



Designnotat

Tittel: Digital Terning

Forfattere: Eirik M. Silnes

Versjon: 1.1

Dato: 7. mai 2023

Innhold

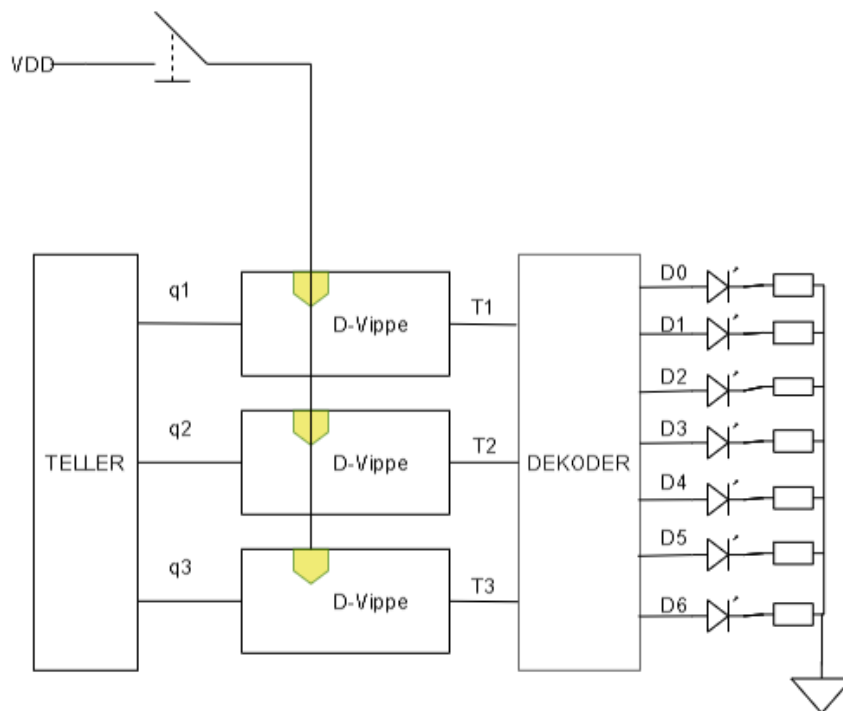
| | | |
|----------|----------------------------|----------|
| 1 | Problembeskrivelse | 2 |
| 2 | Prinsipiell løsning | 2 |
| 3 | Realisering og test | 3 |
| 4 | Konklusjon | 7 |
| 5 | Takk | 7 |

1 Problembeskrivelse

I dette designnotatet så skal det designes en digital terning ved hjelp av en FPGA av typen Lattice ICE40. Terningen skal ha lik sansynelighet for hvert mulig antall øyne og realiseres med LEDs som lyser opp i mønster som øynene på en terning. Det skal også undersøkes effektforbruket av lysdiodene med tilhørende motstander for alle utfall og med forskjellige farger dioder. Det skal også beregnes et froventet effektforbruk.

2 Prinsipiell løsning

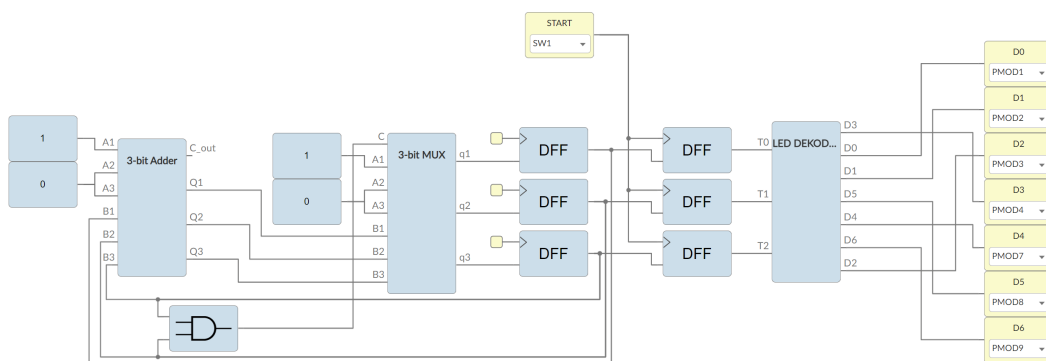
For å designe en terning må man ha et tilfeldig tall man kan dekode til øyne på en terning. Som sett i 1 så vil terningen i dette designnotatet bestå av en teller som teller fra 1 til 6 i klokkefrekvensen til FPGAen, den vil kun stoppe ved et signal som kommer fra en knapp brukeren kan trykke på. Dette vil ikke gi et helt tilfeldig tall, men hastigheten på systemet er så høy at man kan annta at systemet vil stoppe på en tilfeldig verdi. Som vist i 1 Så er det lagt inn register etter telleren for å lagre terningkastets verdi. For så ha en dekode som dekode signalet til LED matrisen.



Figur 1: Prinsipiell overordnet løsning

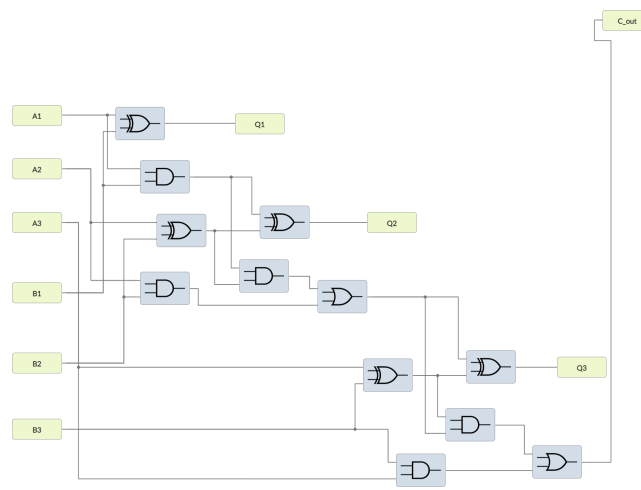
3 Realisering og test

Figur 2 viser den realiserte løsningen på et overordnet nivå. Telleren som er nevnt i den prinsipielle løsningen er realisert med en adderer som tar inn binære 3-bits tall fra d-vippene i slutten av kretsen og legger det sammen med et fast binært tall. Deretter går signalet inn i en 3-bit mux med to innganger, som vil resete kretsen til 001 dersom signalet blir 110. D-vippene lagrer dataen og sender det nåværende tallet tilbake til addereren og starter kretsen på nytt. Med denne oppkoblingen er det klokken til d-vippen som bestemmer hvor raskt telleren teller. Deretter er det et register med d-vipper som lagrer verdien til telleren ved en stigende flanke og på den måten lagre et tilfeldig tall fra telleren og sende det videre til dekodeeren. Dekoderen tar inn et 3-bit signal og bestemmer hvilke LEDs i matrisen som skal lyse.



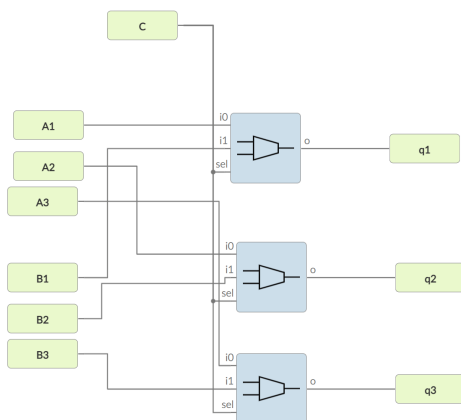
Figur 2: Overblikk over den realiserte kretsen

Figur 3 viser realiseringen av en 3-bits fulladderer. Den er realiser med tre XOR porter som legger sammen ingang A og B dersom de er ulike, og sender det til utgang q. Deretter brukes en kombinasjon av AND og OR porter for å sende bæretallet videre.



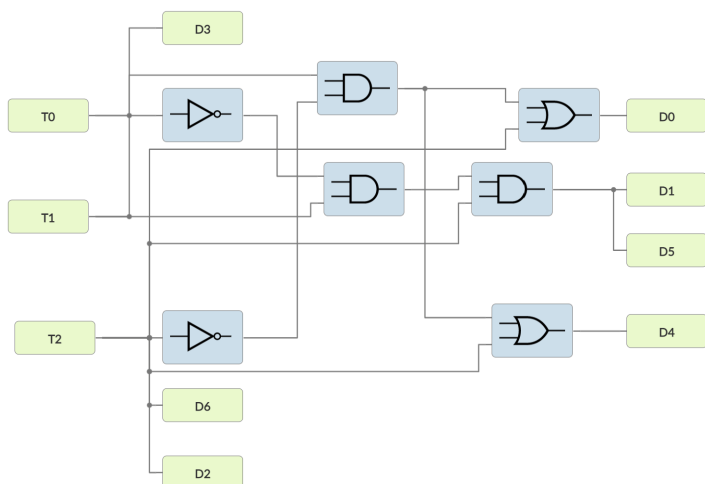
Figur 3: Fulladderer

Figur 4 viser realiseringen av en 3-bits mux, hvor man utnytter FPGAen sine interne MUXer.

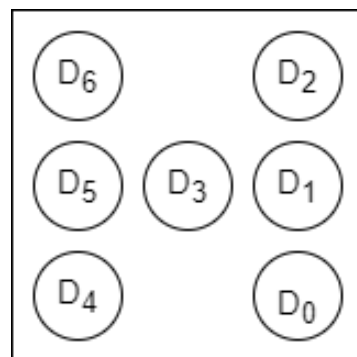


Figur 4: 3-bits mux

For at LED lysene skal lyse i et mønster som tilsvarende terningens øyne, må vi ha en dekode som kan dekode terningkastet til et mønster. I figur 5 viser vi realiseringen av en dekode og i tabell 1 vises sannhetstabellen for dekode og i figur 6 viser LED matrisen som er koblet opp til dekode med navnene utgangen av dekode i riktig mønster.



Figur 5



Figur 6

Telleren er designet på en slik måte at den aldri teller til mer en 6 (110) så man kan se bort ifra når ingagnssignalet er 111. Dette gjør at de boolske uttrykkene som er brukt til å designe dekode i figur 5 blir vesentlig mye enklere. De boolske uttrykkene som er brukt for å designe dekode er vist i likningene 1 - 5.

$$D_0 = \overline{T_0}T_1T_2 \quad (1)$$

$$D_1 = D_5 = T_0T_1T_2 \quad (2)$$

Tabell 1: Sannhetstabell for dekoderen

| T_2 | T_1 | T_0 | D_6 | D_5 | D_4 | D_3 | D_2 | D_1 | D_0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |

$$D_2 = D_6 = T_2 \quad (3)$$

$$D_3 = T_0 \quad (4)$$

$$D_4 = \overline{T_2}T_1 + T_2 \quad (5)$$

For å måle effektforbruket til terningen så måler vi spenningen over hver av de 7 LEDene og regner strømmen gjennom motstandene for å beregne effektforbruket. I tabell 2 viser vi målingene av spenningen over hver av LEDene.

Tabell 2: Målinger av spenningen over hver av LEDene

| Terningkast | V_{LED0} | V_{LED1} | V_{LED2} | V_{LED3} | V_{LED4} | V_{LED5} | V_{LED6} |
|-------------|------------|------------|------------|------------|------------|------------|------------|
| 1 | - | - | - | 3.234 | - | - | - |
| 2 | - | - | 3.123 | - | - | - | 3.123 |
| 3 | - | - | 3.123 | 3.123 | - | - | 3.123 |
| 4 | 3.132 | - | 3.003 | - | 3.003 | - | 3.132 |
| 5 | 3.132 | - | 3.003 | 3.003 | 3.003 | - | 3.132 |
| 6 | 3.132 | 3.403 | 3.003 | - | 3.003 | 3.004 | 3.132 |

Vi regner ut strømmen gjennom motstandene ved å bruke formelen 6.

$$I = \frac{V_{RD}}{R} \quad (6)$$

Der V er spenningen over motstanden og R er motstandens verdi. Vi regner ut strømmen gjennom hver av motstandene og regner deretter ut effektforbruket til hver av LEDene ved å bruke formelen 7.

$$P_e = V * I \quad (7)$$

For å regne ut effektforbruket til hver av lysdiodene så må vi vite hvor ofte den lyser, i tillegg til strømmen gjennom den. Vi regner ut hvor ofte hver av LEDene lyser ved å se på sannsynligheten til hver verdi terningen kan få og om LEDen lyser ved den verdien. Vi regner ut sannsynligheten ved å bruke formelen 8.

$$P = \frac{\text{Antall gunstige}}{\text{Antall mulige}} \quad (8)$$

I tabell 3 kan vi lese ut spenningen over hver av LEDene og hvor mye effekt hver diode bruker.

Tabell 3: Effektforkbruket til hver av LEDene

| Farge | V_{Diode} | V_{Total} | $I(mA)$ | $P_e(mW)$ | Antall | Plassering |
|-------|-------------|-------------|---------|-----------|--------|-------------------|
| Rød | 2.2 | 3.3 | 11.14 | 36.76 | 2 | D_0, D_4 |
| Grønn | 2.1 | 3.3 | 11.77 | 38.84 | 2 | D_6, D_2 |
| Blå | 2.7 | 3.3 | 5.89 | 19.70 | 3 | $3 D_5, D_3, D_1$ |

I tabell 4 kan vi se hvor ofte hver av LEDene lyser og hvor mye effektforkbruket til terningen blir, gitt at kastene er uavhengig hverandre og det er like stor sansynlighet for alle utfall. Så kan vi gange effektforkbruket med sansynligheten for at LEDen lyser og derfra regne ut det idielle effektforkbruket til terningen.

Tabell 4: Sansynligheten til hver av LEDene og effektforkbruk over tid

| Plassering | antall ganger den lyser | P (sansynlighet) | Farge | $P \cdot P_e$ |
|------------|-------------------------|-----------------------|-------|-----------------------------|
| D_0 | 3 | $\frac{3}{6} = 0.5$ | Rød | $0.5 \cdot 36.76 = 18.88$ |
| D_1 | 1 | $\frac{1}{6} = 0.167$ | Blå | $0.167 \cdot 19.70 = 3.29$ |
| D_2 | 5 | $\frac{5}{6} = 0.833$ | Grønn | $0.833 \cdot 38.84 = 32.35$ |
| D_3 | 3 | $\frac{3}{6} = 0.5$ | Blå | $0.5 \cdot 19.70 = 9.85$ |
| D_4 | 3 | $\frac{3}{6} = 0.5$ | Rød | $0.5 \cdot 36.76 = 18.38$ |
| D_5 | 1 | $\frac{3}{6} = 0.167$ | Blå | $0.167 \cdot 19.70 = 3.29$ |
| D_6 | 5 | $\frac{3}{6} = 0.833$ | Grønn | $0.833 \cdot 38.84 = 32.35$ |

Summen av alle effektforkbrukene blir da som vi ser i utregning 9, 118.39mW. Dette er effektforkbruket til terningen når den er i bruk over lang tid.

$$P_{total} = 18.88 + 3.29 + 32.35 + 9.85 + 18.38 + 3.29 + 32.35 = 118.39mW \quad (9)$$

Skulle terningen vært mer effekteffektiv så måtte man ha plasert de diodene som bruker minst effekt på de plassene som har størst sansynlighet for å lyse. En potensiell realisering av dette er plottet i tabell 5.

Summen av alle effektforkbrukene blir da som vi ser i utregning 10 79.38mW. Dette er effektforkbruket til terningen når den er i bruk over lang tid.

$$P_{total} = 18.38 + 6.49 + 17.40 + 9.85 + 18.38 + 6.49 + 17.40 = 79.38mW \quad (10)$$

Besparelsen blir da som vis i utregning 11 39.01mW. Dette er en besparelse på 33.2% av det idielle effektforkbruket til terningen.

$$P_{besparelse} = 118.39 - 79.38 = 39.01mW \quad (11)$$

Tabell 5: Sansyneligheten til hver av LEDene og effektforbruk over tid, optimalisert for effektivitet

| Plassering | antall ganger den lyser | P (sansynelighet) | Farge | $P \cdot P_e$ |
|------------|-------------------------|-----------------------|-------|-----------------------------|
| D_0 | 3 | $\frac{3}{6} = 0.5$ | Rød | $0.5 \cdot 36.76 = 18.38$ |
| D_1 | 1 | $\frac{1}{6} = 0.167$ | Grønn | $0.167 \cdot 38.84 = 6.49$ |
| D_2 | 5 | $\frac{5}{6} = 0.833$ | Blå | $0.833 \cdot 19.70 = 17.40$ |
| D_3 | 3 | $\frac{3}{6} = 0.5$ | Blå | $0.5 \cdot 19.70 = 9.85$ |
| D_4 | 3 | $\frac{3}{6} = 0.5$ | Rød | $0.5 \cdot 36.76 = 18.38$ |
| D_5 | 1 | $\frac{1}{6} = 0.167$ | Grønn | $0.167 \cdot 38.84 = 6.49$ |
| D_6 | 5 | $\frac{5}{6} = 0.833$ | Blå | $0.833 \cdot 19.70 = 17.40$ |

4 Konklusjon

I dette designnotatet har det blitt designet en terning ved hjelp av en FPGA av typen Lattice ICE40 og målt effektforbruket til lysdiodene med tilhørende motstander gitt forskjellige farger på diodene. Som man kan lese utifra tabell 4 og tabell 5 så var det originale oppsettet av lysdioder ikke den mest energieffektive løsningen, men i tabell 5 blir det gjennomgått en potensiell løsning som kan gi en energibesparelse på 39.01 mW som sett i utregning 11.

5 Takk

Takk til Reidar Nerheim for veiledning i dette designnotatet