



# Designnotat

Tittel: Digital Terning

Forfattere: Eirik M. Silnes

Versjon: 1.1

Dato: 8. mai 2023

## Innhold

<b>1</b>	<b>Problembeskrivelse</b>	<b>1</b>
<b>2</b>	<b>Prinsipiell løsning</b>	<b>2</b>
<b>3</b>	<b>Realisering og test</b>	<b>3</b>
3.1	Overordnet . . . . .	3
3.1.1	Adderer . . . . .	3
3.1.2	Mux . . . . .	4
3.1.3	Dekoder . . . . .	4
3.2	Effekt . . . . .	6
3.3	Optimalisering . . . . .	7
<b>4</b>	<b>Konklusjon</b>	<b>9</b>
<b>5</b>	<b>Takk</b>	<b>9</b>

## 1 Problembeskrivelse

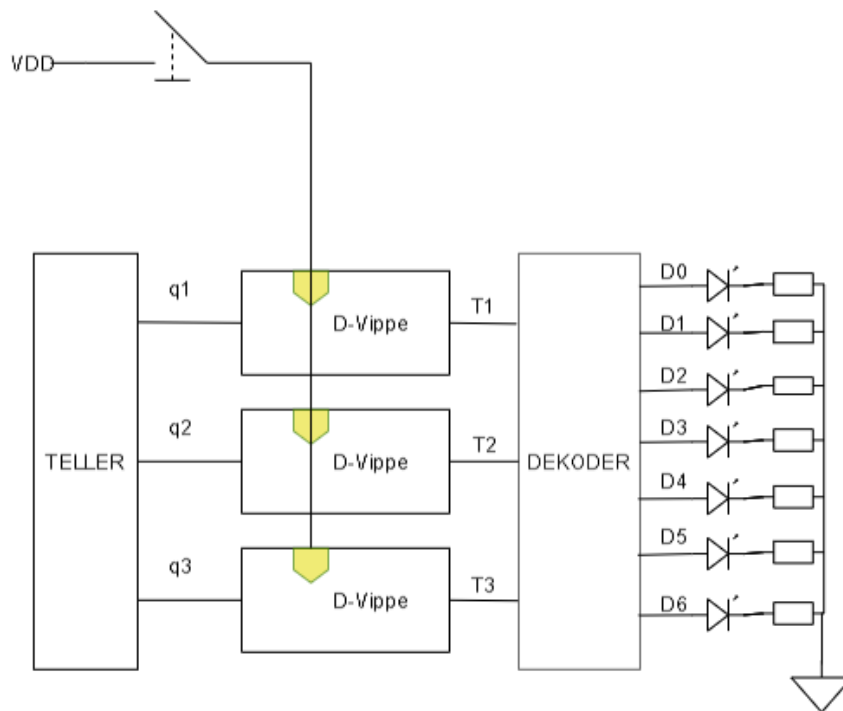
I dette designnotatet så skal det designes en digital terning ved hjelp av en FPGA av typen Lattice ICE40. Terningen skal ha lik sansynelighet for hvert mulig antall øyne og realiseres med LEDs som lyser opp i mønster som øynene på en terning. Det skal også undersøkes effektforbruket av lysdiodene med 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. Tilfeldigheten i dette terningdesignet kommer fra menneskelig faktor, det vil si at systemet vil kunne predikteres, men grunnet at et menneske interagerer med det så vil oppførselen være tilfeldig. Systemet vil bli designet som vist i figur 1 og bestå av 4 hovedelementer.

1. En teller som teller fra 1 til 6 i klokkefrekvensen til FPGAen
2. D-vipper som lagrer terningkastets verdi når knappen trykkes på.
3. En dekodeer som dekodeer signalet til LED matrisen.
4. LED matrise som viser terningkastet.

Tallene som skal vises er fra 1-6, og for å kunne vise de trenger vi et 3 bit binært tall. Systemet er derfor bygd på en 3-bits parallel buss som er synkronisert opp med klokkefrekvensen til FPGAen.

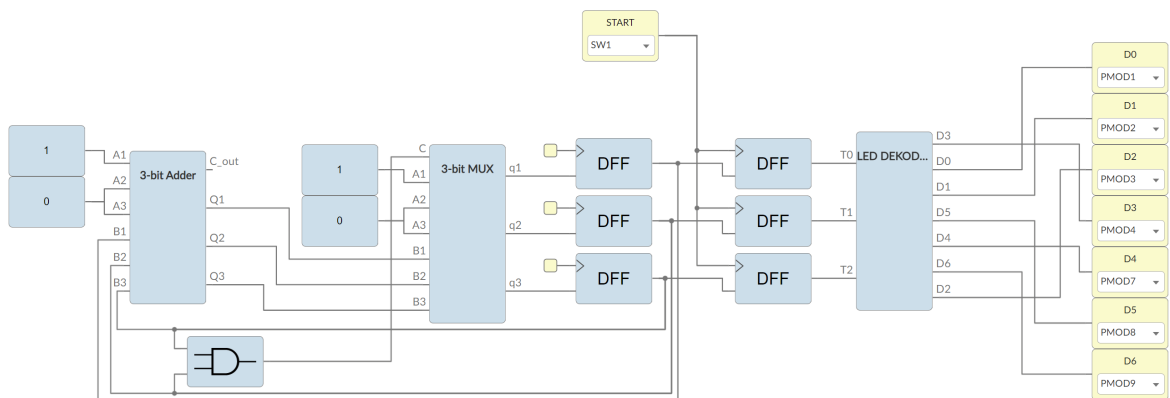


**Figur 1:** Prinsipiell overordnet løsning

### 3 Realisering og test

#### 3.1 Overordnet løsning

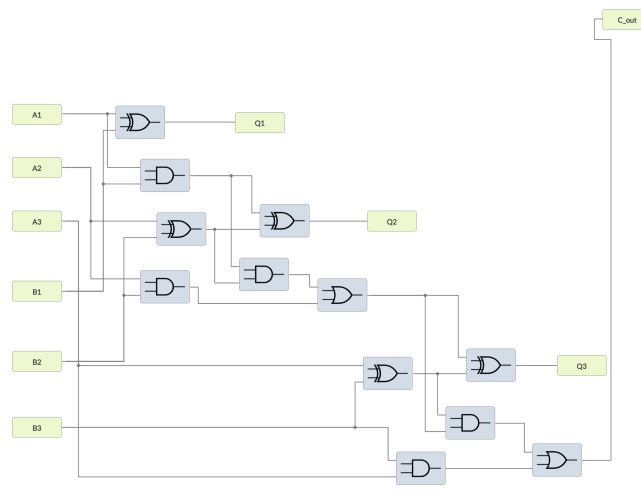
Figur 2 viser den realiserte løsningen på et overordnet nivå. Telleren som er nevnt i den prinsipielle løsningen er realisert med en adderer som tar inn to binære 3-bits tall, 001 og tallet fra d-vippene i slutten av kretsen, legger de sammen og sender signalet videre til muxen. Muxen kan ta inn to binære ord på 3-bit og veksle mellom tallet som kommer fra addereren eller en fastsatt verdi på 001. Den vil resete kretsen til 001 dersom signalet blir 110 ved hjelp av AND porten nederst i skjematikken. 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 stigenede flanke på knappen 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

##### 3.1.1 Fulladderer

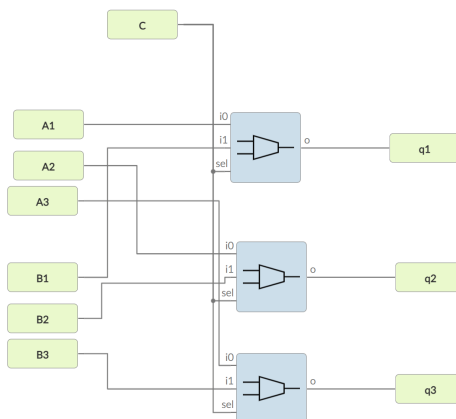
Figur 3 viser realiseringen av en 3-bits fulladderer. Den er realiser med tre XOR porter som legger sammen ingang A og B dersom de er ulike, og sender det til utgang q. Deretter brukes en kombinasjon av AND og OR porter for å sende bæretallet videre.



**Figur 3:** Fulladderer

### 3.1.2 3-bit mux

Figur 4 viser realiseringen av en 3-bits mux, hvor man utnytter FPGAen sine interne MUXer for å lage en mux som kan ta et parallelisert 3-bit signal og velge mellom to 3-bit signal.

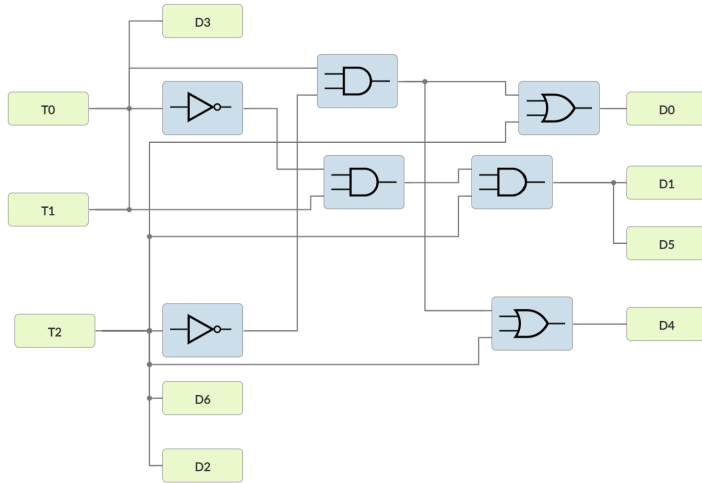


**Figur 4:** 3-bits mux

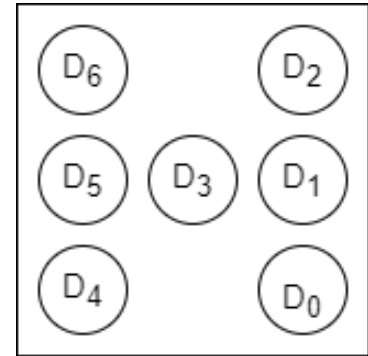
### 3.1.3 Dekoder

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 med logiske portert. 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 og når signalet er 000. Dette gjør at de boolske uttrykkene som er brukt til å designe dekode i figur 5 blir vesentlig mye enklere.

For å vite hvordan dekoderen burde designes kan man lese av sannhetstabellen 1 og regne seg fram til de boolske uttrykkene gitt i 1 - 5. Figur 6 viser LED matrisen som er koblet opp til dekoderen med navnene utgangen av dekoderen i riktig mønster.



Figur 5



Figur 6

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

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

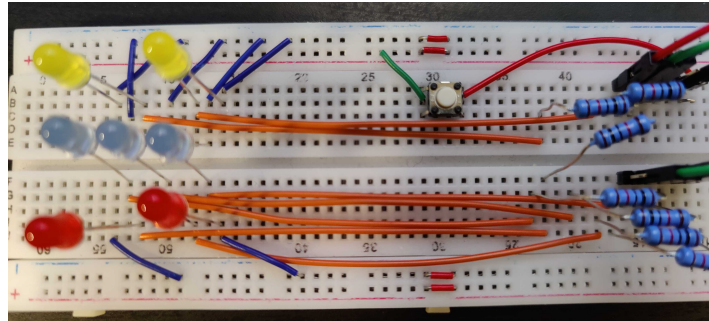
$$D_1 = D_5 = T_0T_1T_2 \quad (2)$$

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

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

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

En realisering av oppkoblingen til LED matrisen, med tilkobling til FPGAen er vist i figur 7.



**Figur 7:** Realisert oppkobling av terningen

### 3.2 Effektforbruk

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 sansynligheten til hver verdi terningen kan få og om LEDen lyser ved den verdien. Vi regner ut sansynligheten 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:** Effektforbruket 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 effektforbruket til terningen blir, gitt at kastene er uavhengig hverandre og det er like stor sansynlighet for alle utfall. Så kan vi gange effektforbruket med sansynligheten for at LEDen lyser og derfra regne ut det idielle effektforbruket til terningen.

**Tabell 4:** Sansynligheten til hver av LEDene og effektforbruk 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 effektforbrukene blir da som vi ser i utregning 9, 118.39mW. Dette er effektforbruket til terningen når den er i bruk over lang tid.

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

### 3.3 Optimalisering av effektforbruket

Skulle terningen vært mer effekt effektiv 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.

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

Plassering	antall ganger den lyser	P (sansynlighet)	Farge	$P \cdot P_e$
$D_0$	3	$\frac{3}{6} = 0.5$	Rød	$0.5 \cdot 36.76 = 18.38$
$D_1$	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	$\frac{3}{6} = 0.167$	Grønn	$0.167 \cdot 38.84 = 6.49$
$D_6$	5	$\frac{3}{6} = 0.833$	Blå	$0.833 \cdot 19.70 = 17.40$

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 ideelle effektforkbruket til terningen.

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



## 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