

## 0.1 Simulazione e sintesi

### 0.1.1 Simulazione

Per tale componente è stata effettuata una simulazione behavioural, durante la quale sono stati fatti variare i due operandi e il riporto in ingresso. I risultati ottenuti sono osservabili in fig.1.

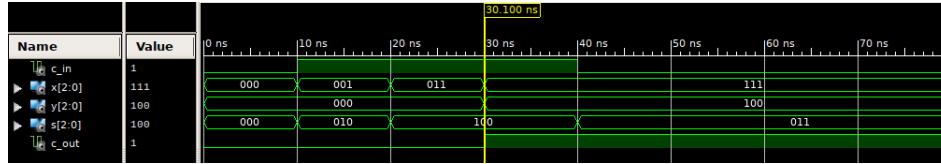


Figure 1: Simulazione behavioural del Carry Look Ahead.

### 0.1.2 Sintesi

Si è proceduto infine alla sintesi del componente utilizzando diversi valori di lunghezza delle stringhe tramite manipolazione del parametro generico *width*. Come nel caso del Ripple Carry Adder, sono stati ottenuti il *numero di slices* e *minimum period* (reciproco della massima frequenza di funzionamento) in funzione del numero *n* di bit per valutare le prestazioni di tale macchina. I risultati sono riportati in fig.3.

Si osservi come, rispetto ai risultati ottenuti con il Ripple Carry Adder, il circuito occupi un numero di slices (e dunque area) sensibilmente maggiore. Ciò è dovuto al fatto che, introducendo il calcolo dei riporti in parallelo, il numero di componenti è aumentato: tuttavia, tale differenza è apprezzabile solo con numero più elevato di bit (dai 32 in poi). Per quanto riguarda i ritardi, invece, non si registrano particolari differenze rispetto al caso RCA: i periodi minimi risultano pressoché identici. Possiamo dunque concludere che, con l'utilizzo del tool

```
[group style=group size=2 by 1, horizontal sep=2cm, yticklabel style=font=,
xticklabel style=font=] [legend style=font=, anchor=north, at=(0.70,0.16),
xmin=0,xmax=128, ymin = 0, ymax = 900, grid=major, width=0.45
height=, xlabel= Numero di bit, ylabel=Numero di slice] coordinates (0,0) (4,
19) (8, 42) (16, 97) (32, 230) (64,432) (128, 849) ; [legend style=anchor=north,
at=(0.50,0.95), xmin=0,xmax=128, ymin = 0, ymax = 5, grid=major,
width=0.45height=, xlabel= Numero di bit, ylabel=Minimum period (ns)]
coordinates (0,0) (4, 1.429) (8, 2.054) (16, 2.480) (32, 2.935) (64, 3.621) (128,
4.134) ;
```

Figure 2: Grafici dei risultati ottenuti post-sintesi in funzione del numero di bit.

```
[group style=group size=2 by 1,horizontal sep=2cm, yticklabel style=font=,
xticklabel style=font=] [legend style=font= , anchor=north, at=(0.70,0.16),
xmin=0,xmax=128, ymin = 0, ymax = 900, grid=major, width=0.45
height=,xlabel= Numero di bit, ylabel=Numero di slice] coordinates (0,0) (4,
23) (8, 47) (16, 97) (32, 230) (64,432) (128, 849) ; coordinates (0,0) (4, 19) (8,
42) (16, 97) (32, 230) (64,432) (128, 849) ; [legend style=anchor=north,
at=(0.50,0.95), xmin=0,xmax=128, ymin = 0, ymax = 5, grid=major,
width=0.45height=, xlabel= Numero di bit, ylabel=Minimum period (ns)]
coordinates (0,0) (4, 1.429) (8, 4.768) (16, 2.480) (32, 2.935) (64, 3.621) (128,
4.134) ; coordinates (0,0) (4, 1.429) (8, 2.054) (16, 2.480) (32, 2.935) (64,
3.621) (128, 4.134) ;
```

Figure 3: Grafici dei risultati ottenuti post-sintesi in funzione del numero di bit.

di sintesi per board Nexys 4, non sono particolarmente apprezzabili le differenze tra l'utilizzo di un RCA e di un CLA grazie alle capacità di ottimizzazione del suddetto tool.

### 0.1.3 Approfondimento: confronto RCA e CLA su board Nexys 2

Per analizzare più approfonditamente le differenze tra queste due macchine, si riportano in fig.3 i risultati ottenuti tramite sintesi su board Nexys 2.