

RELAZIONE DELL'ELABORATO SIS DI ARCHITETTURA DEGLI ELABORATORI

**SIS - ELABORATO:
RILEVAZIONE FORATURA PNEUMATICI**

AA 2013/2014

Andrea Bazerla	VR377556
Doru Lucian Terenteaca	VR378446

INDICE

DESCRIZIONE DEL CIRCUITO DA REALIZZARE	3
STG DELLA FSM	4
DESCRIZIONE DELLA FSM	5
ARCHITETTURA DEL DATAPATH	6
DESCRIZIONE DEL DATAPATH	7
STATISTICHE DEL CIRCUITO	8
SCELTE PROGETTUALI	9

DESCRIZIONE DEL CIRCUITO DA REALIZZARE

Si progetti un dispositivo per la rilevazione di foratura pneumatici di un'automobile. Il dispositivo dovrà essere modellato come circuito sequenziale **FSM+Datapath** in **SIS**. Si consideri che un sistema elettromeccanico rileva le rotazioni dei quattro pneumatici e manda agli ingressi del dispositivo un valore positivo 1 ad ogni giro completo del pneumatico, per ogni pneumatico. Nel caso di una determinata differenza del numero di giri di questi dovuto a foratura, in una distanza di 100 metri, il dispositivo deve accendere una spia di segnalazione foratura sul cruscotto.

Il circuito dovrà avere i seguenti ingressi e uscite:

STEERING[1bit]: se uguale a 1, indica che l'automobile è in fase di sterzo e, di conseguenza, il dispositivo non deve azionare alcun controllo, deve resettare qualsiasi conteggio numero giri e confronti pneumatici.

RESET[1bit]: se uguale a 1, il dispositivo deve resettarsi completamente.

SPEED[8bit]: indica la velocità istantanea (in Km/h) dell'automobile, da 0 a max 255.

Quando è maggiore di 10 Km/h deve far partire il conteggio delle rotazioni dei pneumatici. Altrimenti lascia il dispositivo in uno stato di **PARK**.

ROTDXF[1bit], **ROTSXF**[1bit], **ROTDXR**[1bit], **ROTSXR**[1bit]: se uguale a 1, ogni ingresso indica una completa rotazione dei pneumatici destro anteriore, sinistro anteriore, destro posteriore, sinistro posteriore, rispettivamente. Altrimenti, l'ingresso vale 0.

LIGHT[1bit]: deve essere impostato a uno per accendere la spia di segnalazione foratura sul cruscotto..

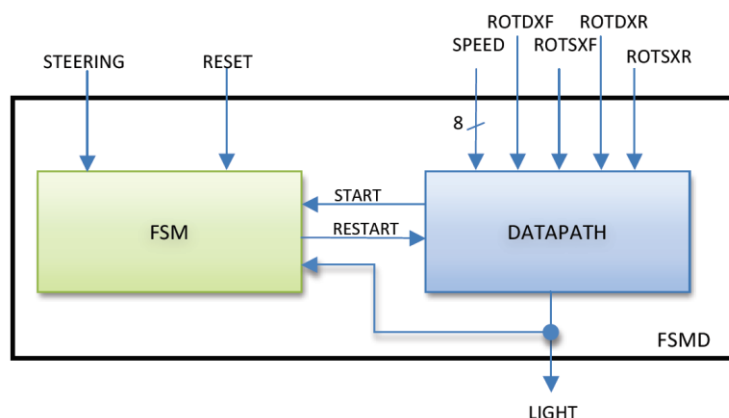
Il controllore è collegato al datapath con tre segnali che hanno il seguente significato:

START[1bit]: è messo a uno se la velocità è maggiore di 10 Km/h.

RESTART[1bit]: è messo a uno se, con una velocità istantanea maggiore a 10 Km/h, l'automobilista aziona lo sterzo. Deve quindi resettare e far ripartire tutti i conteggi giri pneumatici precedenti.

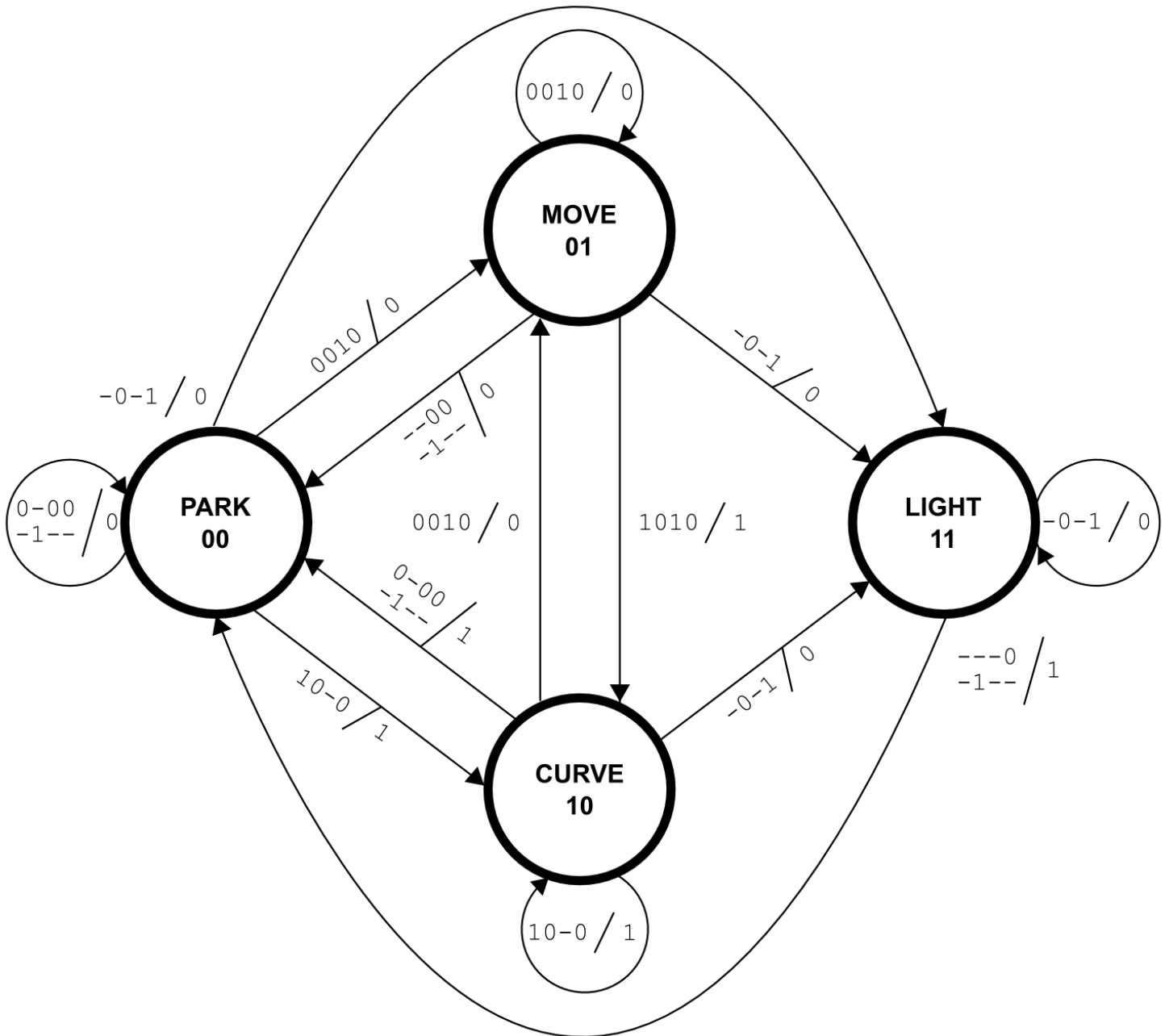
LIGHT[1bit]: porta in ingresso al controllore l'uscita del dispositivo **LIGHT**. Quando **LIGHT** vale 1, il controllore deve portarsi in uno stato (**LIGHT**) fino al reset completo del dispositivo.

Il dispositivo deve contare il numero di giri di ogni pneumatico e confrontare sia i due anteriori che i due posteriori. Il confronto viene effettuato ogni 100 metri e se la differenza in almeno un confronto supera il 20%, il dispositivo deve alzare il segnale di uscita **LIGHT** (e portarsi sullo stato di **LIGHT**). Ogni 100 metri i contatori devono essere resettati. Si consideri che la distanza (100 metri) viene rilevata prendendo come campione la ruota anteriore destra, per la quale ad ogni rotazione corrisponde una distanza di 2 metri.



STG DELLA FSM

STEERING RESET START LIGHT / RESTART



DESCRIZIONE DELLA FSM

Nella progettazione della FSM del circuito abbiamo utilizzato i seguenti ingressi e uscite come richiesto:

STEERING RESET START LIGHT / RESTART

Sulla base dei 4 bit di ingresso della FSM, abbiamo costruito il grafo delle transizioni formato da 4 stati:

PARK[00]: il circuito rimane in questo stato finché la velocità dell'automobile non supera i 10 Km/h. Nel momento in cui viene superata tale velocità, la macchina esce da questo stato.

MOVE[01]: si passa in questo stato se l'automobile supera la velocità dei 10 Km/h e solo se si sta percorrendo un rettilineo (Quindi senza sterzare).

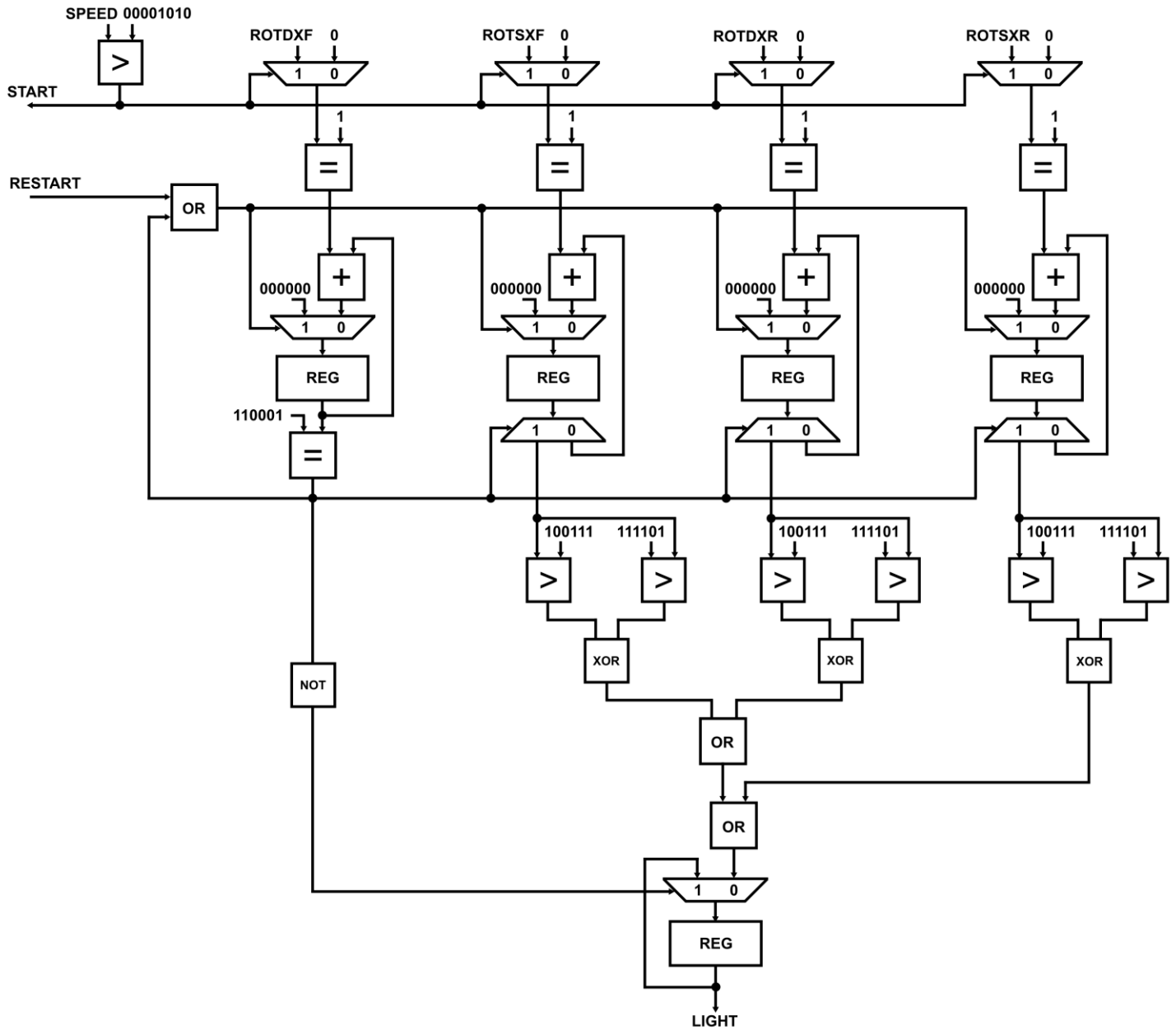
CURVE[10]: quando l'automobile inizia a sterzare ci si trova in questo stato.

LIGHT[11]: si passa in questo stato quando la spia di foratura dei pneumatici si accende.

INPUTS = { STEERING, RESET, START, LIGHT }

OUTPUT = { RESTART }

ARCHITETTURA DEL DATAPATH



DESCRIZIONE DEL DATAPATH

Il datapath che abbiamo progettato presenta 6 inputs e 2 outputs:

INPUTS = { SPEED[8bit], ROTDXF[1bit], ROT SXF[1bit], ROTDXR[1bit],
ROTSXR[1bit], RESTART[1bit] }

OUTPUTS = { START[1bit], LIGHT[1bit] }

- Inizialmente abbiamo un maggioratore che verifica se la velocità è superiore ai **10km/h** e, in questo caso, vengono attivati i 4 multiplexer delle ruote.
- Nel momento in cui vengono attivati i multiplexer, ogni giro di ruota dà come input il valore **1**, il quale viene mandato in un comparatore e successivamente in un sommatore. Questo ha il compito di sommare il giro attuale di ruota con eventuali altri giri precedenti.
- Andando avanti seguono altri 4 multiplexer, uno per ogni ruota, i quali permettono il salvataggio del numero di giri di ruote nel registro che li segue, oppure resettano il registro nel momento in cui il loro **CS** (Chip Select) viene portato ad 1. Questo succede se viene attivato il **RESTART**, oppure se sono stati percorsi **100 metri** corrispondenti a **50 giri** di ruota.
- Per quanto riguarda la ruota di riferimento, **ROTDXF**, dopo il registro abbiamo un comparatore che ha il compito di verificare se sono stati compiuti 50 giri oppure no.
- Invece, per quanto riguarda le altre ruote, dopo il registro abbiamo un demultiplexer, il quale viene attivato quando il comparatore della ruota di riferimento indica che sono stati compiuti 50 giri (100 metri), permettendo così il passaggio del numero di giri di ruote dal registro alla parte di circuito dedicata al confronto tra i giri delle varie ruote.
- Per ognuna delle 3 ruote rimanenti vengono utilizzati 2 maggioratori che si assicurano che il numero dei giri di ruota sia compreso tra **40** e **60** (Dato che ci conferma di non aver superato del **20%** la differenza tra i giri delle ruote).
- Infine, dopo aver utilizzato porte **XOR** e **OR**, abbiamo un registro la cui uscita viene collegata a **LIGHT**. Il registro è stato inserito per mantenere il valore di LIGHT ad 1 fino a che non viene resettato il circuito.

STATISTICHE DEL CIRCUITO

Una volta costruito il circuito, le statistiche che abbiamo ottenuto con il comando **"print_stats"** sono le seguenti:

pi=14 po=1 nodes=215 latches=27 lits(sop)=1009

Poi abbiamo minimizzato il circuito con il file **"script.rugged"** ottenendo una riduzione del numero di letterali e di nodi:

pi=14 po=1 nodes=52 latches=27 lits(sop)=299

Infine abbiamo mappato il progetto sulla libreria **"synch.genlib"**, ottimizzando per area e ottenendo come risultato un aumento di letterali e di nodi:

pi=14 po=1 nodes=170 latches=27 lits(sop)=395

In conclusione riportiamo in modo esteso i risultati ottenuti con il comando **map -m 0 -s** dopo aver mappato il circuito sulla libreria **"synch.genlib"**:

```
>>> before removing serial inverters <<<
# of outputs:      28
total gate area:    6856.00
maximum arrival time: (30.20,30.20)
maximum po slack:   (-0.80,-0.80)
minimum po slack:   (-30.20,-30.20)
total neg slack:    (-645.40,-645.40)
# of failing outputs: 28
>>> before removing parallel inverters <<<
# of outputs:      28
total gate area:    6856.00
maximum arrival time: (30.20,30.20)
maximum po slack:   (-0.80,-0.80)
minimum po slack:   (-30.20,-30.20)
total neg slack:    (-645.40,-645.40)
# of failing outputs: 28
# of outputs:      28
total gate area:    6472.00
maximum arrival time: (29.20,29.20)
maximum po slack:   (-0.80,-0.80)
minimum po slack:   (-29.20,-29.20)
total neg slack:    (-598.60,-598.60)
# of failing outputs: 28
```


SCELTE PROGETTUALI

- Siccome le specifiche progettuali del circuito relative al confronto tra le ruote non erano definite in modo chiaro, abbiamo deciso di confrontare i giri della ruota di riferimento anteriore destra **ROTDXF** con ognuna delle altre ruote, eseguendo così 3 confronti ogni **100 metri**.
- Anche se il circuito confronta le 3 ruote rimanenti con quella di riferimento, esso è progettato in tal modo da segnalare la foratura di pneumatici anche quando è la stessa ruota di riferimento ad essere bucata.
- I registri del circuito che si occupano di memorizzare il conteggio dei giri di ruota vengono resettati solamente quando sono stati compiuti 100 metri oppure quando è stato attivato il **RESTART**. Nel caso in cui la velocità scenda sotto i **10 km/h** in un qualsiasi momento della guida, il conteggio dei giri viene semplicemente sospeso.
- Nel circuito abbiamo utilizzato vari componenti a **6 bit**, invece di **8 bit**, visto che non erano necessari bit aggiuntivi.
- Per quanto riguarda l'**FSM**, ogni volta che viene attivato il **RESET**, viene alzato ad 1 anche il **RESTART**, permettendo così di resettare tutti i conteggi di giri di ruota.