



PROGETTO – GIOCO DELL'OCA

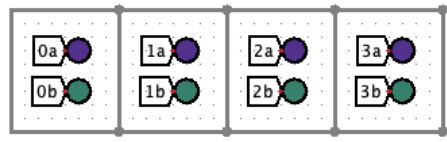
Andrea Cambiaghi

IDEA DEL PROGETTO

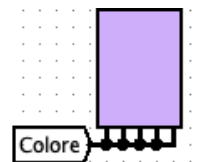
- Il progetto consiste nella creazione di un circuito che simuli il “Gioco dell’oca”
- Questo verrà fatto attraverso l'utilizzo del software «Logisim»
- Il gioco è per due persone e ad ogni turno ogni giocatore lancia due dadi e si sposta del numero di caselle indicate dai dadi.
- La mappa di gioco sarà composta da caselle bianche e/o da caselle che contengono un bonus/malus (es. ritirare i dadi)
- La partita finisce quando un giocatore raggiunge il traguardo.

SCHEMATA DI GIOCO

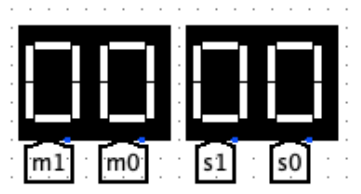
- Luci Led che indicano la casella, si accendono quando il giocatore ci passa sopra



- Una Matrix Led che indica con il colore del giocatore il turno di gioco



- Un timer composto da due Digit Display

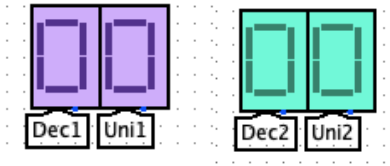


SCHERMATA DI GIOCO

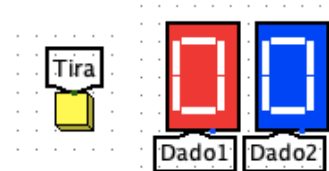
- Due bottoni di start e di stop per avviare/terminare il gioco



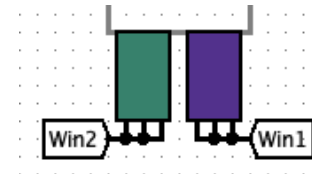
- 4 Digit Display che indicano la posizione dei giocatori



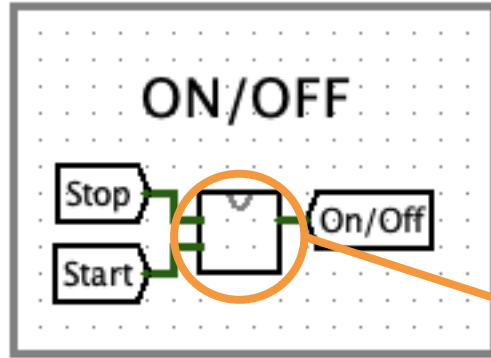
- Un Bottone e 2 Digit Display che simulano il lancio dei dadi



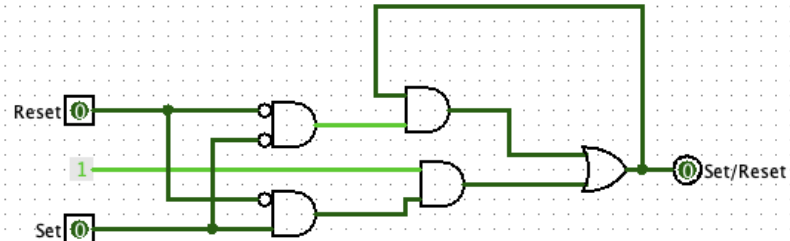
- Due Matrix Led che si accendono quando il rispettivo colore raggiunge il traguardo



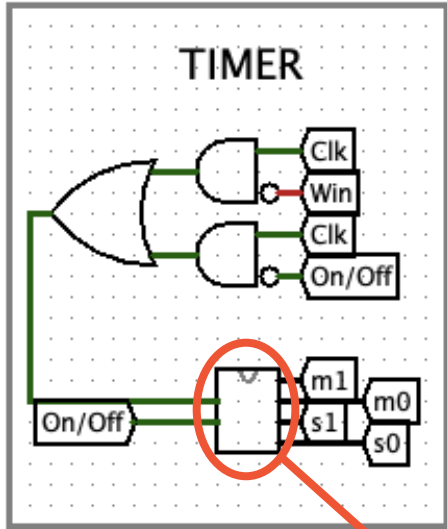
GESTIONE ON/OFF



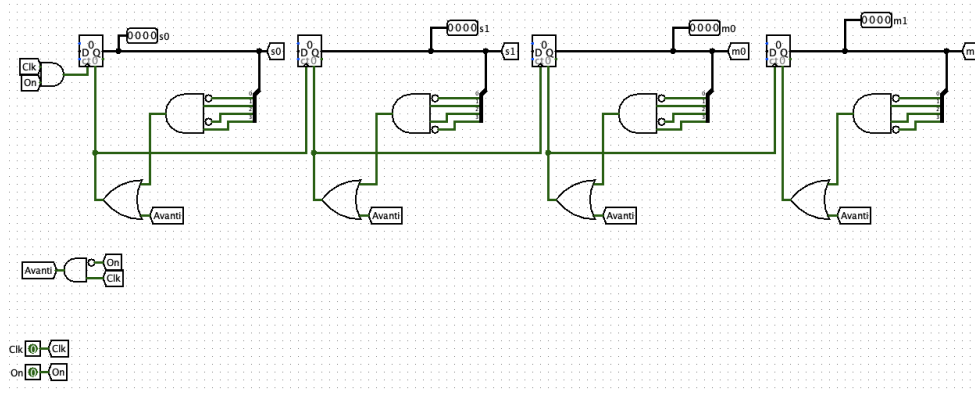
Utilizzo di un set/reset per avviare e terminare il gioco e di conseguenza accendere e spegnere i vari circuiti



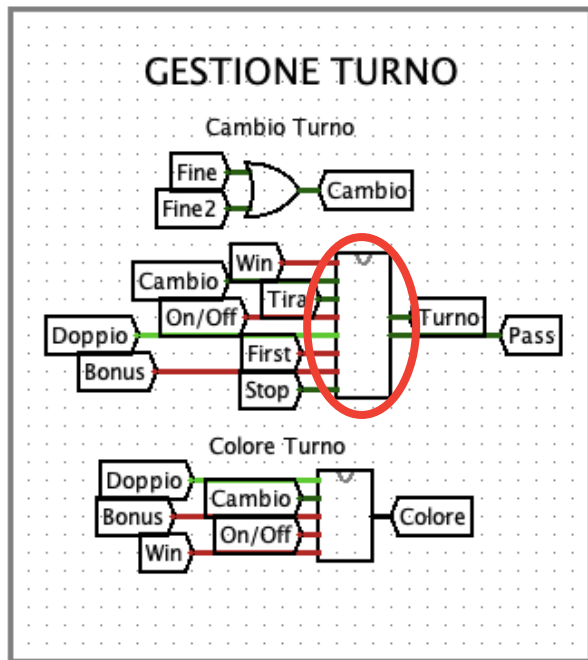
GESTIONE TIMER



Per rappresentare il timer sono stati utilizzati 4 counter per simulare il passare dei secondi. Il primo incrementa ad ogni colpo di clock, i successivi quando il counter precedente raggiunge la cifra massima. Il timer incrementa ad ogni colpo di clock fino a quando qualcuno non vince.

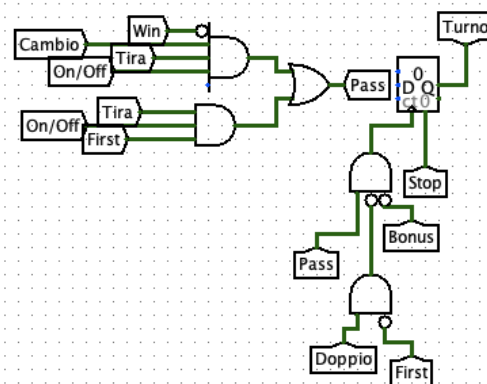
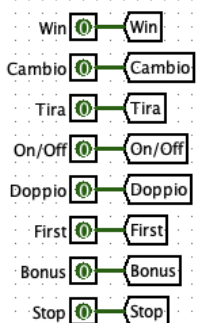


GESTIONE TURNO

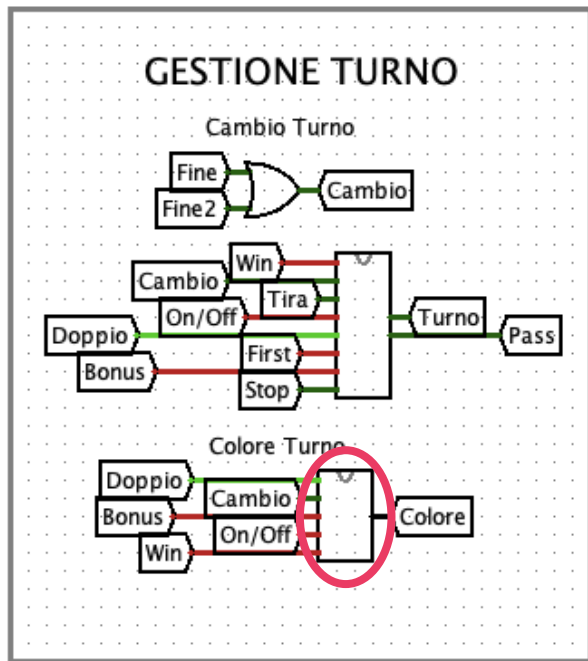


Per la gestione del turno si utilizza un contatore ad un bit, quando l'uscita è 1 è il turno del giocatore 1, altrimenti del secondo.

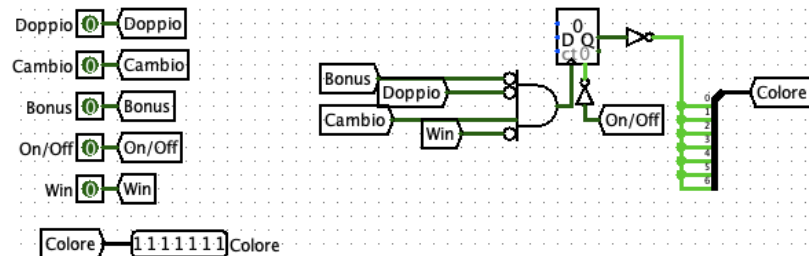
Il cambio turno avviene quando uno ha tirato i dadi, sempre se il giocatore che ha appena tirato non è finito su una casella che gli permette di ritardare o nel caso in cui faccia doppio, in questo caso il turno spetta ancora a lui.



GESTIONE TURNO

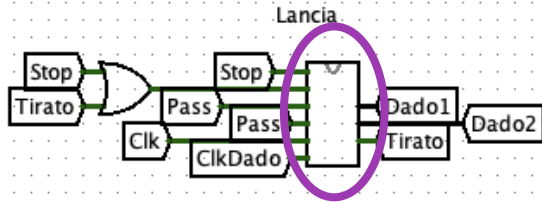


In base al turno del giocatore «Colore» può assumere 2 valori, 0 per indicare il giocatore azzurro e 1 per il colore viola.
Cambia valori in base alle dinamiche di gioco.



GESTIONE DADI

GESTIONE DADI



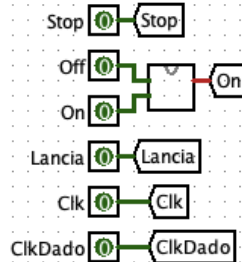
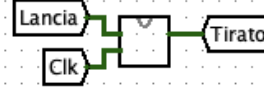
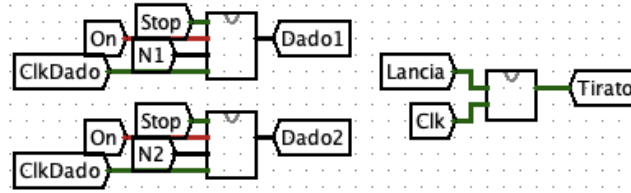
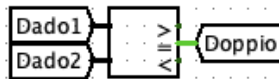
Primo Tiro



Somma Dadi

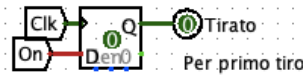


Tiro Doppio

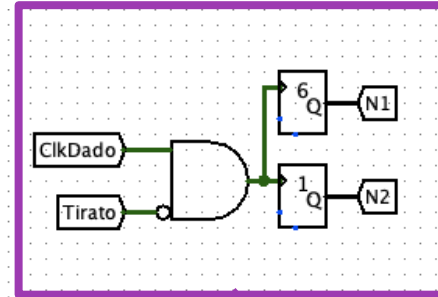


Dado1 0000 Dado1

Dado2 0000 Dado2



Per primo tiro

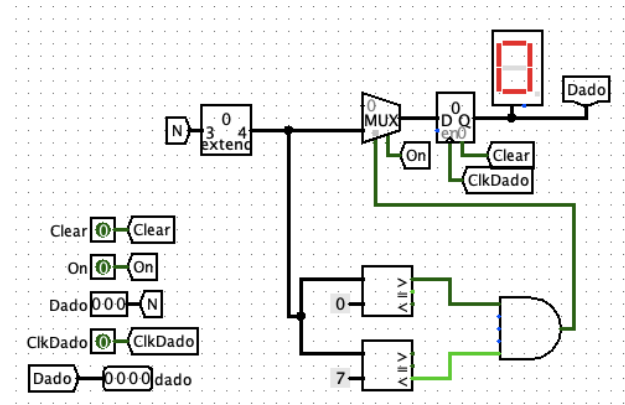


Generazione di numeri a 3 bit attraverso l'uso di generatori random.

GESTIONE DADI

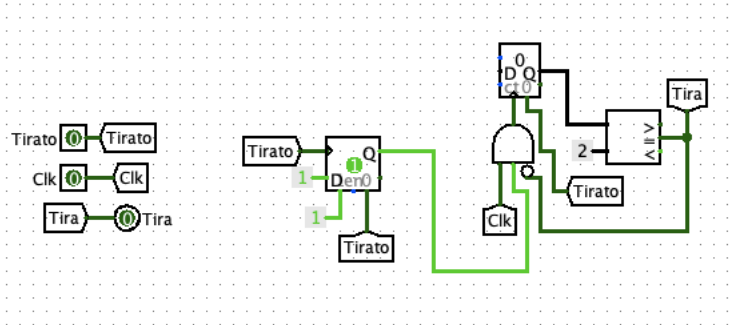
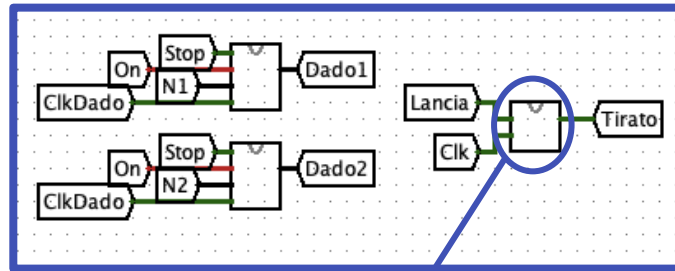
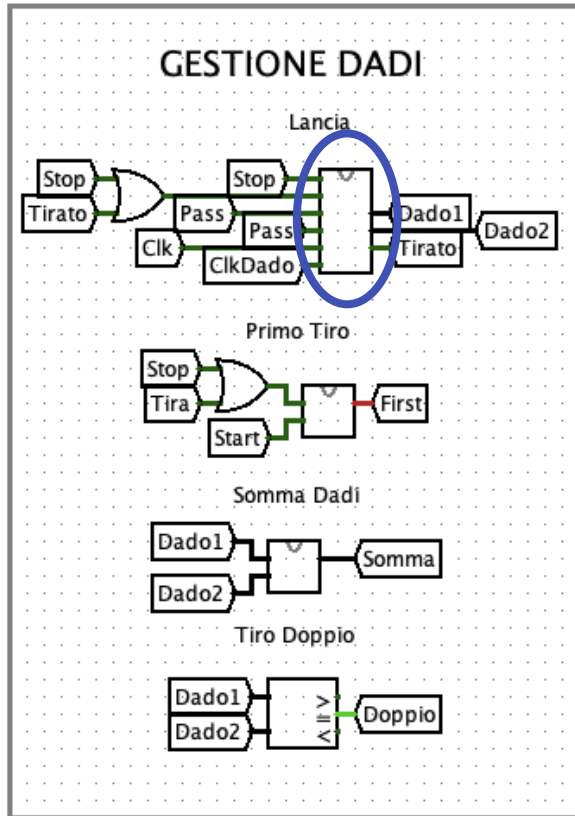
The diagram illustrates the logic for a dice game, divided into four main sections:

- Lancia**: This section represents the initial dice roll. It starts with a 'Stop' block and a 'Tirato' block connected to an AND gate. The output of the AND gate goes to a 'Pass' block, which then connects to a 'Clk' block. The 'Clk' block is connected to a 'ClkDado' block, which is in turn connected to a 'Dado1' block. The 'Dado1' block is connected to a 'Dado2' block. The 'Dado1' and 'Dado2' blocks are connected to a 'Tirato' block, which then connects to a 'Dado2' block. A purple oval highlights the 'ClkDado' block.
- Primo Tiro**: This section represents the first roll. It starts with a 'Stop' block and a 'Tira' block connected to an AND gate. The output of the AND gate goes to a 'Start' block, which then connects to a 'First' block.
- Somma Dadi**: This section represents the sum of the dice. It starts with 'Dado1' and 'Dado2' blocks connected to an adder block. The output of the adder block is connected to a 'Somma' block.
- Tiro Doppio**: This section represents the double roll. It starts with 'Dado1' and 'Dado2' blocks connected to an equality comparator block (labeled with \wedge and \vee). The output of the comparator block is connected to a 'Doppio' block.



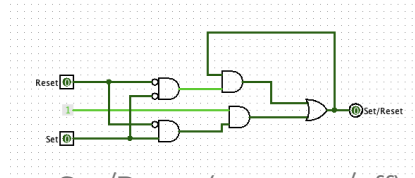
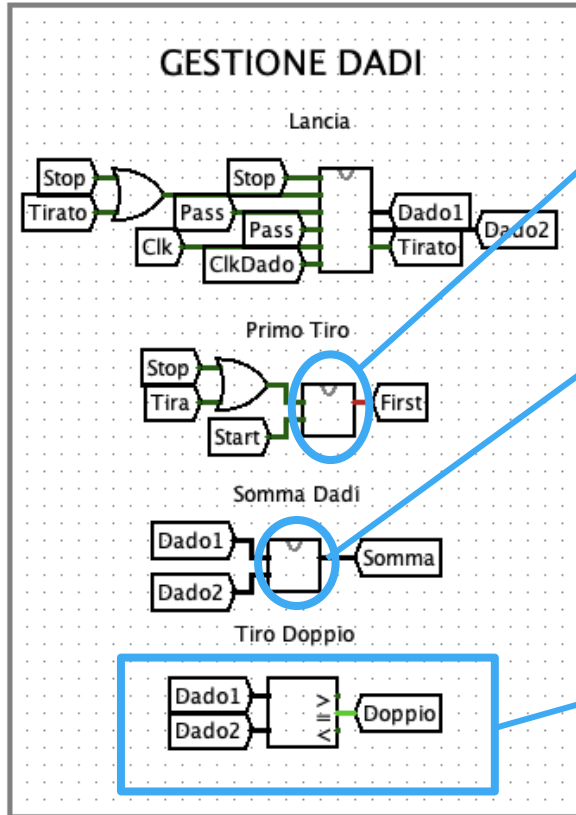
10

GESTIONE DADI

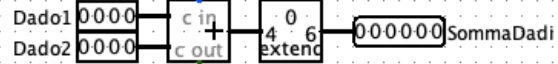


Per simulare il lancio dei dadi vengono generati numeri casuali per 2 colpi di clock, appena raggiunti Tira diventa 1 e i dadi si considerano tirati

GESTIONE DADI



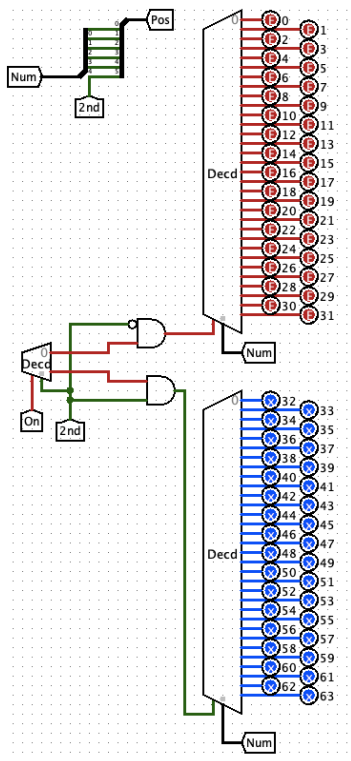
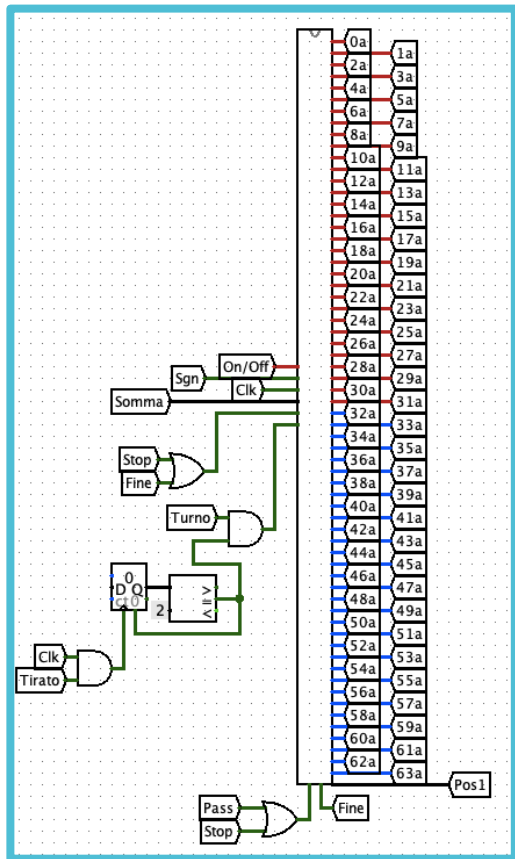
Set/Reset (come on/off)



Somma ed estende a 6 bit il tiro

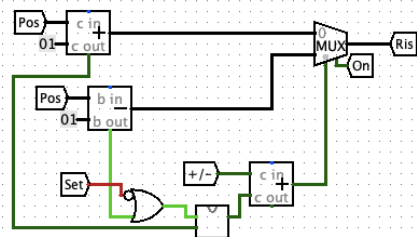
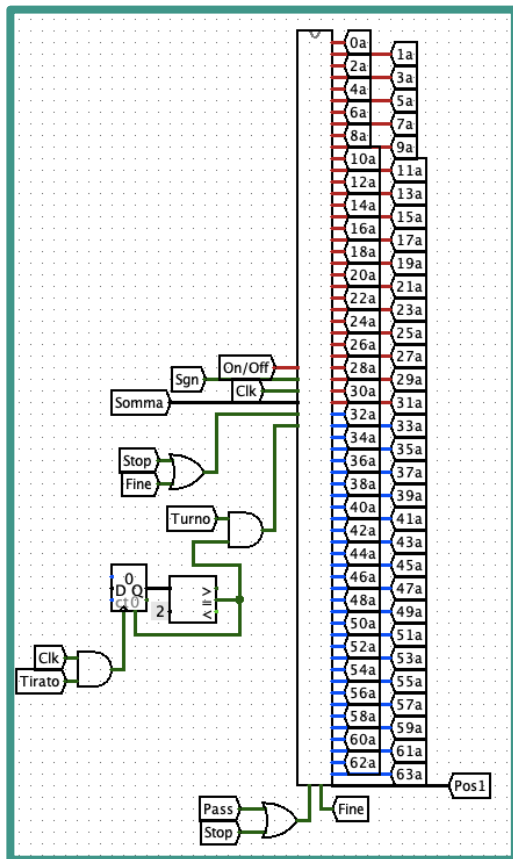
Controlla se il tiro è doppio

GESTIONE MOVIMENTO

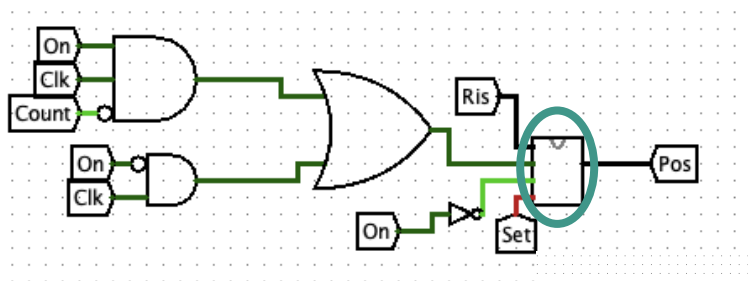


Creazione di un decoder a 6 bit che in base alla posizione accende il led corrispondente, quando il MSB della posizione è 0 si attiva il primo DEC, quando è 1 si attiva il secondo, simulando così un decoder a 6 bit.

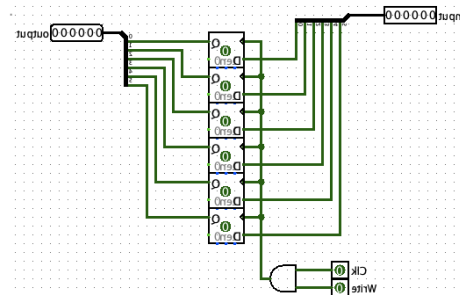
GESTIONE MOVIMENTO



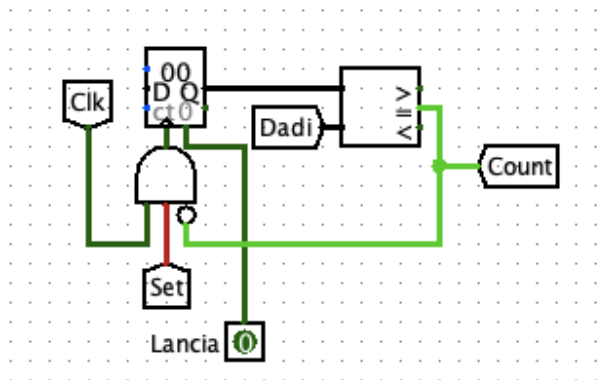
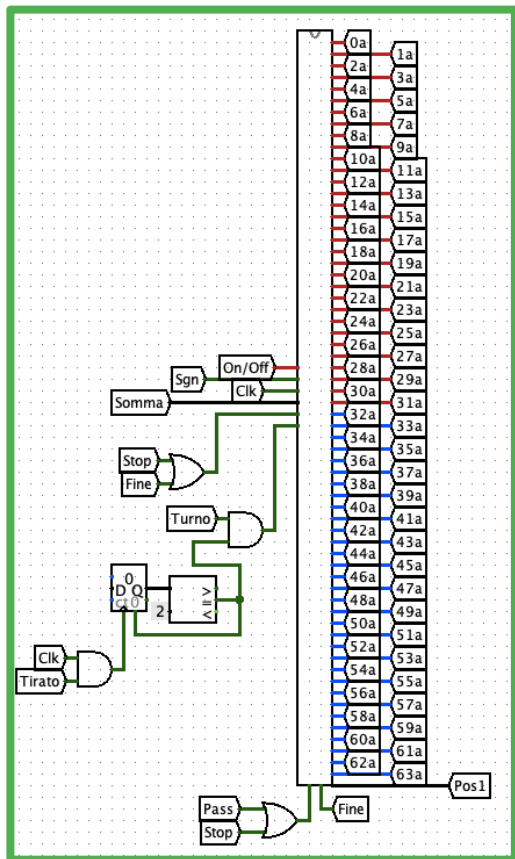
Incrementa o decrementa di 1 (in base al selettore del MUX) posizione della pedina.



Inserisce in Ris in Pos (quindi Pos \pm 1) fintanto che Set è 1



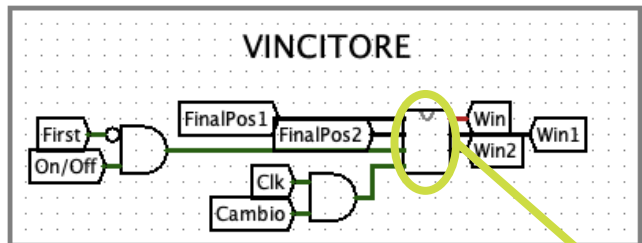
GESTIONE MOVIMENTO



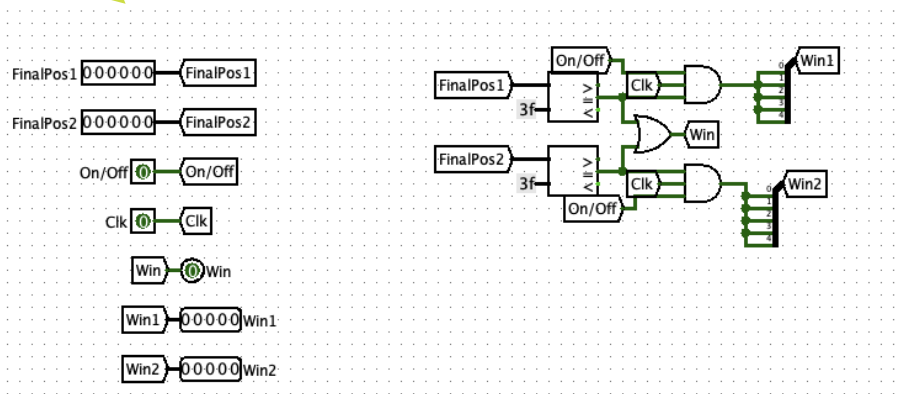
Incrementa di 1 fino a quando non si raggiunge il numero di dadi

[illegible]

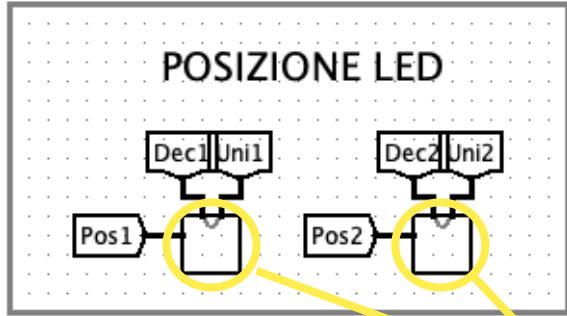
GESTIONE VINCITORE



Appena uno dei due giocatori raggiunge la casella finale con un tiro dei dadi preciso si attiva il comparatore che a sua volta impone ad 1 il tunnel del giocatore vincente



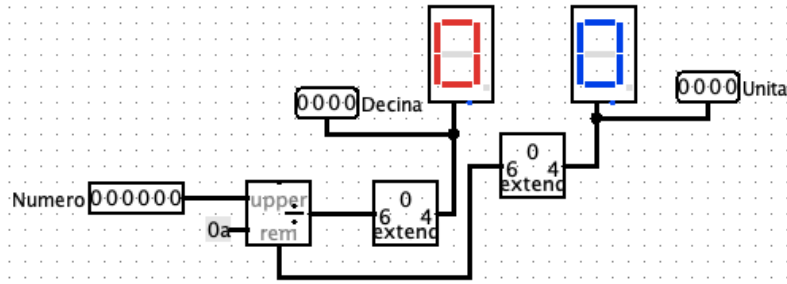
GESTIONE STAMPA POSIZIONE



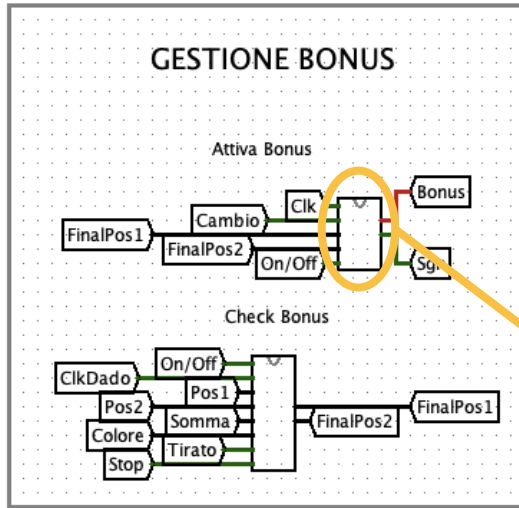
Circuito che stampa il numero in base decimale nei Display il numero della casella.

Le decine sono ottenute prendendo il quoziente della divisione intera per 10, mentre le unità si prende il resto della divisione per 10.

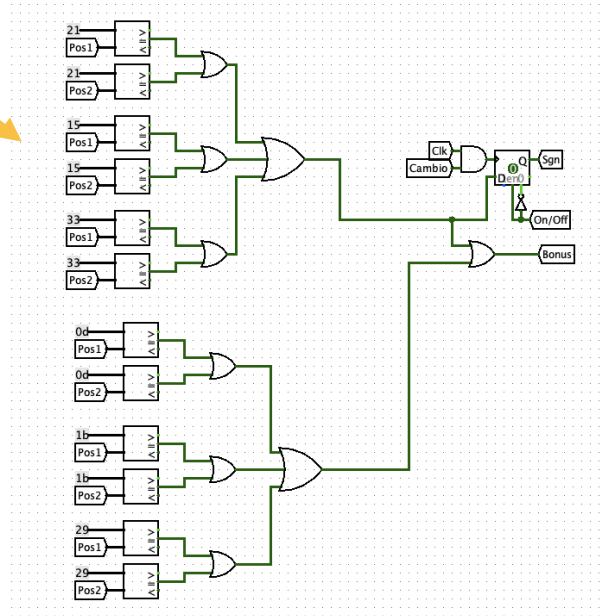
Entrambe le cifre vengono ridotte a 4 bit.



GESTIONE BONUS

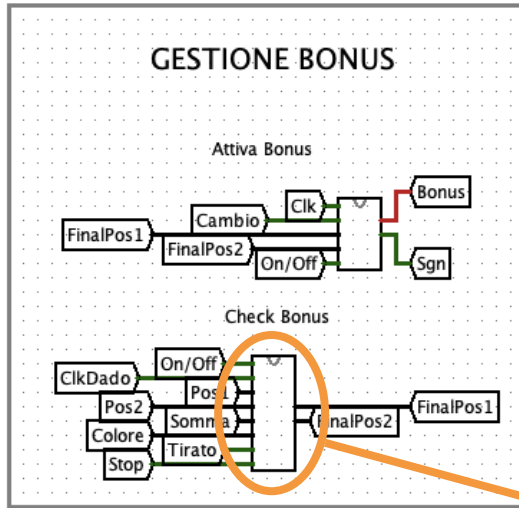


Il circuito rende ad 1 BONUS se un giocatore si ferma su una casella che contiene il bonus.
Il segno in uscita dipende se la casella offre un bonus che permette di spostarsi (0) avanti o indietro (1).



Il segno viene memorizzato attraverso un flip flop d e azzerato quando finisce il movimento.

GESTIONE BONUS



Circuito che controlla se sulla casella dove dovrà finire il giocatore ci sarà un bonus, così da impedire il cambio turno e far ritirare lo stesso giocatore. Memorizzando la posizione finale solo nell'istante in cui tira, calcolando la somma tra la posizione attuale (prima del tiro) e il tiro effettivo.

