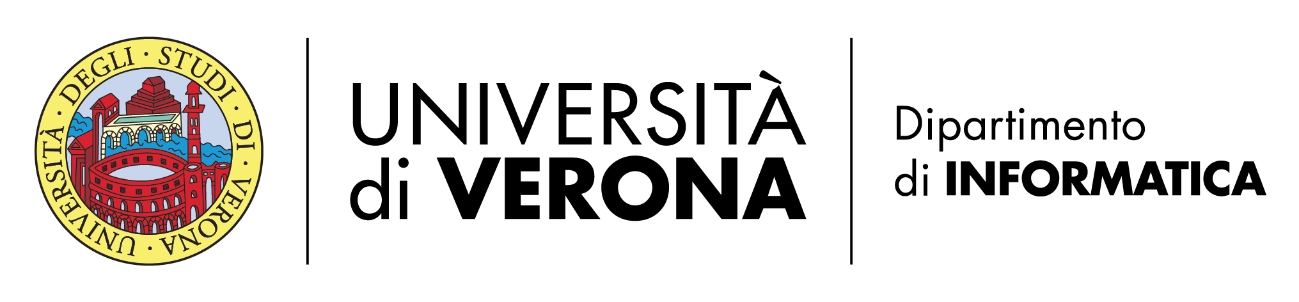
****

**Elaborato SIS**

**Laboratorio di Architettura degli Elaboratori**

Progettazione macchinario chimico

A.A. 2021/2022

*Filippo Barbieri (VR472408)*

*Alessio Brighenti (VR471509)*

Indice

[1. Specifiche 1](#_Toc96375029)

[2. Architettura generale del circuito 3](#_Toc96375030)

[3. FSM 4](#_Toc96375031)

[4. Datapath 5](#_Toc96375032)

[5. Statistiche del circuito 7](#_Toc96375033)

[5.1. Prima dell’ottimizzazione 7](#_Toc96375034)

[5.2. Dopo l’ottimizzazione 7](#_Toc96375035)

[6. Mapping: area e ritardo 8](#_Toc96375036)

[7. Scelte progettuali 9](#_Toc96375037)

# 1. Specifiche

Si progetti il circuito sequenziale che controlla un macchinario chimico il cui scopo è portare una soluzione iniziale a pH noto, ad un pH di neutralità. Il valore del pH viene espresso in valori compresi tra 0 e 14.

Il circuito controlla due valvole di erogazione: una di soluzione acida e una di soluzione basica.

Se la soluzione iniziale è acida, il circuito dovrà procedere all’erogazione della soluzione basica fintanto che la soluzione finale non raggiunga la soglia di neutralità (pH compreso tra 7 e 8).

Analogamente, se la soluzione iniziale è basica, il circuito procederà all’erogazione di soluzione acida fino al raggiungimento della soglia di neutralità.

Per pH acido si intende un valore strettamente inferiore a 7, mentre per basico si intende una soluzione con pH strettamente maggiore a 8.

Il pH viene codificato in fixed-point, con 4 bit riservati per la parte intera e gli altri per la parte decimale.

Le due valvole hanno flussi differenti di erogazione.

La valvola relativa alla soluzione basica eroga una quantità di soluzione che permette di alzare il pH della iniziale di 0.25 ogni ciclo di clock.

La valvola relativa alla soluzione acida eroga una quantità di soluzione che permette di abbassare il pH della soluzione iniziale di 0.5 ogni ciclo di clock.

Il circuito ha tre ingressi nel seguente ordine:

* RST (1 bit)
* START (1 bit)
* pH (8 bit, 4 parte intera e 4 per la parte decimale)

Gli output sono i seguenti e devono seguire il seguente ordine:

* FINE\_OPERAZIONE (1 bit)
* ERRORE\_SENSORE (1 bit)
* VALVOLA\_ACIDO (1 bit)
* VALVOLA\_BASICO (1 bit)
* PH\_FINALE (8 bit)
* NCLK (8 bit)

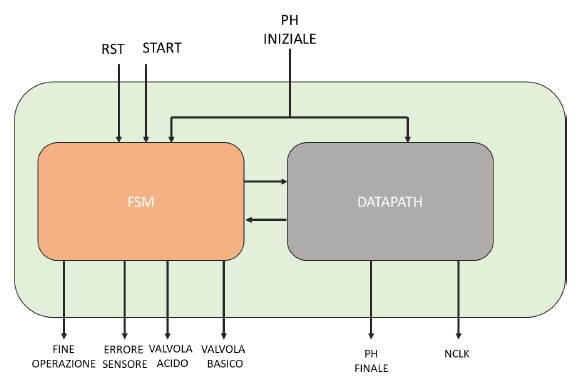
Input e output devono essere definiti nell’ordine sopra specificato (da sinistra verso destra).

Le porte con più bit devono essere descritte utilizzando la codifica con il bit più significativo a sinistra.

Il meccanismo è guidato come segue:

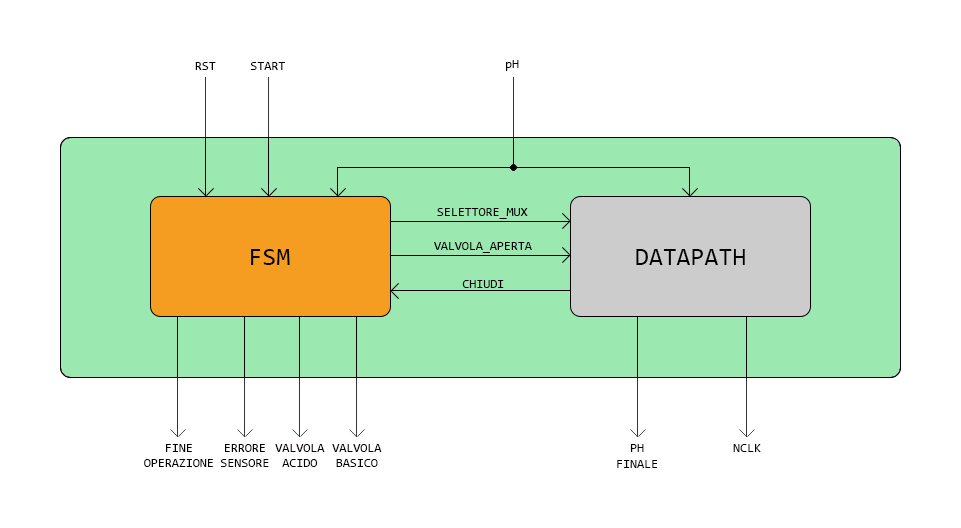
* Quando il segnale RST viene alzato, il sistema torna da un qualsiasi stato allo stato di Reset, mettendo tutte le porte in output a zero.
* Per procedere, Il sistema riceve in input il segnale di START, con valore 1, e il segnale del pH iniziale per un solo ciclo di clock. Il sistema potrà quindi procedere con la fase di elaborazione.
* Se la soluzione iniziale è acida, viene aperta la valvola della soluzione basica, mettendo a 1 il relativo output. Analogamente, se la soluzione iniziale è basica, viene aperta la valvola della soluzione acida mettendo a 1 la porta VALVOLA\_ACIDO.
* Il sistema mantiene aperte le valvole per il tempo necessario al raggiungimento della soglia di neutralità (calcolata dal sistema).
* Una volta terminata l’operazione, il sistema deve chiudere tutte le valvole aperte, riportare il pH finale sulla porta in output PH\_FINALE e alzare la porta di FINE\_OPERAZIONE.
* La porta NCLK riporta quanti cicli di clock sono stati necessari per portare la soluzione a neutralità.
* Se il valore del pH non è valido (> 14) il sistema deve riportare l’errore alzando l’output ERRORE\_SENSORE.

Lo schema generale del circuito deve rispettare la FSMD riportata di seguito:



* È possibile aggiungere degli ulteriori segnali interni per la comunicazione tra FSM e DATAPATH
* Le porte di input e di output devono rispettare l’ordine definito ed essere collegate al rispettivo sotto modulo
* Il DATAPATH deve essere unico: se volete definire più DATAPATH, questi devono essere inglobati in un unico modello.
* È compito della FSM identificare se il pH della soluzione iniziale sia acido o basico!

# 2. Architettura generale del circuito

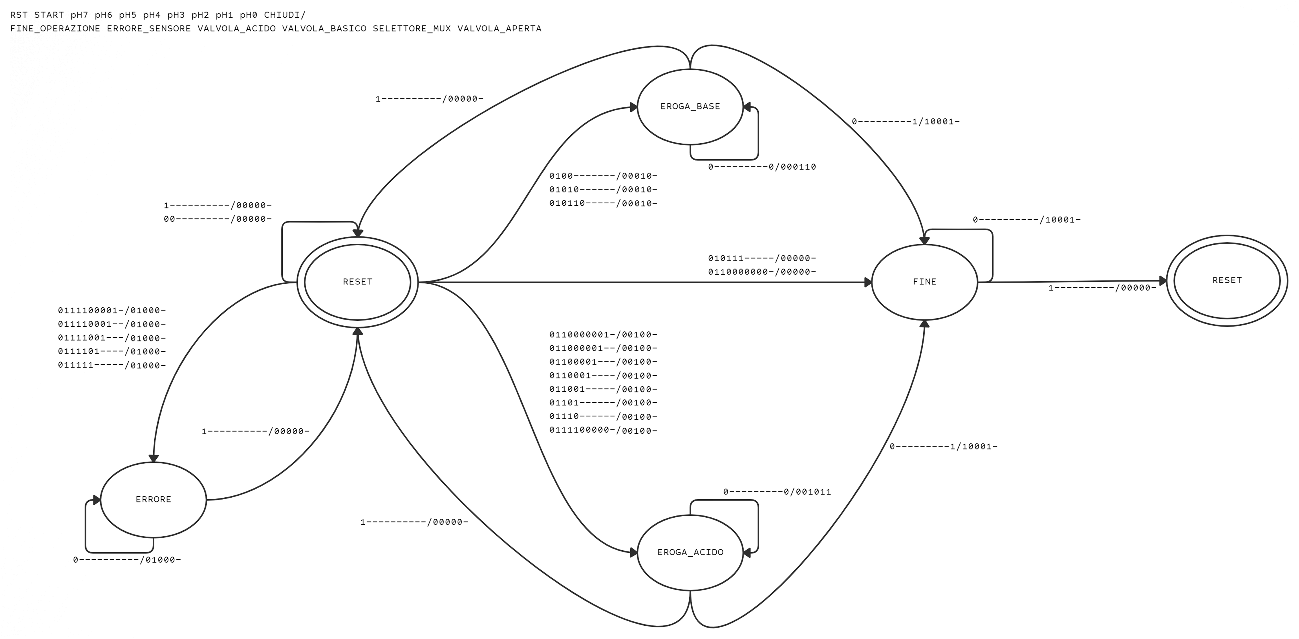


Il circuito si articola in due parti: una FSM, che funge da unità di controllo, e un datapath, che provvede al calcolo dei dati necessari.

Le due parti comunicano tra di loro tramite due segnali di controllo (SELETTORE\_MUX e VALVOLA\_APERTA) e un segnale di stato (CHIUDI).

I file che descrivono singolarmente FSM e datapath si trovano nella cartella “non\_ottimizzato” del progetto, rispettivamente con i nomi “fsm.blif” e “datapath.blif”.

# 3. FSM

****

Il controllore del macchinario chimico è una macchina a stati finiti di tipo Mealy, ovvero la cui funzione d’uscita dipende sia dallo stato corrente che dai valori in ingresso.

I segnali di input e output della macchina sono i seguenti:

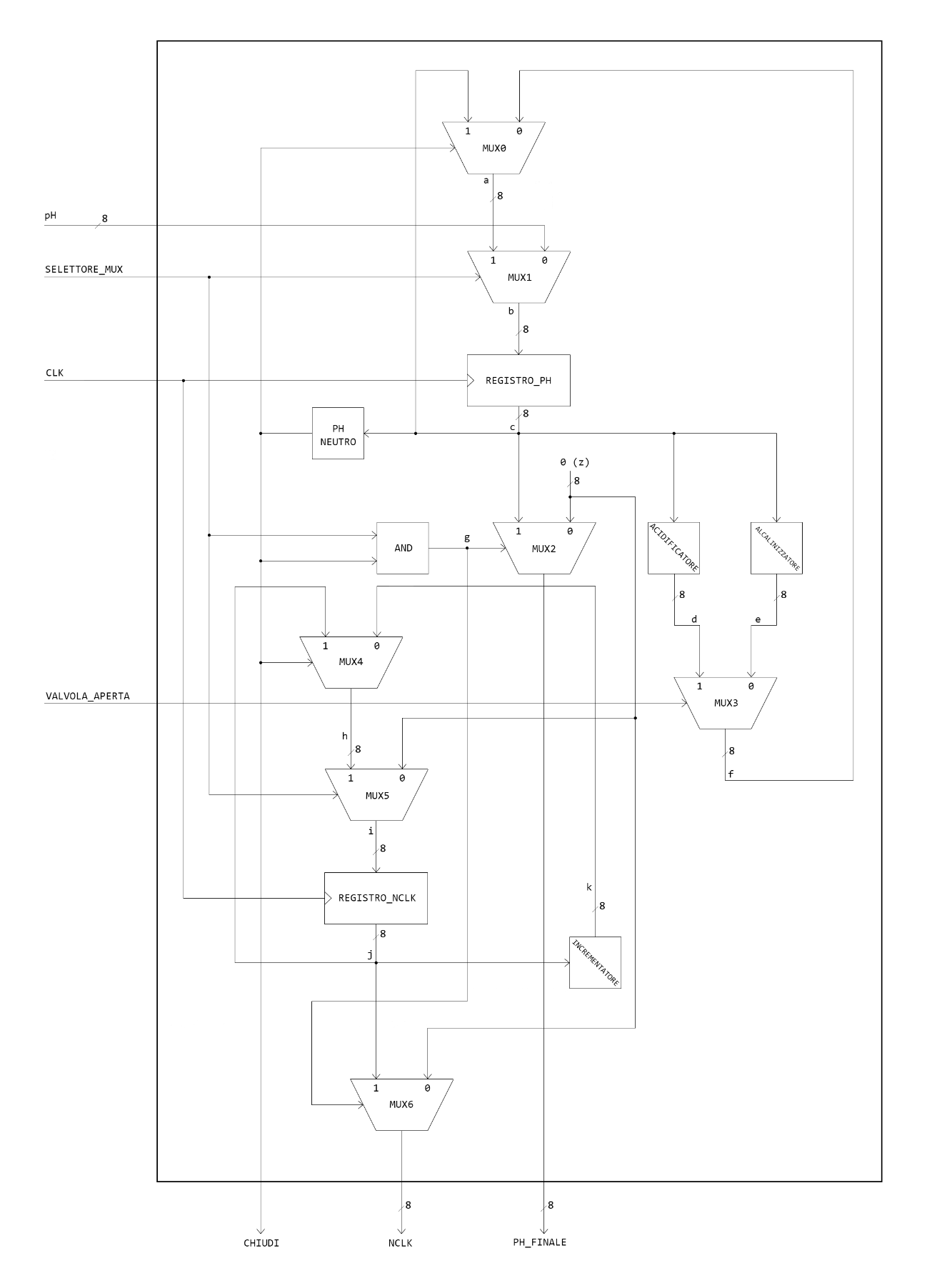
|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| RST [1]  START [1]  pH [8]  CHIUDI [1] | FINE\_OPERAZIONE [1]  ERRORE\_SENSORE [1]  VALVOLA\_ACIDO [1]  VALVOLA\_BASICO [1]  SELETTORE\_MUX [1]  VALVOLA\_APERTA [1] |

Gli stati della macchina sono cinque e rappresentati come segue:

* RESET (stato iniziale): tutti gli output sono a 0 e la macchina è pronta a ricevere un valore di pH in ingresso. La macchina rimane nello stesso stato quando RST è alzato, oppure quando RST è abbassato insieme a START. Quando START è a 1, la macchina passa in stato di ERRORE (alzando il relativo bit d’uscita) se viene inserito un pH non valido (superiore a 14), altrimenti può passare in uno dei due stati di erogazione (con corrispondente bit alzato) o direttamente nello stato di FINE se il pH inserito è già neutro.
* ERRORE: stato di errore causato stato in cui la macchina si trova dopo l’inserimento di un pH invalido e in cui permane, lasciando a 1 l’uscita ERRORE\_SENSORE, finché non viene alzato RST.
* EROGA\_BASICO: stato in cui la macchina si trova dopo l’inserimento di un pH acido e in cui rimane, lasciando a 1 l’uscita VALVOLA\_BASICO, finché dal datapath non arriva il segnale CHIUDI a 1. Quando CHIUDI è alzato, si passa nello stato di FINE alzando la relativa uscita. Se viene alzato RST, la macchina torna nello stato di RESET con tutti gli output a 0.
* EROGA\_ACIDO: stato in cui la macchina si trova dopo l’inserimento di un pH basico e in cui rimane, lasciando a 1 l’uscita VALVOLA\_ACIDO, finché dal datapath non arriva il segnale CHIUDI a 1. Quando CHIUDI è alzato, si passa nello stato di FINE alzando la relativa uscita. Se viene alzato RST, la macchina torna nello stato di RESET con tutti gli output a 0.
* FINE: stato in cui la macchina si trova dopo l’inserimento di un valore di pH neutro o dopo il raggiungimento dell’intervallo di neutralità a seguito dell’erogazione di soluzione acida/basica. La macchina rimane nello stato, con uscita FINE\_OPERAZIONE alzata, finché non viene alzato RST che la porta nello stato di RESET.

Quanto sopra descritto si può visivamente verificare nel grafo delle transizioni riportato. Le transizioni sono in totale 30, così ottenute dopo aver ridotto le combinazioni di ingresso nello stato di RESET utilizzando i don’t care.

# 4. Datapath



Il datapath ha i seguenti segnali di input e output:

|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| SELETTORE\_MUX [1]  VALVOLA\_APERTA [1]  pH [8] | CHIUDI [1]  PH\_FINALE [8]  NCLK [8] |

Il datapath si può considerare suddiviso in due sotto-circuiti: il primo per portare il pH a neutro e il secondo per contare i cicli di clock necessari a compiere l’operazione.

**Aggiornamento pH:**

Le componenti utilizzate per questa parte di circuito sono:

MULTIPLEXER 0: multiplexer regolato da un segnale di selezione proveniente dal blocco PH NEUTRO che: nel caso sia 1 mantiene il pH memorizzato nel registro fino a quando non si alza il segnale RESET, nel caso sia 0 fa passare il pH appena aggiornato.

MULTIPLEXER 1: multiplexer regolato da un segnale di selezione proveniente dalla FSM (SELETTORE\_MULTIPLEXER) che indica se considerare il pH iniziale (primo ciclo di clock) oppure il pH aggiornato.

REGISTRO\_PH: è un registro nella quale viene memorizzato ed aggiornato il pH ad ogni ciclo di clock.

PH\_NEUTRO: componente che ha come segnale di uscita 1 se il pH ricevuto come segnale di ingresso è compreso tra 0110.1111 e 1000.0000 (compreso).

MULTIPLEXER 2: multiplexer regolato da un segnale di selezione, risultato dell’AND tra il segnale di uscita della componente PH\_NEUTRO e il segnale SELETTORE\_MUX, che se vale 1 porta il contenuto di REGISTRO\_PH nell’uscita PH\_FINALE.

ACIDIFICATORE: ha come segnale di entrata il pH proveniente da REGISTRO\_PH e svolge l’operazione di somma (complemento a 2) pH+1111.1000.

ALCALINIZZATORE: ha come segnale di entrata il pH proveniente da REGISTRO\_PH e svolge l’operazione di somma (complemento a 2) pH+0000.0100.

MULTIPLEXER 3: multiplexer regolato da un segnale di selezione proveniente dalla FSM (VALVOLA\_APERTA) che indica quale dei due pH aggiornati è da tenere in considerazione.

**Conteggio cicli di clock:**

Le componenti utilizzate per questa parte di circuito sono:

MULTIPLEXER 4: multiplexer regolato da un segnale di selezione proveniente dalla componente PH NEUTRO che nel caso sia 1 mantiene lo stesso conteggio dei cicli di clock nel registro, se è 0 fa passare il conteggio incrementato.

MULTIPLEXER 5: multiplexer regolato da un segnale di selezione proveniente dalla FSM (SELETTORE\_MULTIPLEXER) che indica se considerare il conteggio precedente oppure azzerare il registro.

REGISTRO\_NCLK: registro che memorizza il conteggio dei cicli di clock impiegati per l’operazione.

INCREMENTATORE: componente che incrementa il numero di cicli di clock.

MULTIPLEXER 6: multiplexer regolato da un segnale di selezione, risultato dell’AND tra il segnale di uscita della componente PH\_NEUTRO e il segnale SELETTORE\_MUX, che se vale 1 porta il contenuto di REGISTRO\_NCKL nell’uscita NCKL.

# 5. Statistiche del circuito

## 5.1. Prima dell’ottimizzazione

|  |  |
| --- | --- |
| **DATAPATH** | |
| NODI | LETTERALI |
| *200* | *717* |
| **FSM** | |
| NODI | LETTERALI |
| *9* | *229* |
| **FSMD** | |
| NODI | LETTERALI |
| *56* | *253* |

## 5.2. Dopo l’ottimizzazione

|  |  |  |
| --- | --- | --- |
| **DATAPATH** | | |
|  | NODI | LETTERALI |
| *script.rugged* | *46* | *211* |
| *Miglior minimizzazione trovata* | *44 (-78.00%)* | *201 (-71.97%)* |
| **FSM** | | |
|  | NODI | LETTERALI |
| *script.rugged* | *11* | *61* |
| *Miglior minimizzazione trovata* | *11 (+22.22%)* | *51 (-77.73%)* |
| **FSMD** | | |
|  | NODI | LETTERALI |
| *script.rugged* | *54* | *254* |
| *Miglior minimizzazione trovata* | *61 (+08.92%)* | *245 (-03.16%)* |

# 6. Mapping: area e ritardo

Statistiche del circuito mappato con la libreria **Synch.genlib**:

|  |  |
| --- | --- |
| **Total Area** | *5656.00* |
| **Gate Count** | *157* |
| **Buffer Count** | *19* |
| **Inverter Count** | *27* |
| **Most Negative Slack** | *-29.60* |
| **Sum of Negative Slacks** | *-601.00* |
| **Number of Critical PO** | *39* |

# 7. Scelte progettuali

Significato segnali intermedi