Tutte le uFSM hanno lo scopo di eseguire un ciclo preciso svolto dalla CPU e di aggiornare dei valori internamente a Z80X. I cicli in esame sono gli unici che durante il normale funzionamento dello Z80X coinvolgono l’esterno dell’entity. Per cui nella descrizione che ne segue terrò conto del loro port, del ciclo che devono svolgere e dell’FSM che lo implementa e del loro compito all’interno dello Z80X.

Tutte le sottomacchine hanno uno schema e un comportamento comune.

Le uFSM presentano tutte la coppia di segnali TRIG e DONE, seguite dal suffisso della macchina, per il controllo all’interno della struttura nidificata delle FSMs.

Le sottomacchine presentano uno stato di attesa IDLE in cui se il segnale di TRIG è attivo e il CLK è a 1 iniziano il loro ciclo rimanendo insensibili ad altri stimoli su TRIG sino alla fine del ciclo. Per cui quando la FSM Principale attiva il segnale TRIG e la macchina è in attesa inizia il ciclo collegato.  
Si possono far andare le uFSM in continuazione tenendo TRIG attivo così la sottomacchina non entra mai nello stato IDLE ma dall’ultimo stato del ciclo passa direttamente al primo ricominciando il ciclo.  
La uFSM segnala la fine del proprio ciclo con il segnale DONE che rimane attivo anche durante l’attesa. In questo modo la FSM Principale può sfruttare due eventi per la temporizzazione: l’inizio del ciclo richiamato quando DONE si disattiva e la fine del ciclo quando si attiva. Per cui Principale mantiene il segnale di TRIG attivo fintantoché non vede disattivarsi DONE per avere la certezza di aver avviato la sottomacchina correttamente.  
Il segnale di DONE viene generato quando la macchina si trova nello stato di attesa o negli ultimi stati del ciclo per permettere una corretta lettura del segnale da parte di Principale che essendo stata implementata come una macchina di Moore presenta mezzo ciclo di clock di ritardo sulle variazioni dei sui segnali.

Instruction Opcode Fetch, OPFET

Figura Y – Schema dell’entity OPFET

La macchina svolge il ciclo di fetch in cui la CPU legge dalla memoria l’opcode all’indirizzo puntato da PC e lo carica in IR. Dopodiché esegue il refresh della RAM mettendo sul bus l’indirizzo di refresh contenuto in R e attivando gli opportuni segnali.

Con il segnale A\_LD carica in A LATCH il PC o R selezionandoli con il segnale PCnRFSH e con il segnale DIN\_LD carica in IR il valore di DIN per cui è collegato al segnale SHIFT del registro.

Per permettere la sovrapposizione delle fasi di fetch e decode, la macchina fornisce alla FSM Principale un ulteriore segnale RFS\_RUN che si attiva nello stato precedente l’inizio della fase di refresh e rimane attivo fino alla fine della fase. L’attivarsi di questo segnale indica alla macchina di passare dallo stato di attesa a quello di decodifica, da FETWT a DEC o da FETEXTWT a DECEXT.

La macchina oltre al comportamento descritto dal diagramma in Figura X, resetta IR nello stato precedente il caricamento cioè in CHKWT, che è anche lo stato in cui campiona nWAIT per entrare o meno nel ciclo di attesa. Carica invece il valore in IR durante RD.

Figura X – Instruction Opcode Fetch [Z80.pdf p.20]

Figura Y^2 – Digramma di stato della uFSM OPFET.