|  |
| --- |
| PROJEKT: SIMULERING REACT-GAME  Test, verifiering och certifiering |
| BILD? |
| **Johan Kämpe**  **2017-12-06**  Mölk Utbildning  Mjukvaruutvecklare inbyggda system |

# Sammanfattning

# Innehållsförteckning

UPPDATERA OCH TA BORT SAMMAATTNING/INNEHÅLL

[Sammanfattning 2](#_Toc500345480)

[Innehållsförteckning 3](#_Toc500345481)

[1 Inledning 4](#_Toc500345482)

[1.1 Syfte och mål 4](#_Toc500345483)

[1.2 Projektkrav 4](#_Toc500345484)

[1.3 Bakgrund 5](#_Toc500345485)

[1.3.1 Beskrivning av tidigare projekt 5](#_Toc500345486)

[1.4 Länkar 7](#_Toc500345487)

[1.5 Noteringar 7](#_Toc500345488)

[2 Genomförande och resultat 8](#_Toc500345489)

[2.1 Använd programvara 8](#_Toc500345490)

[2.2 Planering 8](#_Toc500345491)

[2.3 Metod 9](#_Toc500345492)

[2.3.1 Skapande av simulationer med verktyget TinkerCad 9](#_Toc500345493)

[2.4 Test av komponenter från reaction-game i TinkerCad 10](#_Toc500345494)

[2.4.1 Test av knappar i TinkerCad 10](#_Toc500345495)

[3 Diskussion och slutsats 11](#_Toc500345496)

[4 Bilagor 12](#_Toc500345497)

# Inledning

Detta kapitel beskriver syftet med projektet, samt hur det planerads och utfördes. OCH BAKGRUND OM SIMULERING? ETC

## Syfte och mål

Syftet med projektet är att simulera ett tidigare färdigt hårdvaruprojekt i ett simuleringsverktyg.

## Projektkrav

För betyget godkänt ska simulering av det tidigare projektet utföras i *minst* ett simuleringsverktyg, om simulering misslyckas ska utmaningar som uppstod beskrivas.

Projektet ska levereras med en **projektrapport** med innehåll:

* **Inledning**
* **Syfte**
* **Mål**
* Beskrivning av **metod**
* **Resultat**
* **Diskussionsdel**

Rapporten ska innehålla **bilagor**:

* **Kopplingsschema**
* **Källkod**
* **Demonstration av simuleringen som utfördes**, eventuellt en länk till en video-fil

Projektet bedöms utifrån dess kvalitet och svårighetsgrad.

Om fler än ett projekt väljs att simuleras, ska dessa ha egna rapporter.

## Bakgrund

### Beskrivning av tidigare projekt

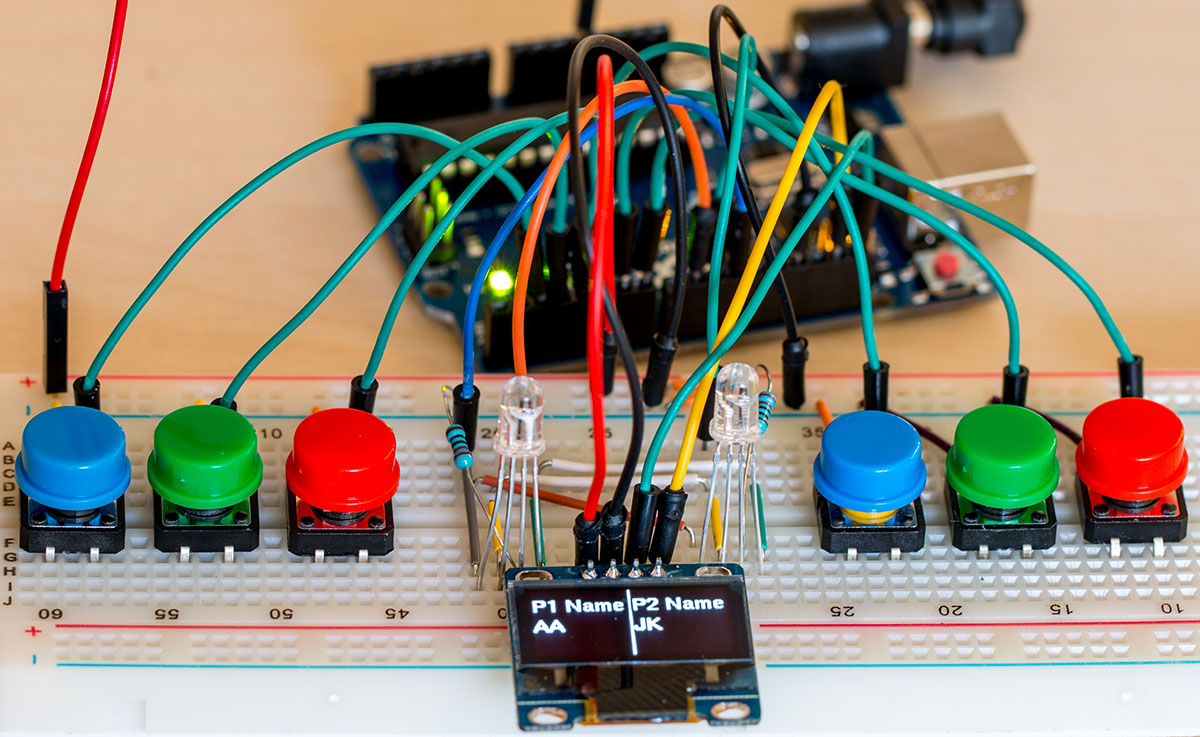
Det befintliga projekt som valdes för simulering kallas *reactionGameRGBLed*, och är ett reaktionsspel för två personer.

Projektet skapades 2016-11-25 av *Mjukvaruutvecklare Inbyggda System*-eleven Johan Kämpe, för att öva på programmering och utveckling i Arduino-miljön. Projektet var ett hobbyprojekt och användes inte i utbildningen.

Projektet bestod av följande komponenter:

|  |  |
| --- | --- |
| Komponent | Antal |
| Tryckknapp röd | 2 st. |
| Tryckknapp grön | 2 st. |
| Tryckknapp blå | 2 st. |
| Lysdiod RGB | 2 st. |
| OLED display SSD1306 | 1 st. |
| Resistor 330Ω | 2 st. |
| Breadboard | 1 st. |
| Arduino UNO-utvecklingskort | 1 st. |

Även flertalet kablar och bygelkablar användes för att koppla samman komponenterna via breadboard-plattan.



Figur 1 Fotografi på uppkopplat och startat reaktionsspel

**Regler för spelet:**

* En av spelarna använder de vänstra tre knapparna, den andra spelaren de högra tre knapparna.
* En av spelarna trycker på sin gröna knapp för att starta en nedräkning från tre sekunder, som visas på displayen.
* När nedräkningen är klar tänds RGB-lysdioderna med en slumpvald färg, antingen röd, grön eller blå.
* Den spelare som först trycker på sin knapp med färg som matchar RGB-lysdioden vinner omgången.
* Om den spelare som trycker först har tryckt på fel färg, så går poängen till den andra spelaren.
* Resultat visas på displayen.

Vid uppstart av spelet ombeds också spelarna att mata in sina initialer med hjälp av knapparna. Blå och röd knapp väljer bokstav från alfabetet, och grön knapp bekräftar.

|  |  |
| --- | --- |
| Figur 2 Displayinformation: vinnare | Figur 3 Displayinformation: poäng |

En länk till en demonstrationsvideo av spelet finns i kapitlet [Länkar](#_Länkar).

## Länkar

**Länkar som tillhör projektet:**

Projektets GitHub-sida  
<https://github.com/GoblinDynamiteer/test-course-simulation-project>

Orginalprojektets GitHub-sida  
<https://github.com/GoblinDynamiteer/arduino_misc/tree/master/reactionGameRGBLed>

Demovideo av reaktionsspelet, på YouTube  
<https://youtu.be/9Vmtv2STFm0>

**Länkar till programvaror och webverktyg:**

TinkerCad  
<https://www.tinkercad.com>

Trello  
<https://trello.com>

## Noteringar

Projektet *reactionGameRGBLed* som används för simuleringen hänvisas till i rapporten som   
*reaction-game.*

SKRIV NÅT OM COPYRIGHT TINKERCAD? BILDER ETX

# Genomförande och resultat

Beskrivning

## Använd programvara

Programvaror som har använts i projektet

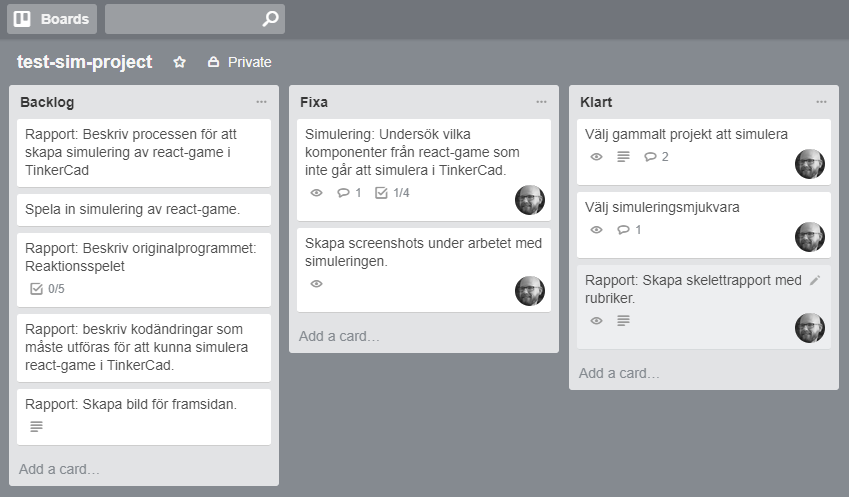
* **Microsoft Word 2016**
* **Autodesk TinkerCad**
  + Webbaserat simuleringsprogram med stöd för Arduino och vanliga komponenter.
* **Atom text editor**
* **Trello**
  + Webbaserat planeringsverktyg.

## Planering

Planering av projektet utfördes med planeringsverktyget *Trello*.

Uppgifter (kallade kort) att göra i projektet lades in till tavlan *Backlog* och flyttas sedan till *Fixa* när de påbörjas. När en uppgift är klar flyttas den till *Klart*.

Kort och kommentarer i Trello-projektet användes delvis som underlag till projektrapporten.

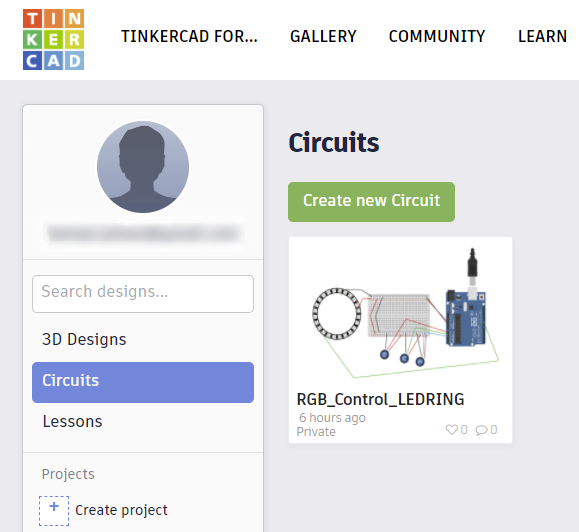


Figur 4 Planering med verktyget Trello

## Metod

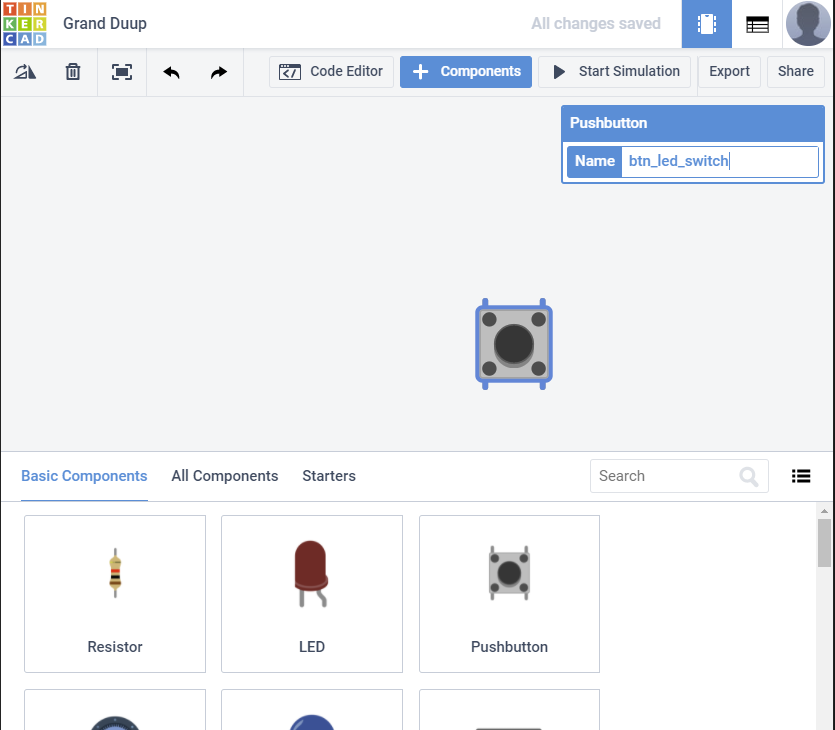
### Skapande av simulationer med verktyget TinkerCad

För att skapa en ny kretssimulering i TinkerCad används undermenyn *Circuits* och sedan knappen *Create new circuit*, från TinkerCads huvudsida efter inloggning av användaren.



Figur 5 Undermenyn Circuits på TinkerCad

När ett nytt kretssimuleringsprojekt har skapats kan komponenter som ska användas läggas till med knappen *+ Components.* Komponenter visas i en lista och kan dras in till arbetsytan med muspekaren.



Figur 6 Tilläggning av tryckknapp i TinkerCad

Komponenter kan namnges, flyttas och raderas.

BESKRIV SIMULERING START

## Test av komponenter från reaction-game i TinkerCad

Komponenter från *reaction-game* testas initialt en och en, för att fastställa vilka som kommer vara möjliga att simulera och om ursprungskällkoden måste modifieras.

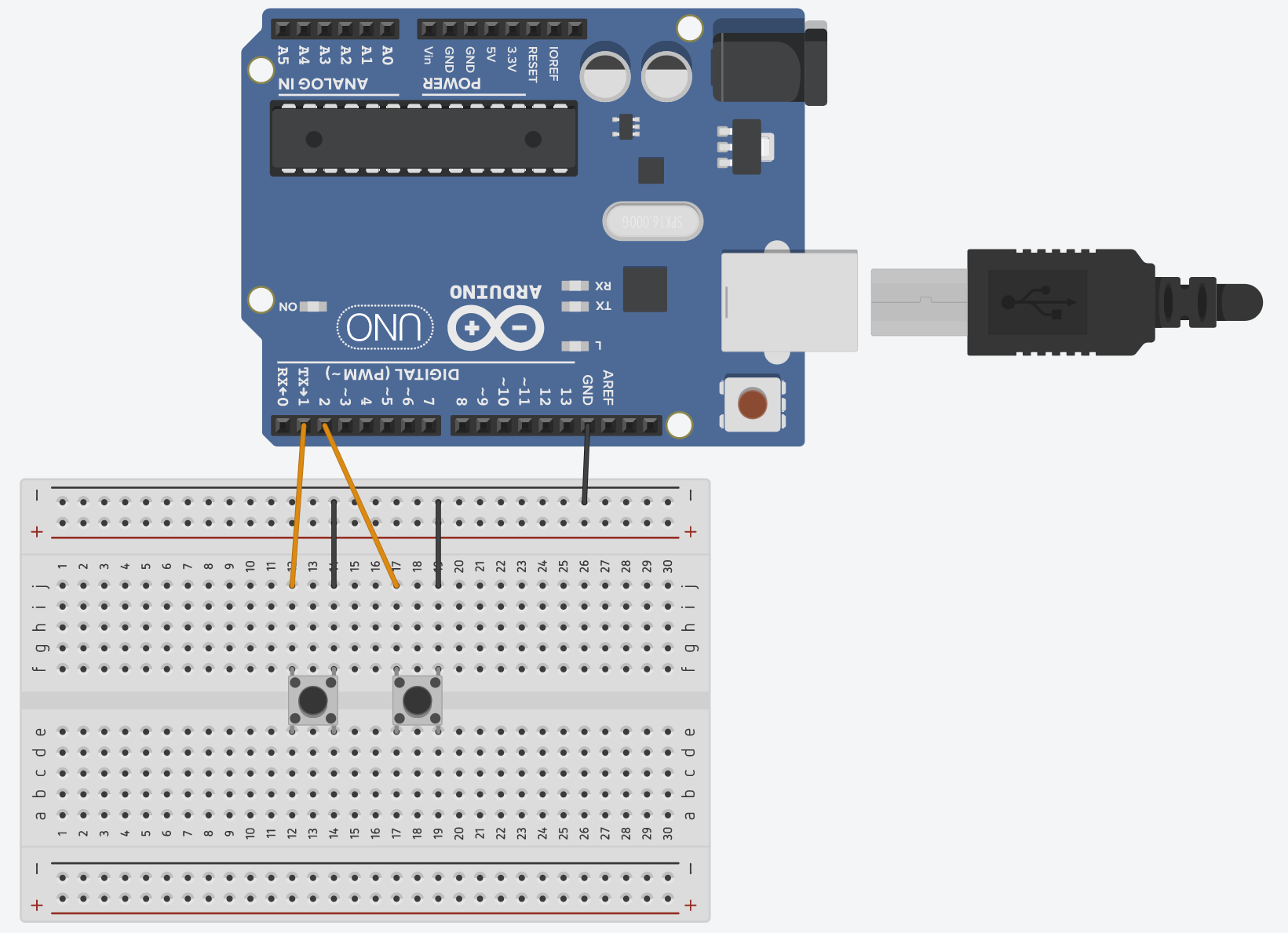
### Test av knappar i TinkerCad

Reaction-game innehåller sex knappar, för att säkerställa att knappar fungerar i TinkerCad skapades en testkod, se [Bilaga 1](#_Bilaga_1:_Testkod).

I testkoden används två knappar, en för att tända onboard-lysdioden kopplad till pin 13 på en Arduino UNO, och en för att släcka den.

Likt orginalkoden för reaction-game användes biblioteket *Button* av Michael Adams för att skapa knappar.

Komponenter och kod lades in i ett nytt TinkerCad-projekt



Figur 7 TinkerCad-projekt med två knappar och en Arduino UNO

Vid simuleringsstart gavs följande fel:

**fatal error: Button.h: No such file or directory**

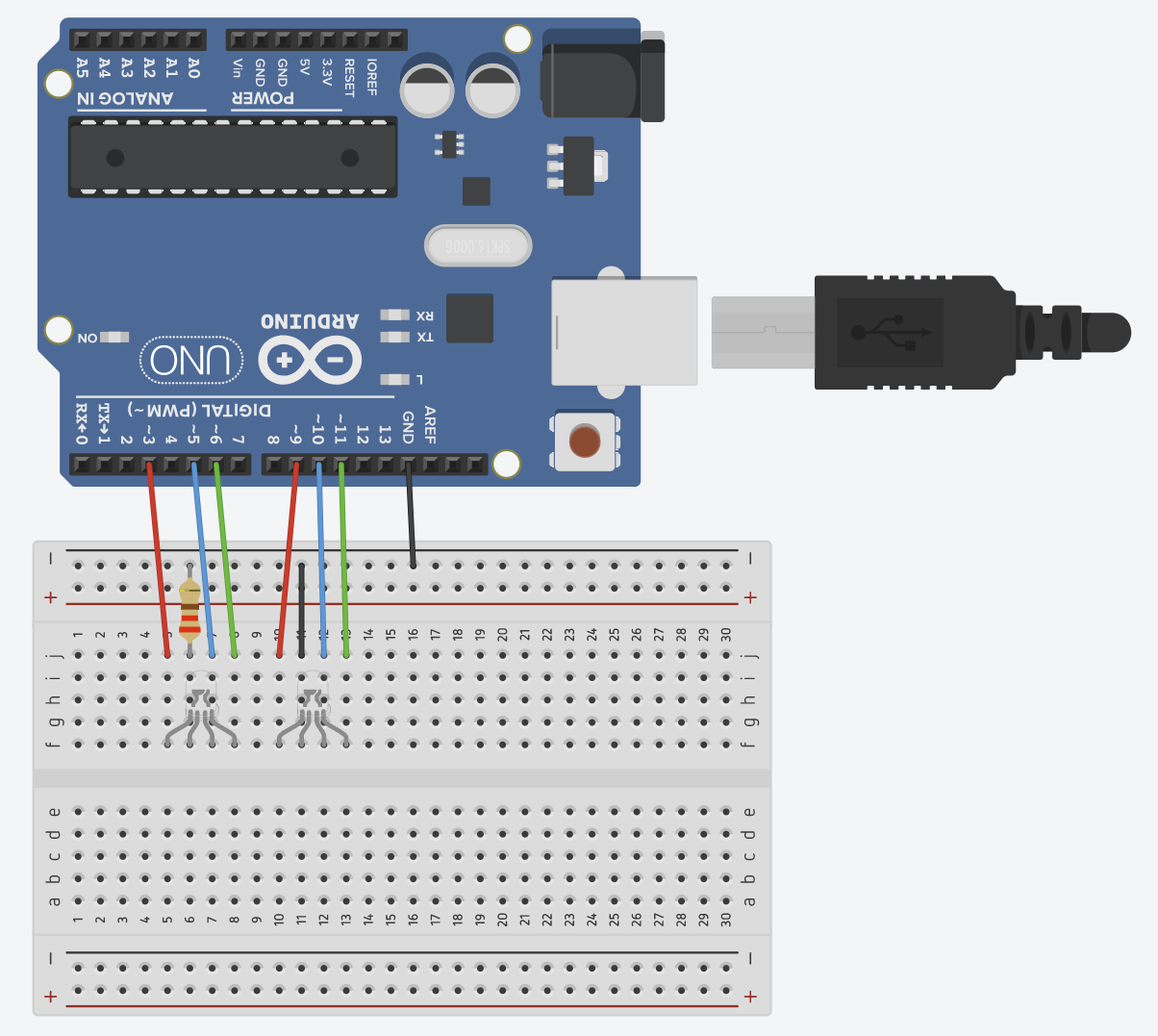
Således antogs det att biblioteket *Button* ej går att använda i TinkerCad. Testkoden byggdes om, se [Bilaga 2](#_Bilaga_2:_Testkod), så att inget bibliotek för knappar används. Ingen hänsyn togs till *bouncing* i denna kod.

Med den modifierade koden startar simuleringen och knapparna fungerar som förväntat.

### Test av RGB-Lysdioder och resistorer i TinkerCad

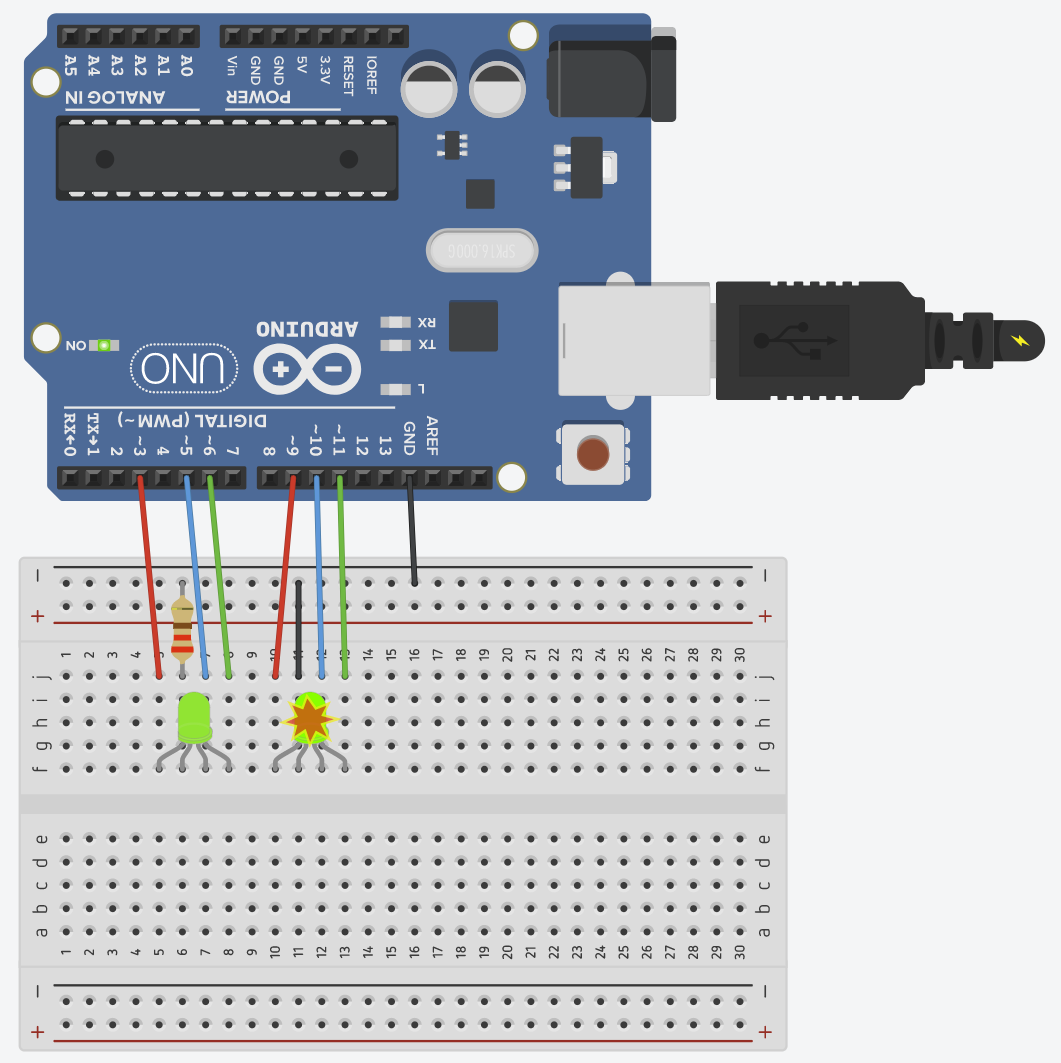
Reaction-game innehåller två RGB-lysdioder, för att säkerställa att dessa fungerar i TinkerCad skapades en testkod, se [Bilaga 3](#_Bilaga_3:_Testkod).

Komponenter och kod lades in i ett nytt TinkerCad-projekt. Två RGB-lysdioder kopplades till en Arduino UNO, för den ena sattes en resistor på 220Ω mellan jord och dess gemensamma katod.



Figur 8 TinkerCad-projekt med två RGB-lysdioder, en resistor och en Arduino UNO

Simuleringen startar och lysdioderna cyklar mellan tre olika färger med olika styrka (PWM). Simuleringen indikerar att lysdioden utan resistor går sönder vid hög styrka.



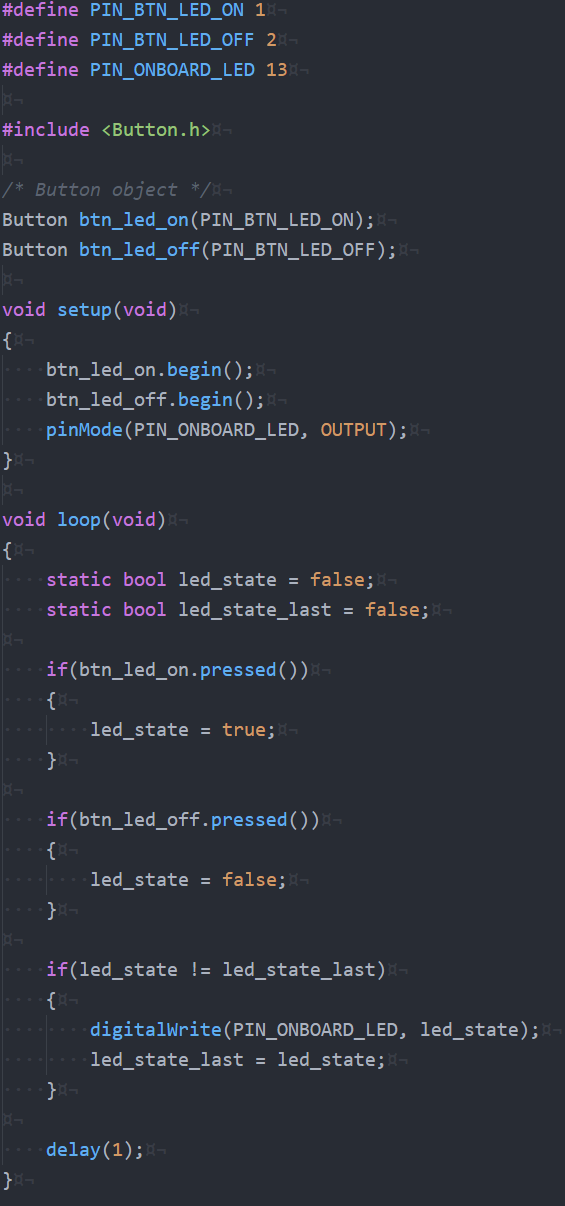
Figur 9 Indikation i TinkerCad, att en RGB-Lysdiod (den högra i bilden) har gått sönder

Detta visar att TinkerCad har möjlighet att simulera RGB-lysdioder och resistorer.

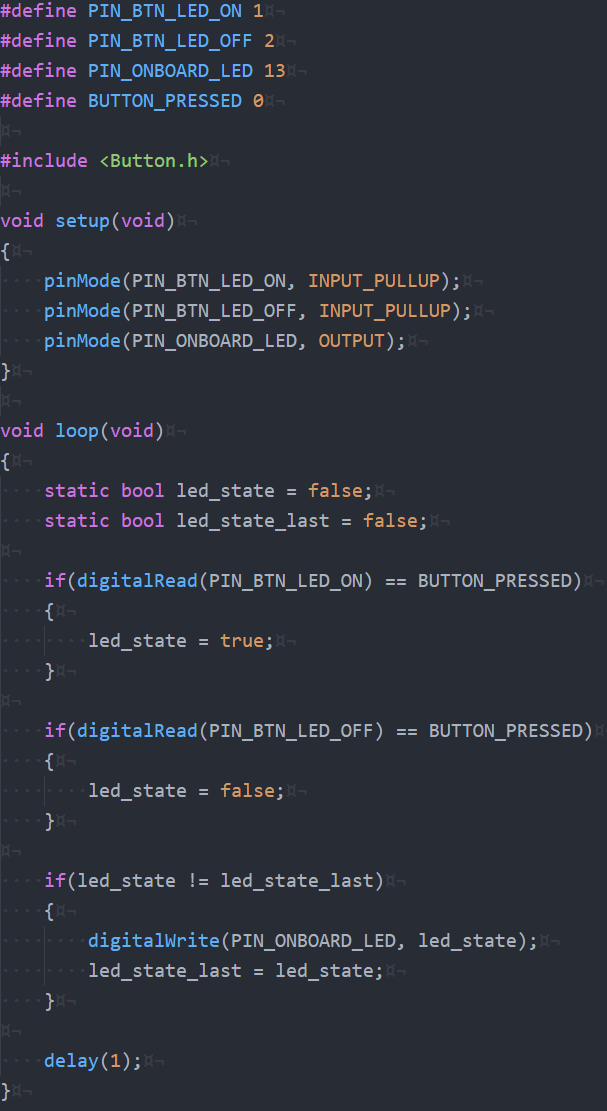
# Diskussion och slutsats

# Bilagor

#### Bilaga 1: Testkod för knappar med bibliotek Button med TinkerCad



#### Bilaga 2: Testkod för knappar utan bibliotek med TinkerCad



#### Bilaga 3: Testkod för RGB-LED och resistor med TinkerCad

