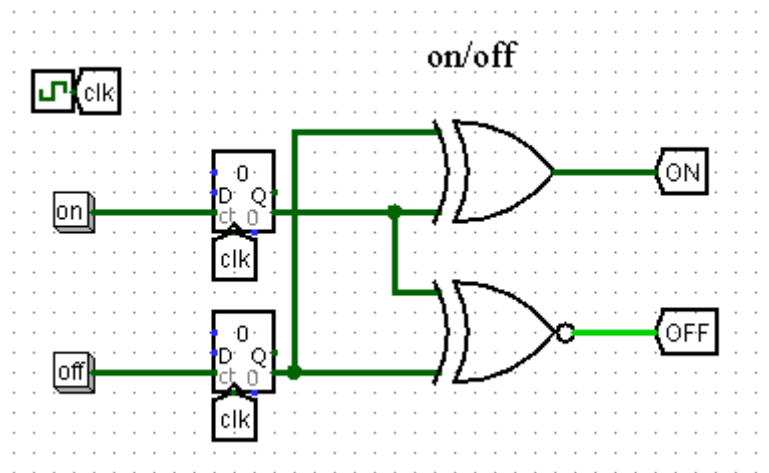
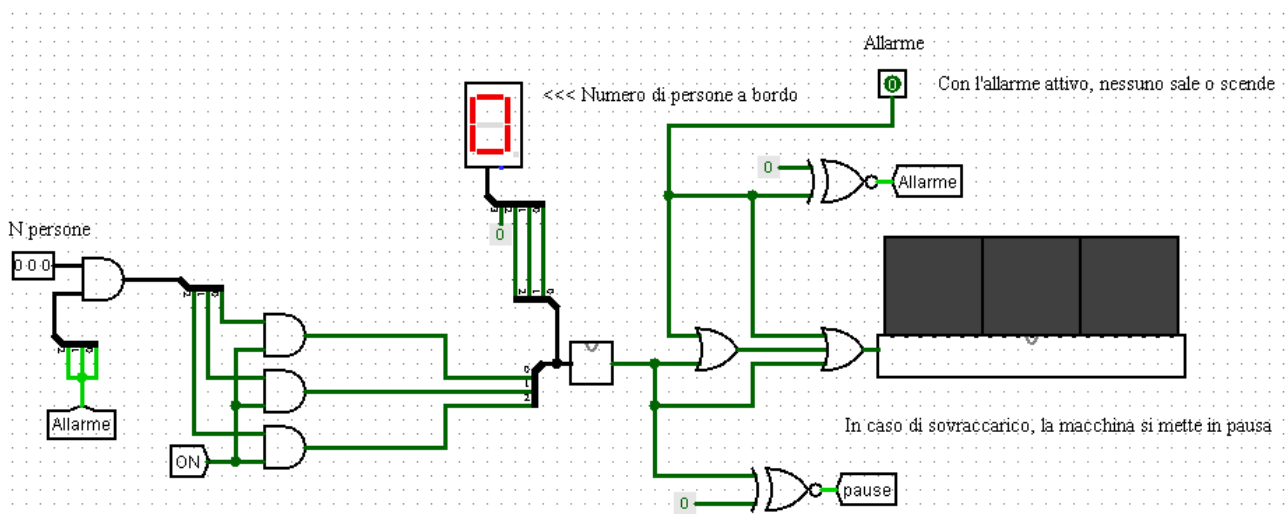


# Simulazione di un ascensore

Il progetto consiste nel simulare il funzionamento di un normale ascensore. La macchina è divisa in diverse parti, le quali controllano differenti funzioni. C'è un blocco che si occupa solo dell'accensione e dello spegnimento, un altro che ci permette di inserire come input il numero di persone da caricare con la relativa gestione e un ulteriore blocco per la scelta del piano e la visualizzazione dello stesso.



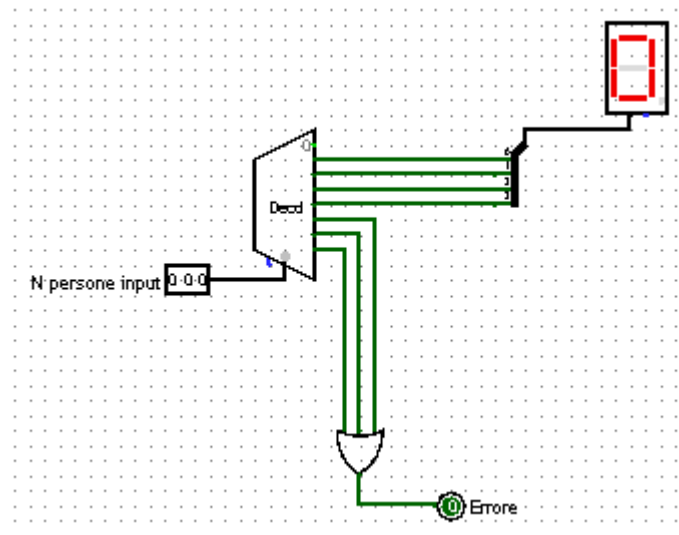
Il circuito mostrato qui sopra ci permette di accendere o spegnere la macchina. È costituito come si può osservare da due contatori che vanno da 0 a 1, una porta XOR e XNOR a 2 ingressi. Il segnale di uscita dei due contatori va a finire in entrambe le porte, così che una volta azionato il clock, in base al bottone premuto l'uscita finale possa essere una e una sola. Per esempio si parte con off a 1 essendo i due ingressi della XNOR entrambi 0, dunque premendo on la XOR passerà a 1 mentre la porta con l'uscita off logicamente passa a 0. Dunque l'utente potrà premere i due bottoni all'infinito, ma ci sarà sempre una e una sola uscita attiva a 1.



Questo circuito invece, permette l'inserimento delle persone come input con la gestione di determinate eccezioni. Una volta inserite, il segnale passa attraverso una AND a 2 ingressi, dove il secondo ingresso è costituito dal segnale d'allarme, che di default è sempre a 1. Una volta

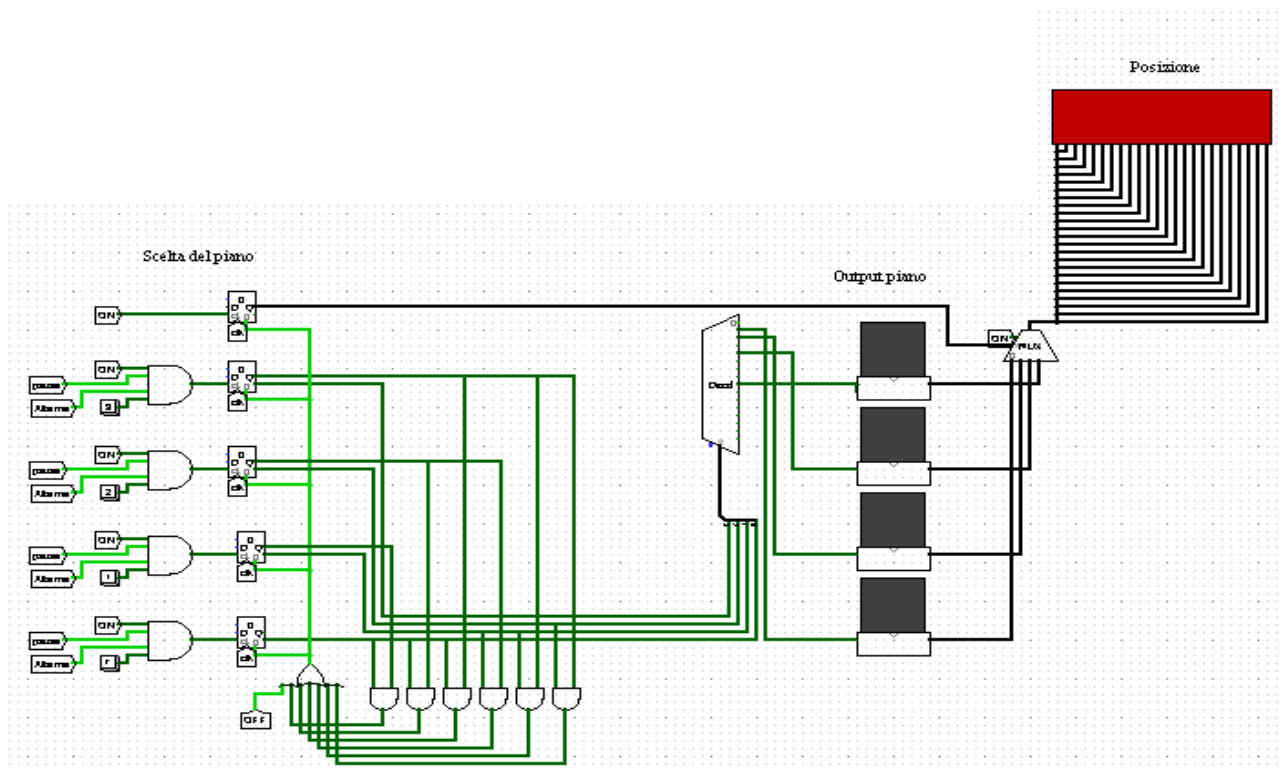
sorpassata questa porta, il segnale su 3 bit viene messo in uno splitter a 3 bit, collegato a sua volta ad altre 3 porte AND a 2 ingressi, dove la seconda entrata è il segnale di on aggiunto mediante tunnel. Queste tre porte, tramite altri 2 splitter, servono per poter visualizzare il numero di persone a bordo su una Hex Digit Display. A questo punto, il segnale arriva ad un sottocircuito il cui compito è valutare quale valore è stato inserito in input e in tal caso mandare l'uscita a 1 e lanciare un'eccezione.

Quindi il circuito secondario mostrato qui accanto, dato un determinato input, decide tramite un decoder quali linee mandare in un OR accendendo l'uscita a 1, mentre le linee rimanenti (quelle accettabili) non vengono toccate.



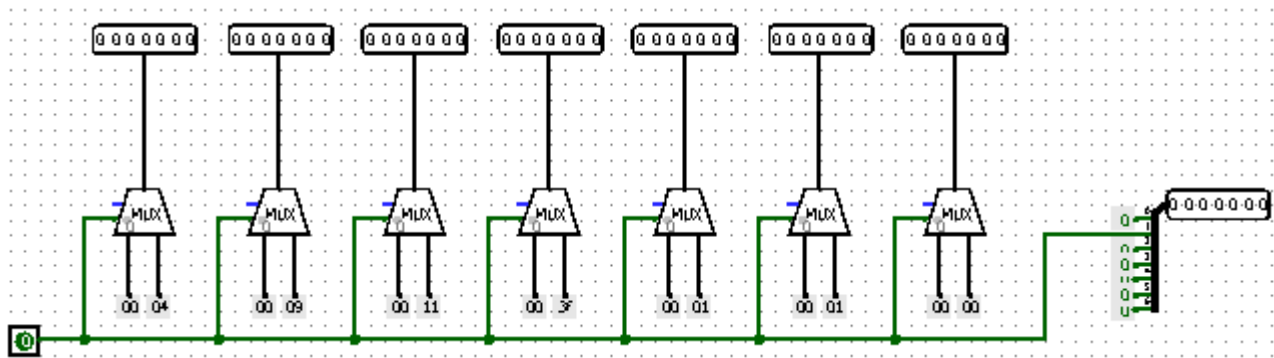
Dunque, se il numero inserito dovesse essere maggiore di 4, l'uscita passerebbe a 1 facendo entrare in gioco la seconda parte di questo blocco. Il segnale passerà per due OR, la prima che serve per mandare a 0 il segnale di pausa nella porta XNOR, mentre la seconda porta servirà ad accendere delle matrici led facendo comparire un messaggio di errore. In questa circostanza la macchina diventerà inutilizzabile e sarà messa in pausa, l'utente dunque non potrà cambiare piano a meno che non faccia scendere le persone in più facendo in modo che la macchina torni a funzionare completamente. L'altra eccezione invece è il segnale d'allarme, attivabile manualmente dall'utente mandando a 1 il relativo pin. In questo modo la porta XNOR relativa all'allarme passerà a 0, oltre ad accendere di nuovo le matrici led come per il sovraccarico, con la differenza di non poter far salire o scendere nessuno poiché con l'allarme a 0, la prima AND iniziale non andrà mai a 1.

Durante questa parte dello sviluppo è risultato leggermente complesso trovare un modo per isolare le due eccezioni senza che entrassero in qualche modo in conflitto, ma dopo vari tentativi con 2 OR differenti, è stato possibile risolvere questo problema.



Grazie a questo grande blocco, l'utente ha la possibilità di selezionare il piano da voler raggiungere con la visualizzazione dello stesso. L'input iniziale è formato in questo modo: AND a 4 ingressi, i quali sono il segnale di on, pausa, allarme e il bottone del relativo piano. L'allarme e la pausa, come descritto in precedenza, partono a 1 e nel momento in cui viene lanciata anche solo una delle due eccezioni, passano a 0 così che tutte le AND risultino sempre 0 in ogni caso rendendo la macchina inutilizzabile. Per ogni AND a 4 ingressi, abbiamo un contatore che va da 0 a 1, il quale serve a mantenere il piano attivo fino alla scelta del successivo. I segnali di stato prossimo dei contatori vanno a finire in uno splitter collegato a un decoder, il quale accende la matrice led corrispondente con il piano scelto. Oltre a ciò, ogni segnale di stato presente e stato prossimo tra i 4 piani, viene accoppiato in alcune AND, così facendo tutte le possibilità di conflitto di piani accesi in contemporanea si possono verificare (Ad esempio, tra il piano T e il piano 1, poi tra T e il 2 e così via per tutti e quattro i piani). In quel caso si accende la relativa porta, la quale andrà a finire in una OR con tanti ingressi quante sono le porte AND mandando un segnale di clear a tutti i contatori. Dunque, ogni cambio di piano avviene in 2 cicli di clock. Questo processo è stato il problema principale di questo progetto poiché è stato davvero complesso trovare una soluzione che riuscisse ad avere in output uno e un solo piano senza dover spegnere manualmente quello precedente.

Infine, dopo che l'input è arrivato alle matrici led che visualizzano il piano, il segnale arriva ad un'ultima grande matrice che tramite un multiplexer a 7 bit con 2 bit di selezione riesce a simulare la posizione all'interno dell'ascensore rispetto al piano scelto.



Per quanto riguarda invece i sottocircuiti che fanno visualizzare i dati sulle matrici led, questo ne è un esempio. Vengono usati multiplexer a 7 bit con una costante 0 e come altro ingresso un valore esadecimale fisso in base a quale dato vogliamo far visualizzare. Il segnale arriva infine a uno splitter sulla linea corrispondente al piano così da avere poi sull'ultima matrice la posizione corretta.