

File PostScript ed EPS .....	809
Rappresentazione di multiplatori, demultiplatori e codificatori	809
Clock .....	811

Tkgate<sup>1</sup> (<http://www.tkgate.org>) è un programma per il disegno assistito (CAD) di circuiti logici, con la possibilità di utilizzare un simulatore, per il quale è possibile costruire anche dei moduli in grado di interfacciarsi con il sistema operativo ospitante.

La versione più recente di Tkgate è incompleta; tuttavia è migliore rispetto a quella ritenuta stabile. In questa sezione si annotano solo alcuni accorgimenti per poter utilizzare proficuamente la versione 2.0-b10 (beta) di Tkgate, nonostante i piccoli difetti che presenta. Va comunque osservato che la versione 2.0-**bn** di Tkgate va compilata personalmente, dato che difficilmente si può trovare un pacchetto pronto per la propria distribuzione; per la compilazione occorre installare nel sistema i sorgenti delle librerie Tcl/Tk 8.5.

### File PostScript ed EPS

I disegni realizzati attraverso Tkgate possono essere «stampati» producendo file PostScript o EPS (a seconda delle opzioni selezionate in fase di stampa). I file PostScript o EPS generati da Tkgate 2.0-b10 hanno un difetto importante che impedisce loro di essere gestiti da Ghostscript. Si osservi l'estratto seguente di codice PostScript/EPS:

```

/Courier-Latin1 /Courier findfont defLatin1
/Courier-Bold-Latin1 /Courier-Bold findfont defLatin1
/Courier-Italic-Latin1 /Courier-Italic findfont defLatin1
/Courier-BoldItalic-Latin1 /Courier-BoldItalic findfont defLatin1
/Helvetica-Latin1 /Helvetica findfont defLatin1
/Helvetica-Bold-Latin1 /Helvetica-Bold findfont defLatin1
/Helvetica-Oblique-Latin1 /Helvetica-Oblique findfont defLatin1
```

Le righe evidenziate in nero provocano un errore irreversibile per Ghostscript, ma è sufficiente commentarle senza per questo impoverire il risultato tipografico del file:

```

/Courier-Latin1 /Courier findfont defLatin1
/Courier-Bold-Latin1 /Courier-Bold findfont defLatin1
% /Courier-Italic-Latin1 /Courier-Italic findfont defLatin1
% /Courier-BoldItalic-Latin1 /Courier-BoldItalic findfont defLatin1
/Helvetica-Latin1 /Helvetica findfont defLatin1
/Helvetica-Bold-Latin1 /Helvetica-Bold findfont defLatin1
/Helvetica-Oblique-Latin1 /Helvetica-Oblique findfont defLatin1
```

Per correggere questi file con l'aiuto di uno script, si può usare SED (sezione 23.5) con un comando come quello seguente:

```
$ cat file.ps ↵
↵ | sed "s/^\(Courier-Italic/\%\)Courier-Italic/g" ↵
↵ | sed "s/^\(Courier-BoldItalic/\%\)Courier-BoldItalic/g" ↵
↵ > fix.ps [Invio]
```

### Rappresentazione di multiplatori, demultiplatori e codificatori

I componenti del tipo multiplatori e simili, hanno dei terminali numerati, per sapere in che ordine sono disposti. Tkgate consente di invertire l'ordine, rispetto a quello predefinito (da sinistra a destra); tuttavia, nella rappresentazione stampata (file PostScript o EPS), questi componenti vengono mostrati sempre come se l'ordine dei terminali fosse quello predefinito. Va osservato che questo problema esiste comunque anche nella versione stabile di Tkgate.

Per risolvere il problema della stampa, conviene annotare con dei commenti l'ordine dei terminali del multiplatore e degli altri componenti simili, poi è opportuno sopprimere il codice PostScript responsabile di questo errore.

Nel file PostScript o EPS, ogni componente ha una sua «funzione», dichiarata nel preambolo. L'estratto seguente mostra le funzioni di multiplatore e componenti analoghi, dove è stato commentato

il codice responsabile della rappresentazione errata dell'ordine dei terminali:

```

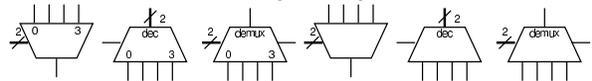
/mux {
  dup /mrot exch def
  startgate
  8 rfont
  -29.5 15.5 moveto
  29.5 15.5 lineto
  16.5 -12.5 lineto
  -16.5 -12.5 lineto
  closepath stroke
  dup
  1 add 58 exch div
  2 copy mul
  mrot -90 eq mrot -180 eq or {
  % 3 -1 roll 1 sub 50 string cvs exch (0) exch % d1 (n) (0) dn
  % -29 add 7 rCT % d1
  % exch -29 add 7 rCT
  } {
  % 3 -1 roll 1 sub 50 string cvs exch % d1 (n) dn
  % -29 add 7 rCT % d1
  % (0) exch -29 add 7 rCT
  } ifelse
  grestore
} def
...
/demux {
  startgate
  8 rfont
  (demux) 0 5 rCT
  -16.5 12.5 moveto
  16.5 12.5 lineto
  29.5 -15.5 lineto
  -29.5 -15.5 lineto
  closepath stroke
  dup
  1 add 58 exch div
  2 copy mul
  % 3 -1 roll 1 sub 50 string cvs exch % d1 (n) dn
  % -29 add -12 rCT % d1
  % (0) exch -29 add -12 rCT
  grestore
} def
...
/decoder {
  startgate
  8 rfont
  (dec) 0 5 rCT
  -16.5 12.5 moveto
  16.5 12.5 lineto
  29.5 -15.5 lineto
  -29.5 -15.5 lineto
  closepath stroke
  dup
  1 add 58 exch div
  2 copy mul
  % 3 -1 roll 1 sub 50 string cvs exch % d1 (n) dn
  % -29 add -12 rCT % d1
  % (0) exch -29 add -12 rCT
  grestore
} def

```

Il codice PostScript/EPS non può essere corretto diversamente, perché queste funzioni non ricevono alcun parametro che informi loro dell'ordine in cui vanno rappresentati i terminali.

Figura u104.4. Multiplatore, decodificatore e demultiplicatore: a sinistra nella loro rappresentazione originale che, però, potrebbe essere errata, a destra togliendo i numeri d'ordine dei terminali.

Se si tolgono i numeri d'ordine, vanno aggiunte delle indicazioni attraverso commenti nel disegno di Tkgate.



### Clock

Tra i moduli già pronti che accompagnano Tkgate, ne esistono due che servono a generare un segnale di *clock*, ovvero un'onda quadra, strutturata in qualche modo. Si tratta precisamente della libreria di moduli denominata 'timer', all'interno della quale sono disponibili i moduli 'ONESHOT' e 'OSCILLATOR'. Questi moduli sono scritti usando codice Verilog di Tkgate.

I due moduli prevedono un parametro, denominato *HZ*, con il quale si presume di inserire una frequenza espressa in Hz, ma non è così: osservando il codice, si può notare che si tratta invece della

durata che dovrebbe avere l'onda o l'impulso, espressa in millisecondi (ms). Tuttavia, pur contando questo fatto, la frequenza che si ottiene nel simulatore è molto differente, perché durante la simulazione non c'è alcuna connessione con l'orologio del sistema operativo ospitante, al quale si rimettono invece i moduli 'ONESHOT' e 'OSCILLATOR'. In pratica, questi due moduli vanno usati con il valore della «frequenza» predefinita, osservando nel simulatore se la frequenza effettiva può andare bene per i propri fini. Va anche osservato che è inutile tentare di modificare il codice di questi moduli, in modo da produrre effettivamente la frequenza desiderata, proprio perché il simulatore non riuscirebbe a farne buon uso.

Per poter disporre di un generatore di frequenza che, in qualche modo, possa essere controllato in relazione al metro usato nella simulazione, occorre costruire qualcosa che si basi sui ritardi di propagazione. Per prima cosa serve un generatore di un impulso; precisamente serve qualcosa che parta producendo un valore azzerato, per poi passare ad attivarsi dopo un certo tempo, mantenendosi attivo per tutto il tempo successivo. Per produrre questo componente virtuale, occorre predisporre un modulo scritto direttamente secondo il linguaggio Verilog di Tkgate:

```
module one_up #(W(1000)) (Z);
  output Z;
  reg Z;

  initial
  begin
    Z = 1'b0;
    $tkg$wait(W);
    Z = 1'b1;
  end
endmodule
```

Si tratta di un modulo privo di entrate e avente una sola uscita: all'avvio l'uscita si presenta a zero e ci rimane per il tempo specificato dal parametro *W*, espresso in millisecondi, quindi passa a uno e ci rimane fino alla fine. Il tempo di pausa iniziale serve a consentire ai componenti di inicializzarsi; il tempo predefinito di 1000 ms, corrispondente a 1 s, viene visto durante la simulazione come un tempo molto più breve, ma generalmente sufficiente: nel caso potrebbe essere aumentato.

Disponendo del modulo di inizializzazione, si può realizzare un circuito che produca un'oscillazione sfruttando il ritardo di propagazione dei componenti; poi, questa oscillazione può essere suddivisa convenientemente in base alle esigenze. Nel capitolo [xxvi](#) si mostra un oscillatore realizzato in questo modo.

<sup>1</sup> **Tkgate** GNU GPL