeignete Steuer- und Statussignale. Testen Sie die Funktion der Schaltung mit einer Testbench. Geben Sie den Zustandsgraphen des Steuerwerks und das Diagramm des Datenpfades an.

Aufgabe 2: Adressen und Inhalte

Gegeben sind eine Waveform und ein Verilog HDL Modul.

Waveform:



Verilog-Modul:

```
module memory_mapped_regs(CLK, RESET, ADDR, DATA_IN, WE, DATA_OUT);
    input CLK;
    input RESET;
    input [7:0] ADDR;
    input [31:0] DATA_IN;
    input WE;
    output [31:0] DATA_OUT;

    reg [31:0] A, B, C, D, E, F, G;

    assign DATA_OUT = ADDR[6] ? D : ADDR[5:4] == 2'b00 ? A :
        ADDR[5:4] == 2'b01 ? B : ADDR == 42 ? E :
        ADDR[7] ? C : ADDR[3:2] == 2 ? G : F;

    always @(posedge CLK or posedge RESET) begin
        if (RESET) begin
            A <= 1; B <= 2; C <= 3; D <= 4;
            E <= 5; F <= 6; G <= 7;
        end
        else
            case ({WE, ADDR[7], ADDR[4:2]})
                5'h17: A <= DATA_IN;
                5'h1F: B <= DATA_IN;
                5'h13: C <= DATA_IN;
                5'h14: D <= DATA_IN;
                5'h1A: E <= DATA_IN;
                5'h15: F <= DATA_IN;
                5'h1C: G <= DATA_IN;
                default: begin
                    G <= 42;
                    A <= DATA_IN;
                end
            endcase
        end
    end

endmodule
```

Die in der Waveform dargestellten Signale werden als Stimulus in das Verilog HDL Modul eingegeben. Welche Werte enthalten danach die Register A bis G und auf welchen Adressen können sie jeweils ausgelesen werden?