

# Laborübung 3

## Latches, Flipflops, Counter

### Teil 1

Abbildung 1 zeigt den Schaltkreis eines gated D-Latches. In Listing 1 wird exemplarisch ein Stück VHDL-Code vorgestellt, der den abgebildeten Schaltkreis in Form von Logikausdrücken realisiert.

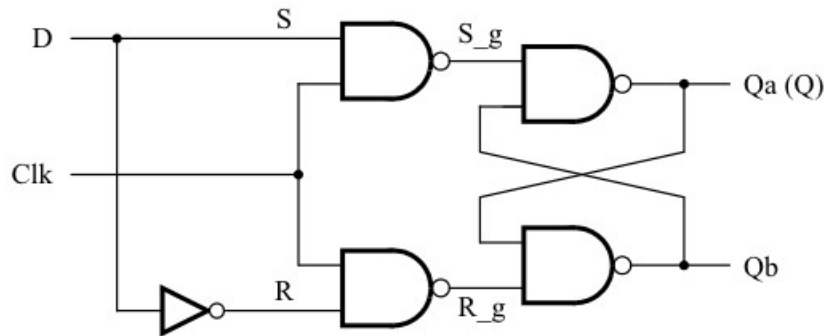


Abbildung 1: Schaltkreis eines Gated D-Latch

```
-- A gated RS latch described the hard way
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY part1 IS
    PORT (
        Clk, D : IN STD_LOGIC;
        Q      : OUT STD_LOGIC
    );
END part1;

ARCHITECTURE Structural OF part1 IS
    SIGNAL R, S, R_g, S_g, Qa, Qb : STD_LOGIC ;

BEGIN
    S <= D;
    R <= NOT D;
    R_g <= R AND Clk;
    S_g <= S AND Clk;
    Qa <= NOT (R_g OR Qb);
    Qb <= NOT (S_g OR Qa);
    Q <= Qa;
END Structural;
```

Listing 1: Gated D-Latch (kombinatorisch)

Die gleiche Funktionalität lässt sich mit Hilfe eines „PROCESS“ realisieren, der sensitiv auf die Signale „D“ und „Clk“ ist. Der entsprechende VHDL-Code ist in Listing 2 wiedergegeben.

```

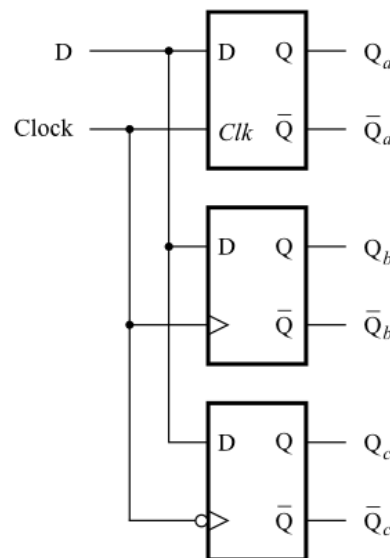
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY latch IS
  PORT (
    D, Clk : IN STD_LOGIC ;
    Q : OUT STD_LOGIC
  );
END latch ;
ARCHITECTURE Behavior OF latch IS
BEGIN
  PROCESS ( D, Clk )
  BEGIN
    IF Clk = '1' THEN
      Q <= D ;
    END IF ;
  END PROCESS ;
END Behavior ;

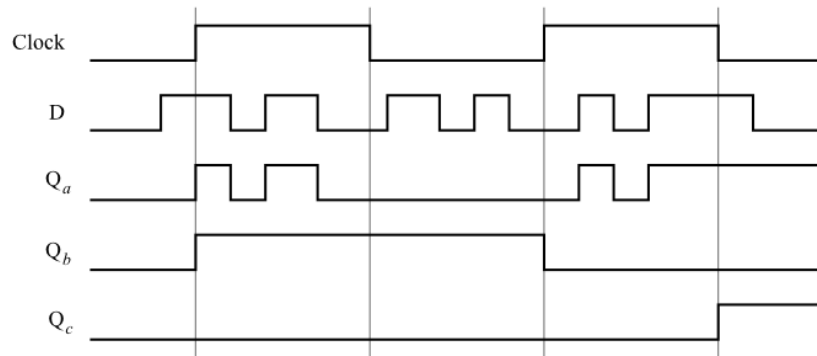
```

*Listing 2: Gated D-Latch (Prozess)*

Abbildung 2 zeigt nun einen Schaltkreis mit drei verschiedenen Speicherelementen. Neben einem gated D-Latch sind auch ein D-Flipflop mit positiver Taktflanke sowie eines mit negativer Taktflanke vorhanden. Aus Abb. 3 können die Signale an den Ausgängen abhängig von den Eingangssignalen „Clock“ und „D“ entnommen werden.



*Abbildung 2: Schaltkreis*



*Abbildung 3: Timing-Diagramm*

Implementieren Sie den abgebildeten Schaltkreis.

1. Machen Sie sich mit dem Konstrukt „PROCESS“ vertraut. Beachten Sie besonders das Schlüsselwort „event“ im Zusammenhang mit (Takt-)Signalen.
2. Schreiben sie eine VHDL-Datei, welche die drei verschiedenen Speicherelemente instanziiert und implementieren Sie die Elemente als Komponenten. Verwenden Sie dazu wie in Listing 2 einen „PROCESS“.
3. Erstellen Sie eine Vector Waveform File (.vwf) um die Inputs und Outputs zu spezifizieren. Übernehmen sie die Inputs aus Abb. 3. Simulieren sie den Schaltkreis und überprüfen sie das unterschiedliche Verhalten der drei Speicherelemente.

## Teil 2

Der Schaltkreis in Abb. 4 zeigt einen synchronen 4-Bit-Zähler, der aus vier T-Flipflops aufgebaut ist. Ist das Enable-Signal gesetzt, so erhöht er seinen Zählerstand bei jeder positiven Taktflanke. Mit dem Reset-Signal lässt sich der Zähler wieder auf Null zurücksetzen.

Ein T-Flipflop wechselt seinen Zustand („toggle“) bei jeder positiven Taktflanke, solange an „T“ ein High-Pegel anliegt, ansonsten wird der gespeicherte Zustand gehalten.

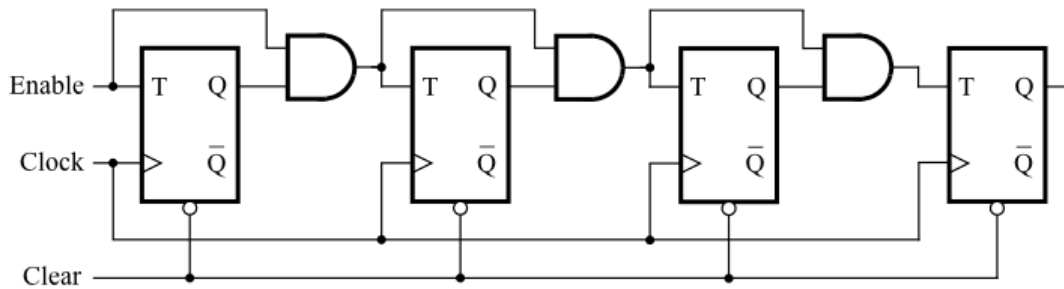


Abbildung 4: 4-Bit-Zähler

Implementieren Sie auf diese Weise einen 16-Bit-Zähler.

1. Erstellen sie ein T-Flipflop (als Komponente). Falls gewünscht, können Sie dafür auf ein bereits implementiertes Speicherelement zurückgreifen.
2. Schreiben sie eine VHDL-Datei, welche den 16-Bit-Zähler mit Hilfe der in Abb. 4 gezeigten Struktur umsetzt. Simulieren sie den Schaltkreis.
3. Erweitern Sie ihren Code so, dass der Taster KEY0 als Takteingang und die Schalter SW1 und SW0 als Enable und Reset dienen. Benutzen Sie die 7-Segment-Anzeigen HEX3-0, um hexadezimal den Zählerstand auszugeben.

### Teil 3

Vereinfachen Sie ihren Code so, dass die Spezifikation des Zählers auf dem VHDL-Ausdruck

$$Q \leq Q + 1;$$

basiert.

Erstellen sie wieder einen 16-Bit-Zähler und überprüfen Sie seine Funktion.

### Teil 4

Entwerfen und implementieren Sie einen Schaltkreis, welcher der Reihe nach die Ziffern 0 bis 9 auf der 7-Segment-Anzeige HEX0 ausgibt. Dabei soll jede Ziffer etwa eine Sekunde lang angezeigt werden. Benutzen Sie einen Zähler, um die Sekunden-Intervalle zu erzielen.

Der Zähler soll dabei vom auf dem Board vorhandenen 50 MHz Takt gespeist werden. Dieser ist an einen Pin mit der Bezeichnung CLOCK\_50 angebunden, der wie einer der bekannten Schalter-Eingänge verwendet werden kann.

**Leiten sie keine anderen Taktsignale ab – stellen Sie sicher, dass alle Flipflops in der Schaltung direkt am 50 MHz Takt hängen.**

## Teil 5

Erstellen sie eine Schaltung, die das Wort „HELLO“ in Form eines Tickers durch die 7-Segment-Anzeigen scrollt. Das Verschiebe-Intervall soll wieder ca. eine Sekunde betragen. Die anzuzeigenden Muster sind noch einmal in Abb. 5 wiedergegeben.

Clock cycle	Displayed pattern
0	H E L L O
1	H E L L O
2	H E L L O
3	H E L L O
4	E L L O       H
5	L L O         H E
6	L O           H E L
7	O             H E L L
8	H E L L O
...	and so on

*Abbildung 5: Scrolling des Wortes "HELLO"*

## Teil 6

Implementieren Sie einen dreistelligen BCD-Zähler. Geben Sie den Inhalt auf den Anzeigen HEX2-0 aus. Leiten Sie aus dem vorhandenen 50 MHz Takt ein Kontrollsignal ab, um den Zähler automatisch jede Sekunde zu erhöhen. Benutzen Sie den Taster KEY0, um den Zähler auf Null zurückzusetzen.