



UNIVERSITÀ
DEGLI STUDI
DI PADOVA



DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Implementazione della CPU Z80 su FPGA: un primo approccio

Scheda di sviluppo RUCS7

Z80 – Generalità e Storia

Z80 – Architettura

Z80 – Informazioni
sull'organizzazione

Z80X – Implementazione su
FPGA

Z80X – Cicli di controllo

RUCS7 – Controllore via
UART

Conclusioni

Relatore:
Prof. Daniele Vogrig

ANNO ACCADEMICO 2021/2022

Data di laurea 21/09/2022

Laureando:
Giacomo Biasin
1216276

Scheda di sviluppo RUCS7

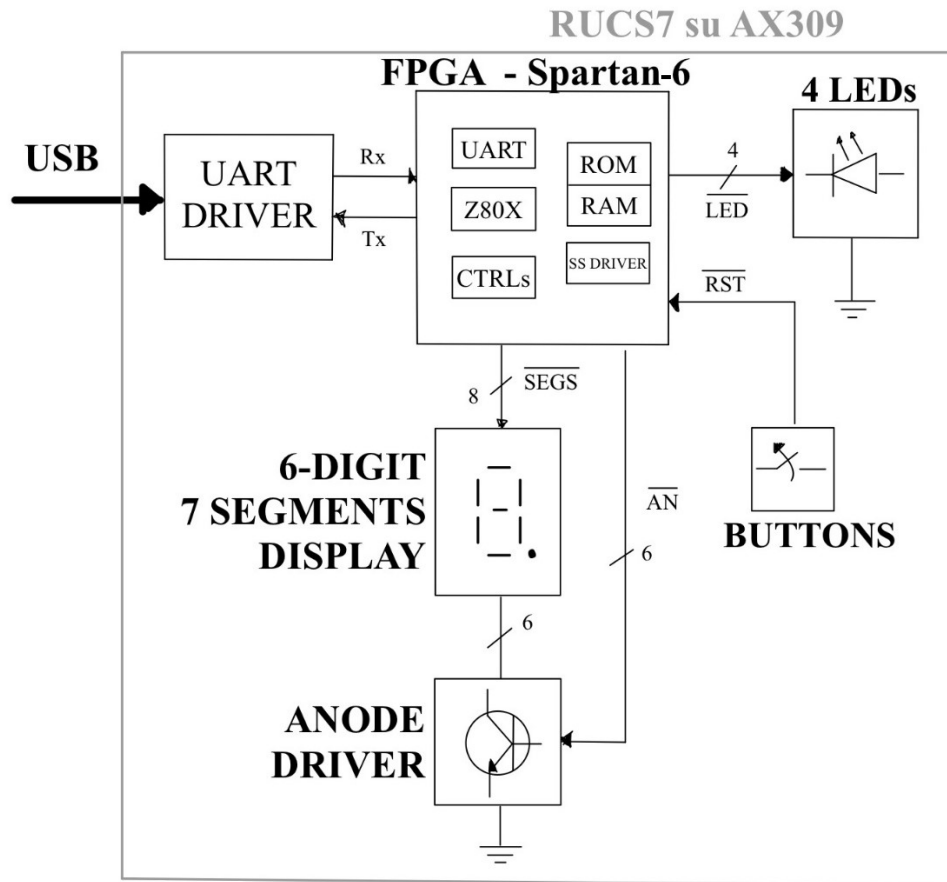


Figura 1 – Schema delle unità usate all'interno della scheda RUCS7

- Z80X - Implementazione comportamentale dello Z80
- Memorie e interfacce per far funzionare lo Z80X
- Sistema per programmare, controllare e fare il debug dello Z80X
- Controllore con interfaccia UART
- Driver per display 7-segmenti con buffer circolare e compatibile con ASCII

Z80 – Generalità e storia



Figura 2 – Federico Faggin e Masatoshi Shima all'evento
Computer History Museum's 2009 Fellows Award

- Progettato da Federico Faggin e Masatoshi Shima
- In relazione con: Intel 4004, Intel 8008, Intel 8080
- Prodotto da Zilog nel 1976
- Singola alimentazione e singolo clock
- Compatibilità con Intel 8080
- Famiglia di periferiche dedicate

Z80 - Architettura

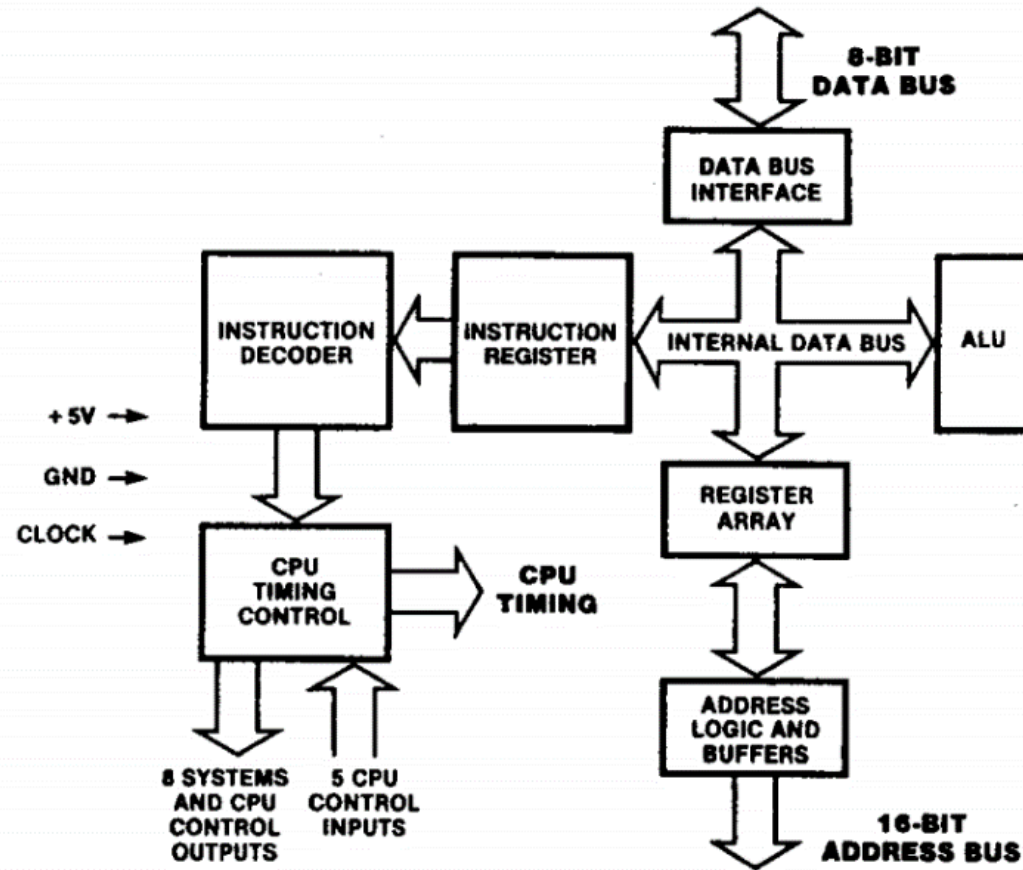


Figura 3 – Diagramma a blocchi dello Z80 dal datasheet

- 2 tipi di interrupt: NMI e INT
- 3 modalità di gestione degli interrupt
- Gestione BUS: BUSREQ
- 6 registri GP: B, C, D, E, H, L
- Registri ombra
- 2 registri d'indicizzazione: IX e IY
- Puntatore stack: SP
- Accumulatore e registro di stato
- Registri dedicati: I e R

Z80 - Informazioni sull'organizzazione

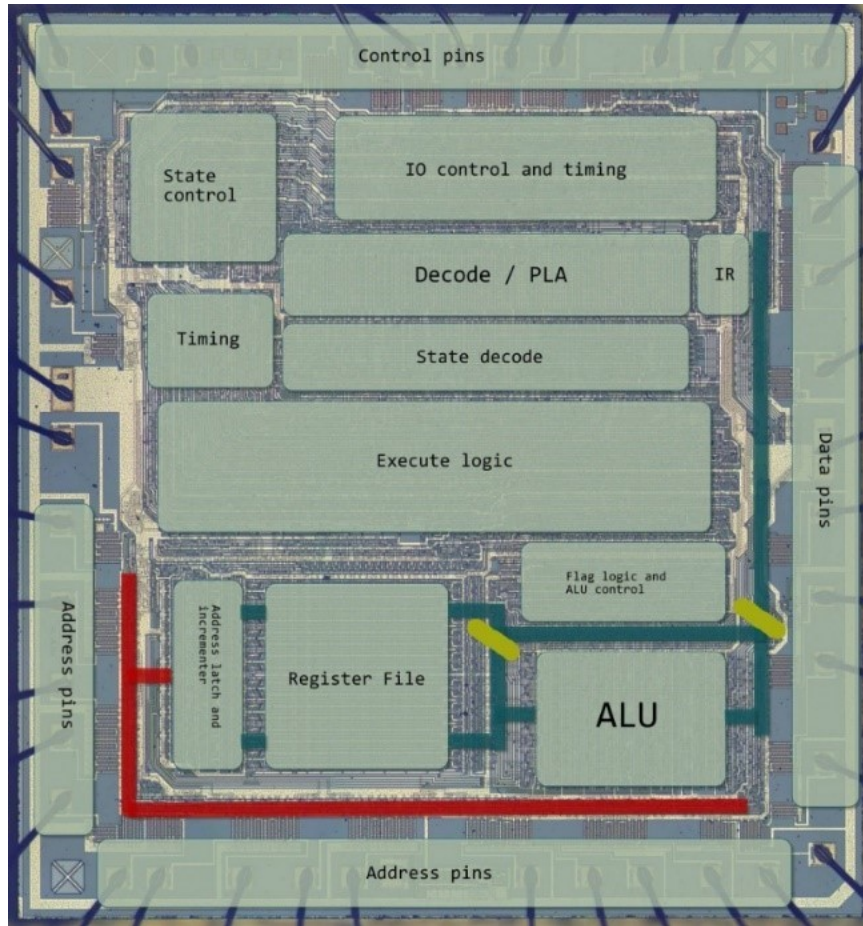


Figura 4 – Die dello Z80 con le unità operative

- ALU a 4 bit anziché 8: registri per i nibble e shift in ALU
- Incrementer/decrementer a 16 bit: PC e I-R, funzione di stop, CLA, 0001H detector
- Due set di registri identici per exchange e due registri d'appoggio
- 3 sezioni del bus a 8 bit
- Decoder basato su PLA e contatore M-cycles e T-cycles
- Comportamenti non documentati

Z80X – Implementazione su FPGA

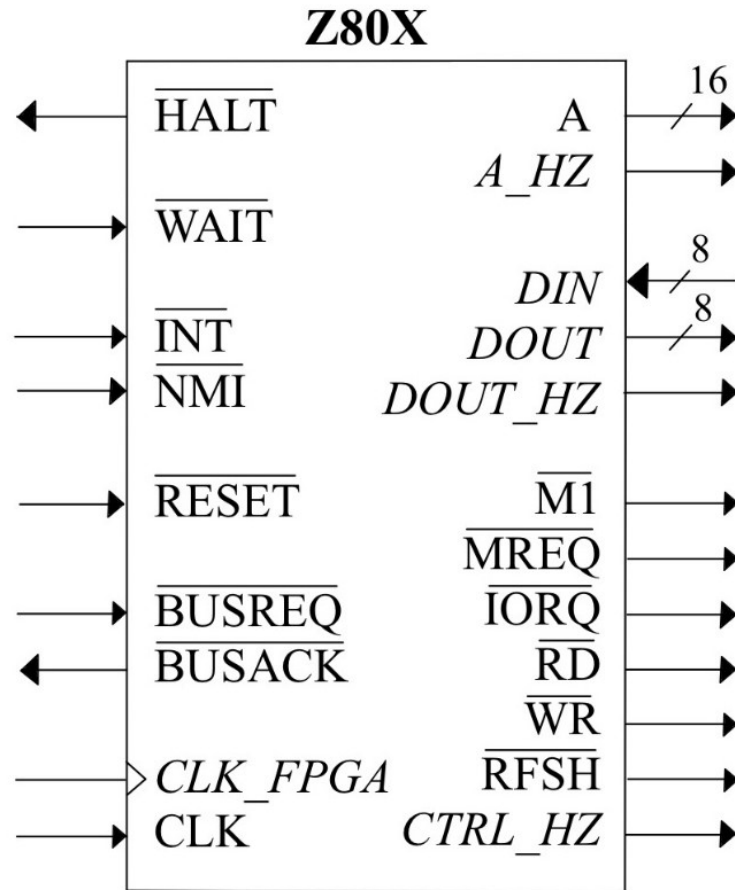
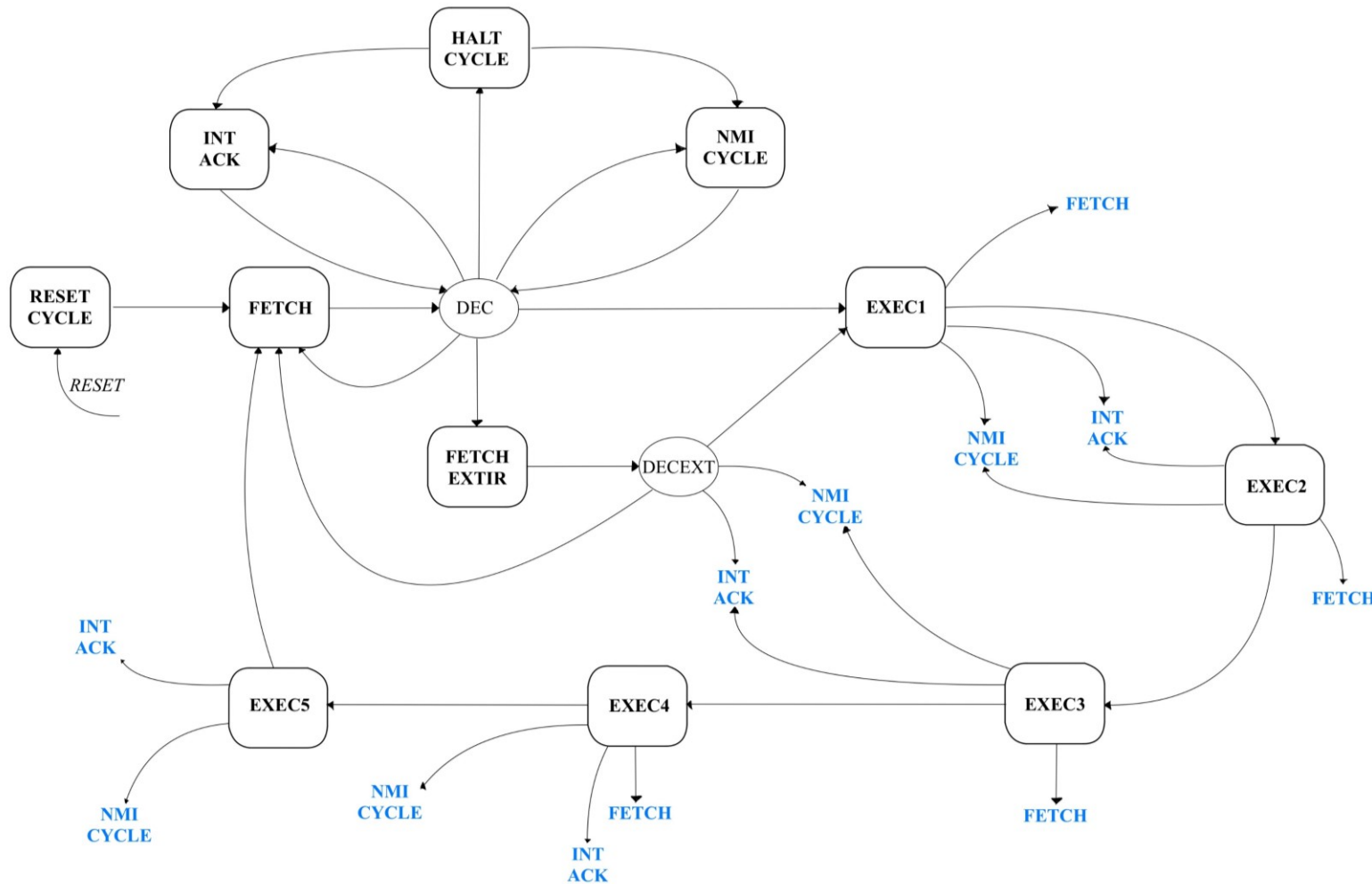


Figura 5 – Schema dell'entity Z80X

Sviluppato dal punto di vista comportamentale:

- Pin dedicati alla gestione 3-state
- Gestione Clock
- Bus dedicati a 8 e 16 bit
- Separazione e gestione registri
- ALU e inc/dec
- Sezione di controllo

Z80X – Cicli di controllo



- FSM Master per BUSREQ
- FSM annidate per il ciclo principale
- μ FSM dedicate: OPFET, MEMRDWR, IORDWR, INTRQ

Figura 6 – Diagramma di stato semplificato della FSM Main di Z80X

RUCS7 – Controllore via UART

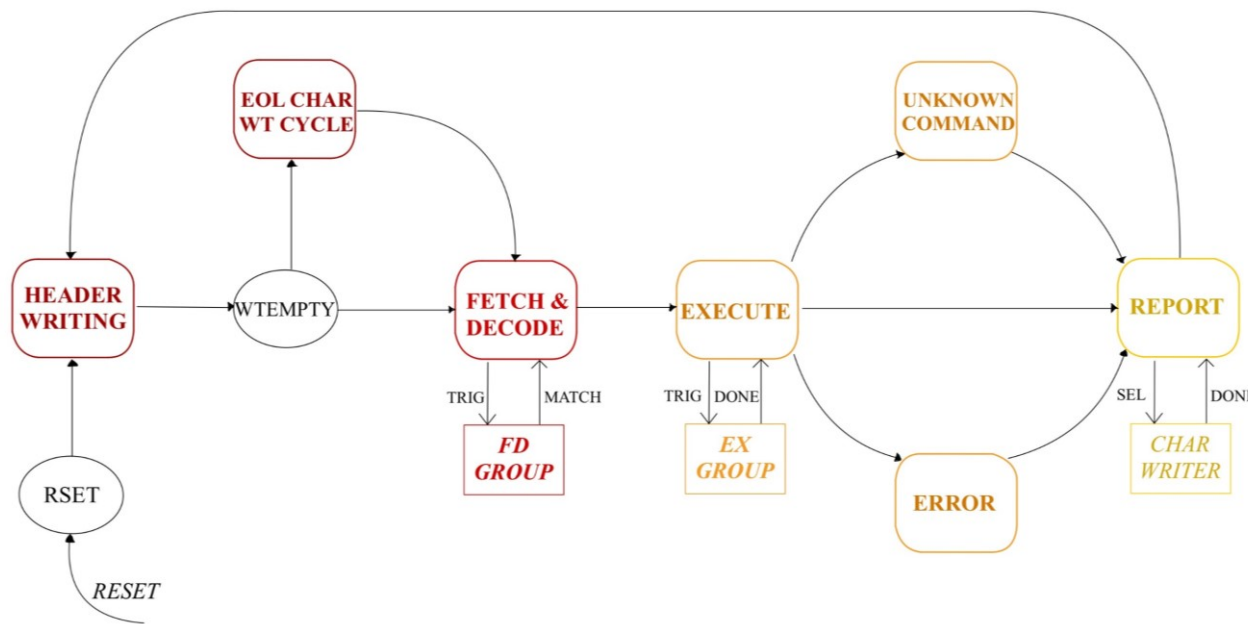


Figura 7 – Diagramma di stato semplificato della FSM di CMD_CTRL

- Realizzato come FSM annidate
- Due gruppi di sottomacchine:
 - FD
 - EX
- Comandi via UART:
 - R/W Memorie
 - Controllo generatore di Clock
 - Istantanee stato dello Z80X
 - Ciclo reset Z80X

Conclusioni

Problemi noti

- Rapporto fra le frequenze di CLK_FPGA e CLK
- OUTI istruzione difettiva
- DECODER richiede molte risorse

Sviluppi futuri

- Decoder basato su BRAM
- Interfaccia tastiera
- Interfaccia VGA
- Z80-DMA



Figura 8 – HP-35