Progetto di Laboratorio
Architettura degli Elaboratori I

Macchina Enigma Semplificata

di Giovanni Petrillo
Matricola T15550

Introduzione

L'idea di questo progetto è stata ispirata dal video "How the Enigma Machine Worked" di Jared Owen, che trovate qui, nel quale si spiega nel dettaglio il funzionamento della famosa macchina e delle sue varie componenti.

Specifica

La macchina ha lo scopo di criptare un messaggio lettera per lettera: queste vengono cambiate grazie al passaggio in 2 componenti, una **Plugboard** e dei **Rotori**.
Quella che presento è una versione di Enigma molto semplificata: sono previste 7 lettere in input, 2 cavi di collegamento e solo un passaggio per lettera in ogni componente.
Ho volutamente non considerato la possibilità di errori nella fase di settaggio iniziale, (ad esempio inserire la stessa lettera sostituibile più volte) poiché questo viene eseguito da utenti razionali con istruzioni specifiche su come settare la macchina.

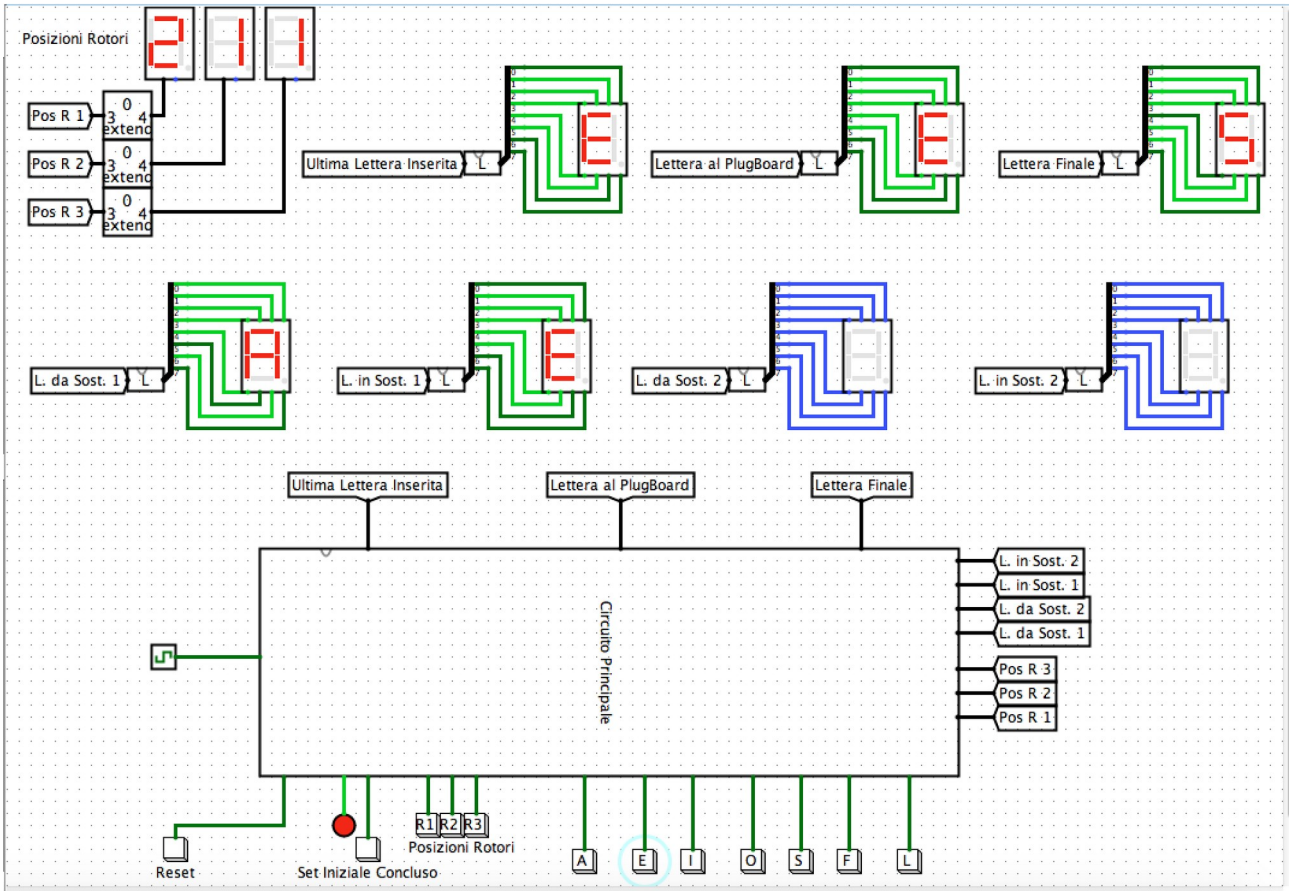
Funzionamento

L’utente prima di usare la macchina deve confermare di aver completato il settaggio iniziale (opzionale) ovvero scegliere quali lettere collegare tra loro nella **Plugboard** e quali posizioni iniziali abbiano i **Rotori**.
Premendo il pulsante di “**Settaggio Completato**” si attiva la macchina che inizia a criptare le lettere premute e a far spostare i rotori come in seguito spiegato.
Con il pulsante di **Reset** si riporta la macchina allo stato iniziale.
Il funzionamento della macchina è con **Clock** attivo con frequenza di 8 Hz.

Componenti

Di seguito analizzo uno per uno i vari sotto circuiti del progetto

Interfaccia Utente



Per rendere più visibile e immediato il funzionamento della macchina ho creato una interfaccia collegata al **Circuito Principale**.
Qui vediamo in un’unica schermata tutti i passaggi fatti nel settaggio e nel passaggio di una lettera nei vari componenti della macchina.
L’utente tramite i pulsanti **Posizione Rotori** può settare la loro posizione iniziale visibile nei led (Hex Digit Display) in alto a sinistra.

Premendo sulle lettere l'utente le collega tra loro nella Plugboard: la prima verrà sostituita dalla seconda e la terza dalla quarta; tutte sono visualizzate su schermi Led (7 Segment) in modo che rimangano sempre visibili all'utente.
Una volta completato quanto sopra, o volendo saltarlo, si preme il pulsante di **Set Iniziale Concluso**, la cui attivazione viene confermata dal led acceso al suo fianco.
Da questo momento ogni lettera che viene inserita viene visualizzata nel led **Ultima Lettera Inserita**, l'output della plugboard sul secondo led **Lettera al Plugboard** e invece la lettera all'uscita sul led **Lettera Finale**.

Circuito Principale

Questo è il nucleo della macchina dove sono collegate tutte le sue principali componenti: la **Pulsantiera**, la **Plugboard** e i **Rotori**.
Abbiamo inoltre 4 memorie: una per lo stato di **Setting Completato** con un Flip-Flop S-R (collegato all'output led), una per la memoria dell'ultima lettera inserita (**Mem. Lettera**), una per i collegamenti plugboard (**Mem. Plug**) e una per la posizione dei rotori (**Mem. Rotori**).
Il suo funzionamento è il seguente: La pulsantiera converte la mia lettera in un numero su 3 bit che viene passato ad un de-multiplexer collegato allo stato del setting: finché questo è incompleto l'input viene passato alla memoria della plugboard e salvato come una delle 4 lettere collegate, a setting completo invece viene passato alla memoria della lettera inserita poi visualizzata su led.
Questo valore viene poi passato alla **Plugboard** che lo confronta con le 2 lettere da sostituire e nel caso lo cambia, passandolo anche come output su led.
Infine l'output della **Plugboard** viene passato ai rotori che restituiscono a loro volta l'output finale.

Pulsantiera

In questo semplice sotto-circuito converto il mio input lettera in un numero da 7 a 1.
Aggiungo inoltre l'output binario di attivazione tastiera **Pulsante Premuto** che mi serve per aumentare la memoria di posizione dei **Rotori**.

Ultima Lettera Inserita

Questa è una semplice memoria Register di 3 bit dove salvo il mio input con l'aggiunta di un sotto-circuito **Not 0** che verifica che il mio input passato non sia 0. Se lo fosse restituisce 0 (falso) e, collegandolo all'enabler del mio register, mi assicuro che il valore sia salvato solo quando il valore è diverso da 0.

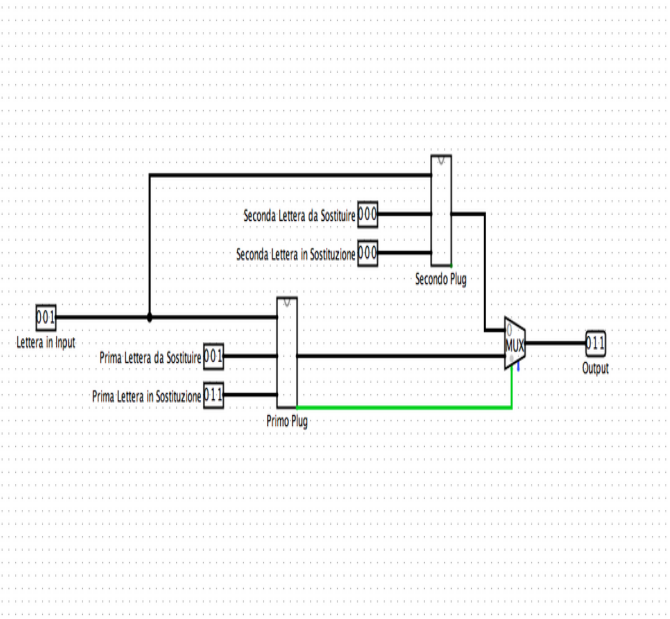
Convertitore Led

Per comodità dovendo identificare una lettera con numero a 3 bit e passarla ad un led che funziona a 8 bit ho salvato tutte le lettere con delle costanti e le ho collegate ad un multiplexer per scegliere direttamente quella giusta col mio input.

Plug Singolo

Questo sotto-circuito controlla se due lettere sono state “collegate” nel setting iniziale. Semplicemente verifico se l’input è uguale alla lettera che va sostituita e, nel caso, passo il valore da sostituire al suo posto.
Aggiungo anche un bit di verifica **Sostituzione Avvenuta** che userò nella plugboard.

PlugBoard



Nella Plugboard unisco 2 “cavi” singoli. Verifico se l’input viene sostituito nel primo cavo: se così il suo output è quello che passa come output finale, se no passo il secondo output a prescindere che la lettera venga sostituita o meno.

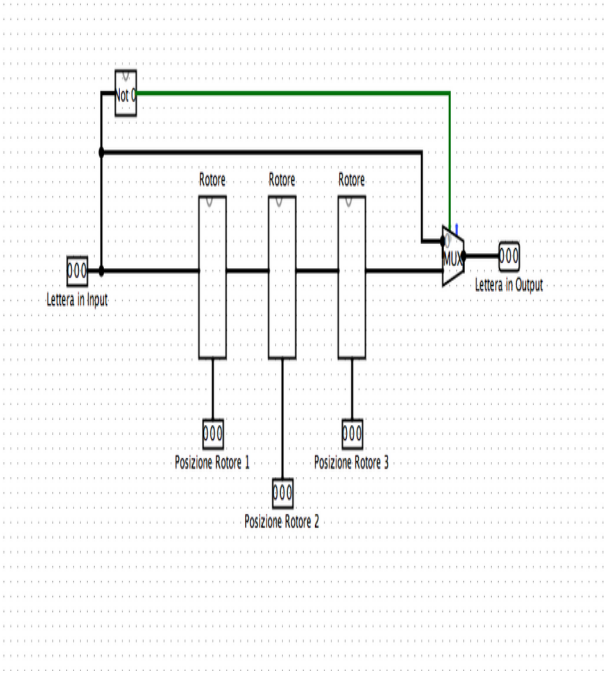
Memoria Plug Board

Qui salvo le possibili 4 lettere da sostituire. Come detto sopra riceverà input diversi da 0 solo se il settaggio iniziale è incompleto.
Qui ad ogni input diverso da 0 (riuso il mio sotto-circuito **Not 0** di verifica) passo il valore alla memoria attiva in quel momento.
Per far attivare una memoria servono 3 condizioni: che il valore passato sia diverso da 0, che la memoria sia vuota e che la precedente sia piena.
Con questo meccanismo mi assicuro che i valori vengano salvati uno per volta e che le precedenti memorie non siano sovrascritte.

Rotore Singolo

Qui in base alle posizioni ruotate della rotella e alla lettera iniziale trovo la lettera finale sommando i 2 valori o, nel caso in cui la somma superi 7 (cosa indicativa che la rotella ha ri-superato la posizione iniziale) uso il bit di riporto per passare in output il risultato dell'operazione più 1 (Es: sono in posizione 6 mi passano la lettera 2 dovrebbe tornarmi 8, ovvero lettera 1. Quindi sommo i 3 bit dell'operazione che sono 000 ad 1 in 3 bit).

Rotori



Qui semplicemente collego i 3 rotori passando ad ognuno dalla memoria la sua posizione attuale. Nel caso l'input sia 0 non lo considero.

Memoria Posizioni Rotori

Qui salvo le posizioni attuali dei 3 rotori in 3 Counter.
Queste variano in modo diverso a seconda del settaggio, (input **Setting Concluso** collegato a 3 multiplexer).
Questi Counter vengono azzerati se si raggiunge la posizione 7 (ho 6 shift possibili quindi al 7 sono tornato a quella iniziale, sotto-circuito **Verifica Posizione Rotore**)
Col setting in corso le posizioni aumentano in base all'input da tastierino.
Concluso questo il primo rotore girerà ad ogni pressione di una lettera (che passo come Clock), gli altri 2 girano di una posizione solo ad ogni giro completo del rotore precedente (sempre verificato grazie a **Verifica Posizione Rotore**).
Per evitare casi particolari come lo scattare in avanti dei 2 rotori se il precedente già all'inizio era 0 ho dovuto aggiungere le 2 memorie "**Primo Scatto Fatto**" (Flip-Flop S-R) che diventano 1 solo dopo che il disco precedente ha fatto almeno uno spostamento (cosa che verifico usando il bit 0 che al primo spostamento diventa 1).

Queste memorie, messe in AND cogli input **Ritorno a 0** e **Tastiera Attivata**, attivano il Counter ed aumentano la posizione del Rotore.

Esempio di Funzionamento

Aprendo il file e cliccando su “Interfaccia Utente” comparirà la schermata principale.
Si abiliti la simulazione ed imposti il Clock a 8 Hz.
Cliccando sui tasti sotto “Posizioni Rotori” 1 volta su ognuno in alto comparirà 111.
Scelgo inoltre di sostituire le A con E cliccando una volta su entrambe le lettere, anch’esse saranno visualizzate sui led.
Clicco su Set Iniziale Concluso e si accende il led di conferma.
A questo punto provo a scrivere FALLO una lettera alla volta: le lettere in uscita comporranno la parola criptata in EOALI.
Da notare la particolarità che 2 lettere uguali, le 2 L, vengano criptate in modo diverso grazie all’effetto dei rotori.