Designnotat

Tittel: Elektrisk terning	
Forfattere: Temmam Altaie	
Versjon: 1.0	Dato: 03.03.25

Innhold

1 Problembeskrivelse **1**

2 Prinsipiell løsning **2**
2.1 Kretsdiagram . . . 2
2.2 Kretsløsning . . . 2
2.3 Argumentasjon . . . 3
2.4 Generalitet . . . 3

3 Realisering og test **3**
3.1 Dimensjonering . . . 6
3.2 Reproduserbarhet . . . 6

4 Effektberegning for lysdiodene **6**
4.1 Gitte data . . . 7
4.2 Beregninger per lysdiode . . . 7
4.3 Effektforbruk i ulike situasjoner . . . 7

5 Konklusjon **8**
5.1 Sentrale tallverdier . . . 8
5.2 Oppsummering . . . 8

6 Takk **8**

---

1 Problembeskrivelse

I mange situasjoner er det behov for en digital representasjon av en terning, for eksempel i elektroniske spill, undervisning eller simuleringer der tilfeldige tall mellom 1 og 6 må genereres og vises på en forståelig måte. Tradisjonelle fysiske terninger har begrensninger i automatiserte systemer, da de krever manuell interaksjon og kan være utsatt for feiltolkning eller juks.

For å løse dette, må et elektronisk system kunne generere og vise tilfeldige verdier mellom 1 og 6 på en pålitelig måte. Dette innebærer utfordringer knyttet til hvordan verdiene genereres for å oppnå et jevnt og rettferdig fordelingsmønster, samt hvordan de presenteres visuelt slik at brukeren lett kan lese av resultatet. I tillegg må systemet ha en mekanisme for å starte og stoppe genereringen av nye verdier på en kontrollert måte, slik at brukeren får et klart og

endelig resultat.

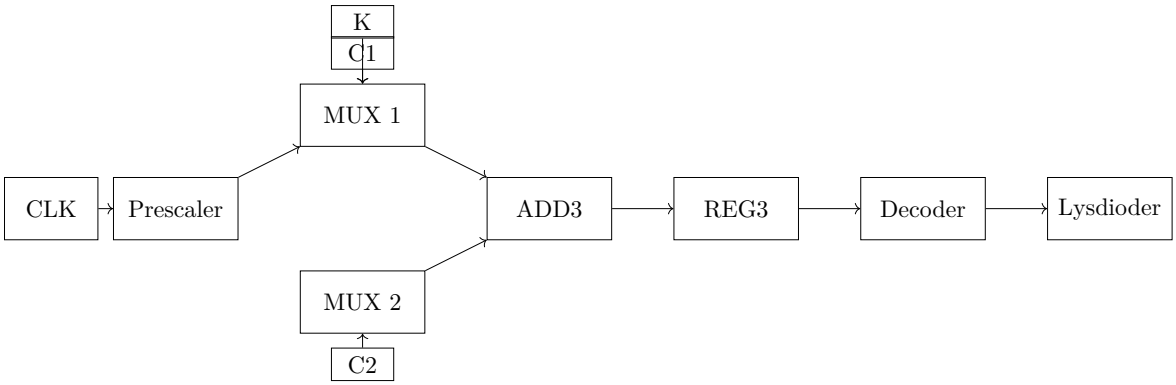
En annen viktig faktor er at løsningen må være enkel å implementere og bruke, samtidig som den skal være energieffektiv og egnet for bruk i ulike digitale systemer. Dette krever en balansert tilnærming mellom funksjonalitet, responsivitet og teknisk kompleksitet.

2 Prinsipiell løsning

For å realisere en elektronisk terning, er det nødvendig å utvikle en digital krets som genererer tallverdier mellom 1 og 6 og viser disse på et display. Løsningen må inkludere en metode for å kontrollere når nye verdier genereres, samt sikre en rettferdig fordeling av tallene.

2.1 Kretsdiagram

For å illustrere hvordan kretsen er koblet sammen, viser figur 1 en forenklet blokkskjematisk representasjon.



Figur 1: Blokkskjema av den digitale terningkretsen

2.2 Kretsløsning

Den digitale kretsen består av følgende hovedkomponenter:

- **Teller (Counter)**: En 3-bits teller som sekvensielt genererer verdier fra 000 til 101 (1 til 6 i desimal).
- **Multiplexer (MUX)**: To multiplexere som velger mellom å øke verdien eller tilbakestille telleren, avhengig av et kontrollsignal.
- **Register (REG)**: Brukes til å lagre den nåværende verdien av telleren og sikre stabil output.
- **Dekoder (Decoder)**: Konverterer 3-bits utgangen fra telleren til separate utganger til lysdiodene.

- **Styresignal (Switch)**: En kontrollmekanisme som starter og stopper verdigenereringen.

Kretsen fungerer ved at telleren inkrementerer ved hver klokkeimpuls når styresignalet er aktivt. Multiplexerne avgjør om den nye verdien skal lagres eller om systemet skal tilbakestilles. Dekoderen oversetter deretter verdien til riktig displayformat.

For å beregne effektforbruket til lysdiodene bruker vi Ohms lov og effektformelen:

$$I = \frac{V_{cc} - V_f}{R} \tag{1}$$

$$P = V_{cc} \cdot I \tag{2}$$

2.3 Argumentasjon

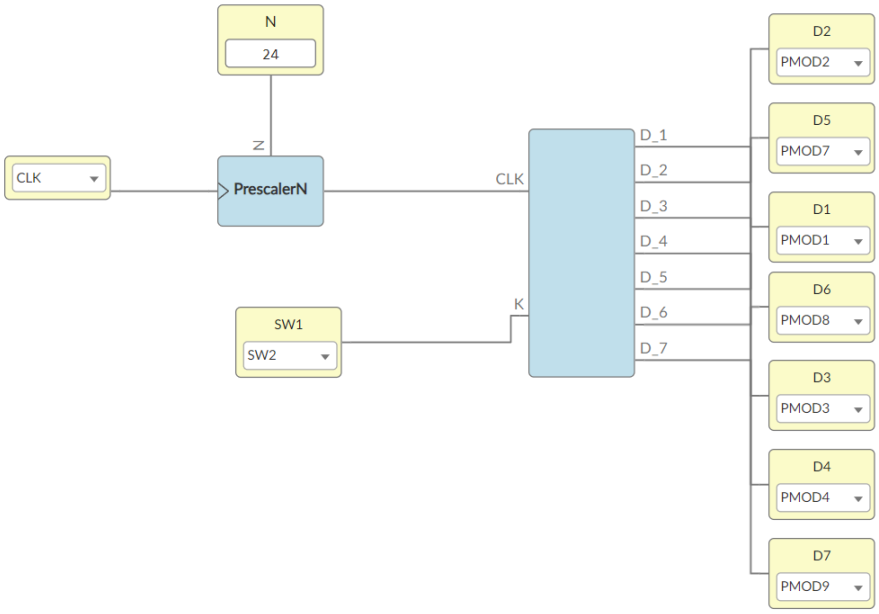
Løsningen er valgt fordi den gir en presis og digitalt kontrollerbar måte å generere tilfeldige terningverdier på. Bruk av tellere og registre sikrer sekvensiell økning og stabilitet, mens multiplexerne gjør det mulig å implementere start/stopp-mekanismen. Denne løsningen er også enkel å realisere med FPGA eller andre digitale logiske komponenter.

2.4 Generalitet

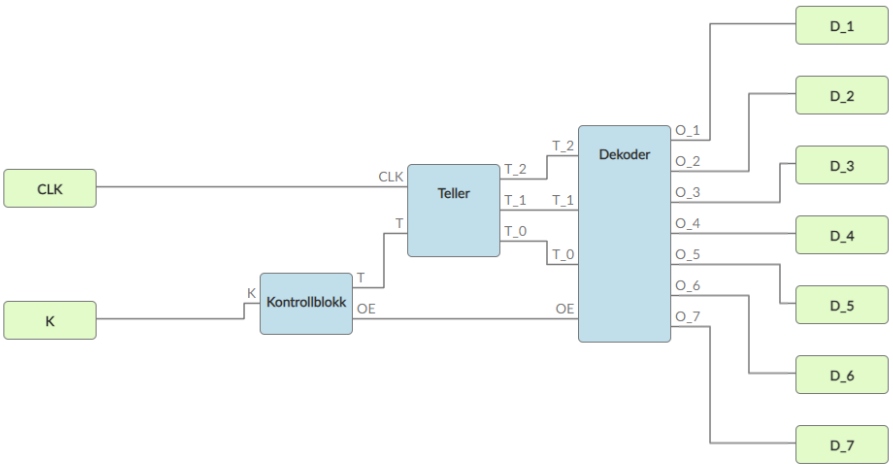
De matematiske operasjonene i kretsen, slik som telling og dekodering, er implementert uten faste numeriske verdier, noe som gjør det enkelt å utvide løsningen til for eksempel en 8-sidet terning eller andre sekvensbaserte genereringer.

3 Realisering og test

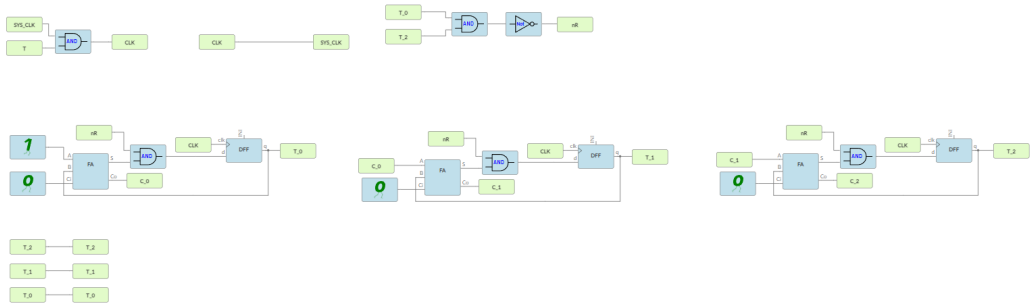
I denne seksjonen beskrives hvordan løsningen er implementert og testet i praksis. Detaljer om valg av komponentverdier, reproduserbarhet, målinger og argumentasjon for resultatene blir presentert.



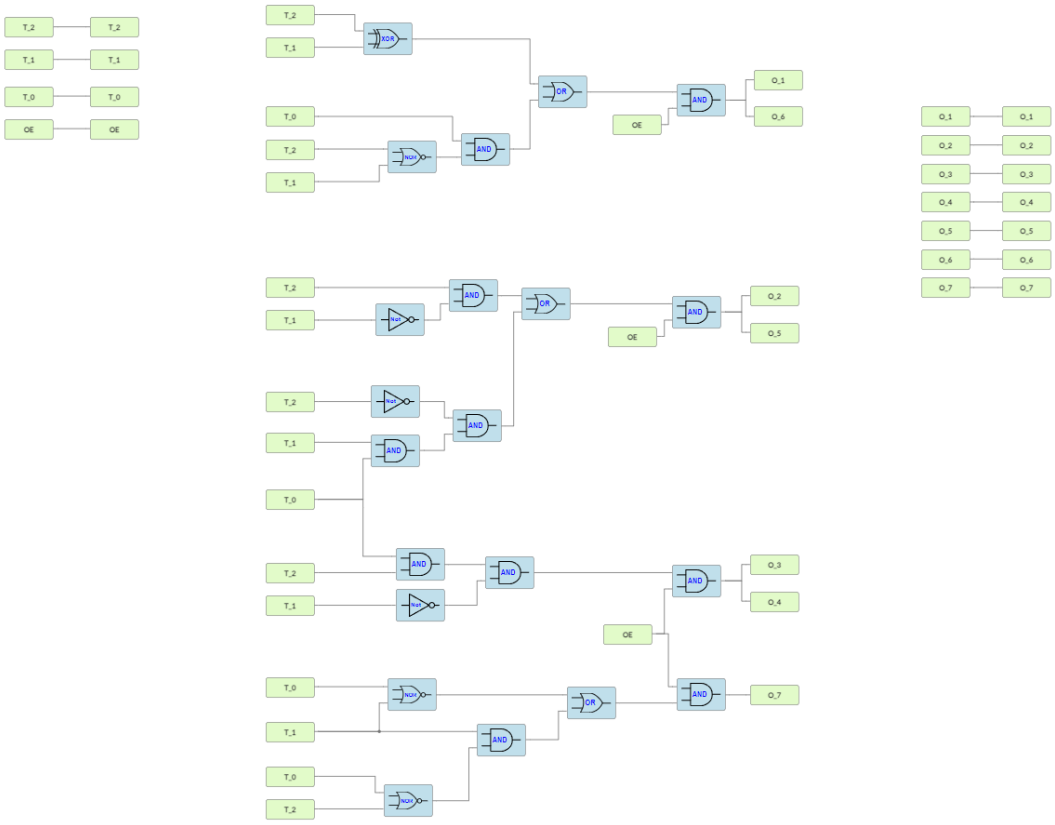
Figur 2: Designet i sin helhet



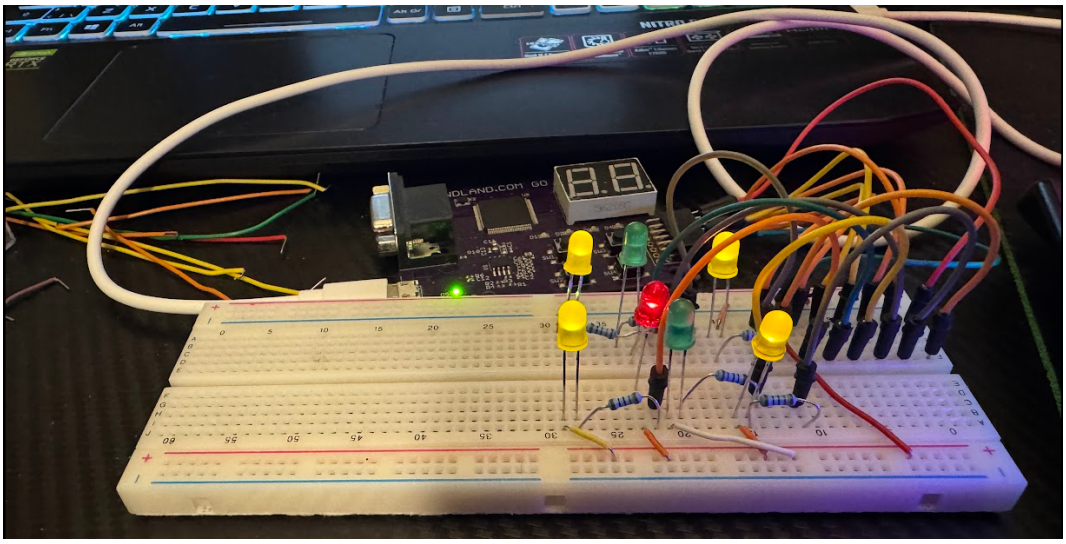
Figur 3: Design av innmaten til selve telleren



Figur 4: Telleren med D-vipper og Full-adderere



Figur 5: Design av dekoderen med utganger til Diodene (terningen)



Figur 6: Fysisk krets koblet opp ved FPGA, breadboard og Dioder

3.1 Dimensjonering

For å sikre at systemet fungerer som ønsket, er komponentverdiene nøye valgt. Prescaleren er konfigurert til å dele ned klokkesignalet for å oppnå en passende oppdateringsfrekvens for telleren. Multiplexerene styrer signalflyten i systemet basert på styresignalene C_1 og C_2. ADD3-modulen legger til én for hver klokkeperiode, og registeret REG3 holder på verdiene mellom hver klokkesyklus. Dekoderen konverterer de binære verdiene fra telleren til signaler til de syv lysdiodene. Valgene er gjort basert på prinsippene i den konseptuelle løsningen.

3.2 Reproduserbarhet

Løsningen kan gjenskapes nøyaktig ut fra dokumentasjonen i denne rapporten. Kretsen er realisert i IceStudio med en tydelig strukturert oppbygging. Figur 1 viser det endelige kretsdiagrammet, og komponentverdiene som er brukt, er dokumentert i tabell 4.1. Systemet er også testet fysisk med faktiske komponenter, og fotografier av oppkoblingen er inkludert i vedlegget.

4 Effektberegning for lysdiodene

For å beregne effektforbruket til lysdiodene bruker vi Ohms lov og effektformelen:

$$I = \frac{V_{cc} - V_f}{R} \tag{3}$$

$$P = V_{cc} \cdot I \tag{4}$$

4.1 Gitte data

- Antall lysdioder: 7 (4 gule, 2 grønne, 1 rød)
- Seriemotstand per diode: $R = 120\Omega$
- Forspenning på lysdiodene:
  - Gule: $V_f = 2.0V$
  - Grønne: $V_f = 2.2V$
  - Rød: $V_f = 1.8V$
- Forsyningsspenning: $V_{cc} = 3.3V$

4.2 Beregninger per lysdiode

Lysdiode	Strøm I (mA)	Effekt P (mW)	Totalt for n dioder (mW)
Gul ($n = 4$)	$\frac{3.3V-2.0V}{120\Omega} = 10.83$	$3.3V \cdot 10.83mA = 35.74$	$4 \times 35.74 = 142.96$
Grønn ($n = 2$)	$\frac{3.3V-2.2V}{120\Omega} = 9.17$	$3.3V \cdot 9.17mA = 30.26$	$2 \times 30.26 = 60.52$
Rød ($n = 1$)	$\frac{3.3V-1.8V}{120\Omega} = 12.5$	$3.3V \cdot 12.5mA = 41.25$	41.25
Total effektforbruk			244.73

Tabell 1: Strøm og effektberegning for lysdiodene

4.3 Effektforbruk i ulike situasjoner

Tilstand	Effektforbruk (mW)
Teoretisk maksimum (alle dioder på)	244.73
1 øye (1 rød diode)	41.25
2 øyne (2 grønne dioder)	60.52
3 øyne (2 grønne + 1 rød)	101.77
4 øyne (4 gule)	142.96
5 øyne (4 gule + 1 rød)	184.21
6 øyne (4 gule + 2 grønne)	203.48
Forventet effektforbruk ($E[P]$)	122.4

Tabell 2: Effektforbruk i ulike terningkast

Konklusjon:

- Maksimalt effektforbruk: **244.73 mW**
- Gjennomsnittlig effektforbruk: **122.4 mW**
- Minimum (når kun én diode lyser): **41.25 mW**

5 Konklusjon

Denne rapporten har omhandlet utviklingen av en digital terning ved hjelp av FPGA og IceStudio. Målet med prosjektet var å designe og implementere en fungerende elektronisk terning som kunne vise tilfeldige verdier mellom 1 og 6 via lysdioder.

Løsningen ble realisert ved å bruke en tellerstyrt krets med en preskaler, en adder, to multiplexer i serie, et register, en dekoder og en logikkblokk for styring av C1, C2 og K-signaler. Prosjektet ble testet og evaluert med tanke på funksjonalitet, ytelse og effektforbruk.

5.1 Sentrale tallverdier

De viktigste spesifikasjonene og målingene fra prosjektet er oppsummert i tabell 3 og tabell 2.

Parameter	**Verdi**
Spenning til lysdioder	3.3V
Motstandsverdi for LED	120 Ω
Antall dioder	7 (4 gule, 2 grønne, 1 rød)
Effektforbruk (maks)	244.73 mW
Effektforbruk (snitt)	122.4 mW

Tabell 3: Oppsummering av spesifikasjoner og ytelse

5.2 Oppsummering

Løsningen som er presentert i denne rapporten oppfyller kravene til en digital terning. Testresultatene viser at systemet fungerer som forventet, med korrekt genererte utfall og stabil drift. Effektforbruket er innenfor forventede rammer, og kretsens design muliggjør enkel reproduksjon.

Videre arbeid: Potensielle forbedringer inkluderer optimalisering av effektforbruket, samt implementering av mer avanserte tilfeldighetsgeneratorer for å sikre et enda mer uforutsigbart resultat.

Konklusjon: Prosjektet har vært en vellykket implementering av en digital terning, og systemet fungerer i henhold til spesifikasjonene. Kretsen er enkel å forstå, bygge og teste, og oppfyller kravene satt i oppgaven.

6 Takk

Takk til medstudentene for inspirasjon og hjelp underveis. Takk for utsatt innleveringsfrist.