Prova Finale Progetto Reti Logiche

**Anno Accademico 2019-2020**

Giusti Leonardo (Codice Persona: 10633778)

La Manna Giuseppe (Codice Persona: 10608466)

**INDICE**

1. **Introduzione . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2**

1. **Architettura . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3**

**2.1. *Schema progettuale***

***2.1.1. Interfaccia del componente***

***2.1.2 Segnali utilizzati***

***2.1.3 Macchina a Stati Finiti***

**2.2. *Scelte progettuali***

1. **Risultati di test funzionali . . . . . . . . . . . . . . . . . . . . . . . . . . 4**
2. **Risultati test non funzionali . . . . . . . . . . . . . . . . . . . . . . . . . 5**

1. **Risultati della sintesi . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6**

**1. Introduzione**

**L’obbiettivo del progetto è creare un metodo di codifica Working Zone pensato per trasformare il valore di un indirizzo trasmesso, se questo appartiene a certi intervalli denominati *working-zone.***

**Il codice si occuperà di leggere l’indirizzo da codificare (ADDR) e gli 8 indirizzi base delle Working Zone, aventi dimensione di 4 indirizzi incluso quello base, confrontarli e produrre l’indirizzo opportunamente codificato.**

**Se l’indirizzo da trasmettere non appartiene a nessuna Working Zone, ai bit di indirizzamento verrà concatenato un bit addizionale WZ\_BIT messo a 0:**

****WZ(ADDR) = WZ\_BIT & ADDR****

Se l’indirizzo da trasmettere appartiene ad una Working Zone, verrà trasmesso WZ\_BIT posto a 1 concatenato ad WZ\_NUM, che rappresenta la codifica in binario del numero della working-zone al quale l’indirizzo appartiene, e WZ\_OFFSET, che rappresenta lo spiazzamento in codifica one-hot :

****WZ(ADDR) = WZ\_BIT & WZ\_NUM & WZ\_OFFSET****

Nel seguente esempio viene approfondito il metodo di codifica:

* *Indirizzo Base WZ 0*: 7
* *Indirizzo Base WZ 1*: 14
* *Indirizzo Base WZ 2*: 21
* *Indirizzo Base WZ 3*: 28
* *Indirizzo Base WZ 4*: 35
* *Indirizzo Base WZ 5*: 42
* *Indirizzo Base WZ 6*: 49
* *Indirizzo Base WZ 7*: 56
* *Indirizzo Base WZ 8*: 63
* *Address1* da codificare: 78 (01001110)
* *Address2* da codificare: 22 (00010110)

**L’*Address1* non appartiene a nessuna Working Zone, quindi avrà WZ\_BIT posto a 0 mentre i successivi bit rimarranno invariati.**

**Verrà codificato come (0 - 1001110), che corrisponde a 78.**

**L’*Address2* appartiene alla Working Zone 2 con WZ\_OFFSET = 1. WZ\_NUM corrisponde alla codifica binaria di 2 su 3 bit (010) invece il WZ\_OFFSET sarà la codifica one hot di 1 su 4 bit (0010).**

**Il valore codificato avrà come output (1 – 010 – 0010), che corrisponde al valore 162.**

**2. Architettura**

**2.2 Schema progettuale**

**2.2.1. Interfaccia del componente**

**L’interfaccia del componente descritto è la seguente:**

*entity* ***progetto\_reti\_logiche*** *is*

*port(*

***i\_clk*** *: in std\_logic;*

***i\_start*** *: in std\_logic;*

***i\_rst*** *: in std\_logic;*

***i\_data*** *: in std\_logic\_vector ( 7 downto 0 );*

***o\_address*** *: out std\_logic\_vector ( 15 downto 0 );*

***o\_done*** *: out std\_logic ;*

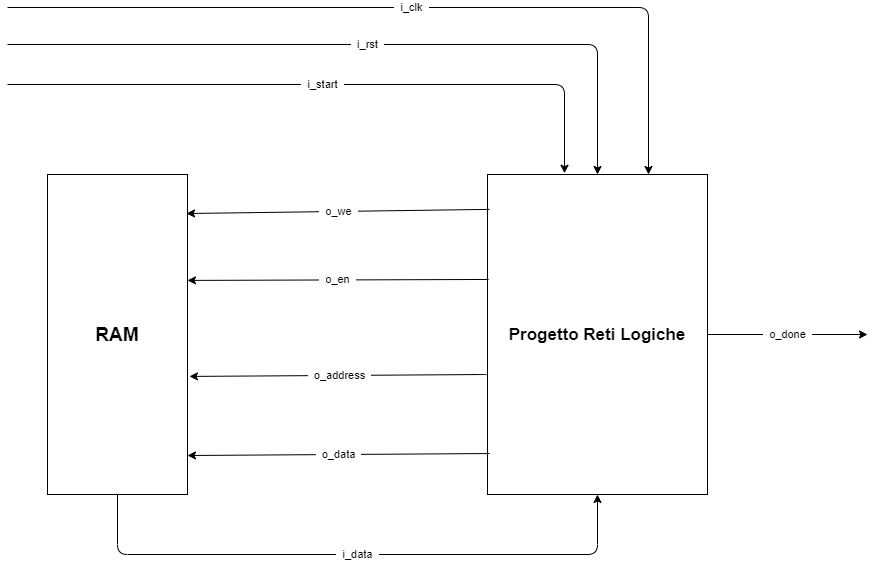
***o\_en*** *: out std\_logic ;*

***o\_we*** *: out std\_logic ;*

***o\_data*** *: out std\_logic\_vector ( 7 downto 0 )*

*);*

*end entity;*



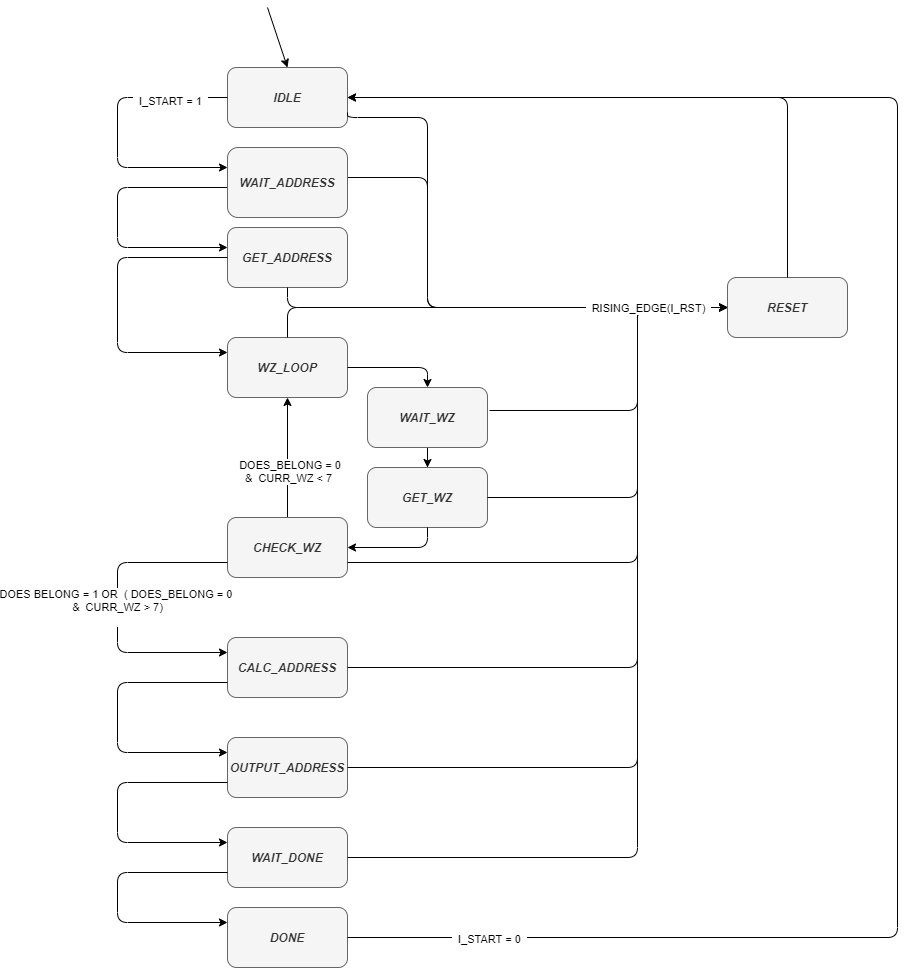
**2.2.2. Segnali utilizzati**

**La seguente tabella riassuntiva rappresenta i segnali introdotti per la realizzazione del modulo gestiti all’interno degli stati della FSM. Per ogni segnale presente in tabella ne è stato inserito uno aggiuntivo (Nome\_Segnale\_next) nel quale viene memorizzato il valore del segnale all’istate di clock successivo.**

|  |  |
| --- | --- |
| ****Nome Segnale**** | ****Contenuto**** |
| **Current\_State** | **Memorizza lo stato corrente della FSM** |
| **Current\_Working\_Zone** | **Memorizza l’indirizzo base della Working Zone** |
| **Read\_Address** | Memorizza temporaneamente il valore dell’indirizzo ricevuto da codificare |
| **Does\_Belong** | **Specifica se il valore appartiene o meno alla Working Zone** |
| **Working\_Zone\_Num** | **Contiene il numero della Working Zone cui si riferisce l’address ricevuto da codificare** |
| **Coded\_Address** | Contiene l’indirizzo finale da restituire |

**2.2.3. Macchina a Stati Finiti**

Il funzionamento alla base del componente è stato implementato attraverso una FSM che usa *i\_start* come segnale di avvio della codifica e *i\_rst* come segnale per l’inizializzazione della macchina. Il passaggio tra gli stati successivi sarà approfondito in seguito alla figura dello schema.



* **Idle: s**tato iniziale in cui la macchina attende un segnale di *start* per iniziare la sua computazione: una volta ricevuto la macchina richiede l'*address* da codificare alla RAM;
* **Wait\_Address:** stato in cui attendo che la RAM presenti in *i\_data* l'*address* da codificare;
* **Wz\_Loop:** stato in cui richiedo alla RAM l'address base della Working Zone corrente, inizialmente carico WZ 0;
* **Get\_Wz:** stato in cui salvo l'address base della Working Zone corrente presentatomi dalla RAM su *i\_data;*
* **Check\_Wz:** stato in cui verifico se l'*address* da codificare appartiene alla Working Zone corrente o meno: se appartiene mi preparo a codificarlo, in caso contrario passo alla Working Zone successiva; se sono all'ultima Working Zone allora l'*address* non appartiene ad alcuna di queste e mi preparo a codificarlo di conseguenza;
* **Calc\_address:** stato in cui calcolo l'*address* codificato tenendo in considerazione il numero della Working Zone di appartenenza e l'offset dell'*address* da codificare;
* **Output\_Address:** stato in cui richiedo alla RAM di poter scrivere l'*address* codificato;
* **Wait\_Done:** stato in cui attendo che l’*address* codificato sia stato effettivamente scritto in memoria;
* **Done:** stato di completamento in cui attendo che il segnale di *i\_start* scenda a 0 per tornare in Idle e poter continuare a codificare l'*address* successivo;
* **Reset:** stato in cui azzero il valore di ogni segnale del componente in seguito ad una richiesta di *i\_rst*, per tornare poi allo stato di Idle e aspettare la codifica successiva;

**3. Scelte progettuali**

**4. Risultati di test funzionali**

**5. Risultati di test non funzionali**

**6. Risultati della sintesi**