

Esercizio 1

Sia X la rappresentazione in complemento alla radice su n cifre in una base generica (pari) β del numero intero x . Sia Y la rappresentazione *in traslazione* dello stesso numero.

- 1) esprimere la relazione algebrica che consente di trovare Y in funzione di X ;
- 2) sintetizzare a costo minimo il circuito che produce Y avendo X in ingresso nel caso $\beta = 6$ (con codifica 421);
- 3) sintetizzare a costo minimo il circuito che produce Y avendo X in ingresso nel caso $\beta = 16$ (con codifica 8421).

Esercizio 2

Descrivere e sintetizzare il circuito XXX che (vedi Fig. 1) legge dalla EPROM e comanda il Convertitore A/D che emette una tensione unipolare $v = K \cdot x$, in accordo alle specifiche che seguono.

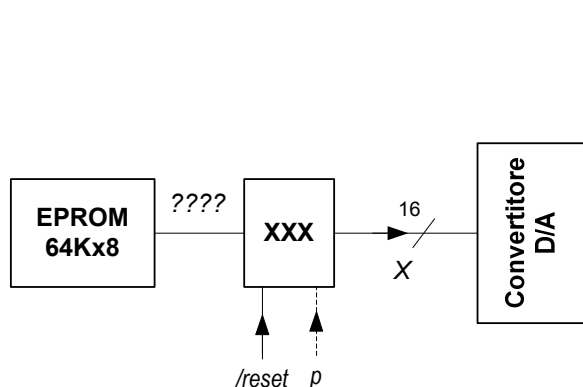


Fig. 1

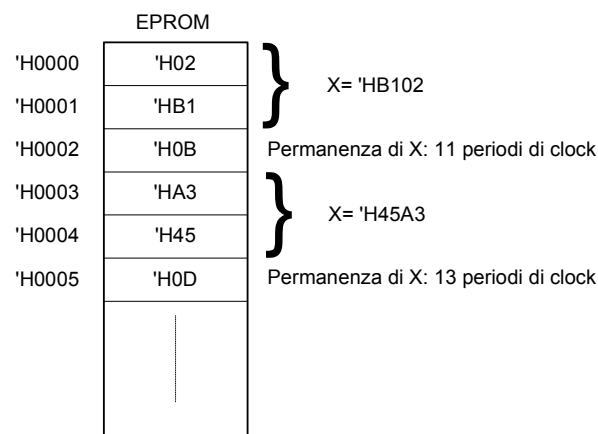


Fig. 2

Nel primo ciclo dopo il reset iniziale XXX emette un campione di valore 0 per (circa) 10 periodi di clock, poi emette quelli prelevati dalla EPROM. L'Unità XXX termina dopo aver emesso 20000 campioni (compreso quello emesso nel primo ciclo successivo al reset iniziale) lasciando in uscita l'ultimo campione emesso.

La struttura della EPROM è la seguente (Fig.2). Le prime due locazioni contengono un campione di x espresso come numero naturale X a 16 bit; la locazione seguente contiene una indicazione su quanto tempo tale campione deve essere mantenuto all'ingresso del convertitore. Stessa cosa per le locazioni successive.

ULTERIORI SPECIFICHE

- 1: Gli 8 bit meno significativi di X si trovano nella locazione di indirizzo più basso.
- 2: L'indicazione sul tempo sta tra 10 e 250 e se tale indicazione vale N , allora X deve essere mantenuto per N periodi di clock.
- 3: La EPROM è sufficientemente veloce da non necessitare di stati di wait.

NOTA: Il caso minimo di 10 periodi di clock non può creare problemi in quanto XXX è descrivibile con un numero di stati interni ampiamente minore di 10.

Usare la seguente impostazione

```
module XXX(...,p,reset_);
  input      p,reset_;
  ...
  output [15:0] X;
  reg [15:0]  CAMPIONE,NEXT_CAMPIONE; assign X=CAMPIONE;
  reg [7:0]   DURATA,NEXT_DURATA;
  ...
  always @(posedge p or negedge reset_)
    if (reset_==0) begin DURATA=10; CAMPIONE=0; ...; STAR=S0; end
    else #3
      casex(STAR)
        S0: ...
        ...
      endcase
endmodule
```

Si tracci un diagramma di temporizzazione facendo riferimento alla EPROM di Fig 2 e terminando dopo aver emesso 2 campioni, compreso il campione di valore 0 emesso dopo il reset.

