Lo Z80 per funzionare deve essere connesso ad una memoria che contenga il programma da svolgere e i dati necessari. Allo stesso modo lo Z80X deve essere connesso all’interno dell’FPGA con tutto il necessario per farlo funzionare, come le memorie. Per svolgere delle operazioni utili deve poter comunicare con l’utente e per avere un buon flusso di programmazione e di debug ci dev’essere un’interfaccia adatta e veloce.

Figura X – Schema delle unità usate all’interno della scheda AX309

Per questo ho creato delle entity che fanno funzionare lo Z80X e ne permettono il controllo.  
Per prima cosa, lo Z80X è controllabile attraverso la USB presente sulla scheda che viene convertita in protocollo UART e che invia serialmente dei comandi ad un entity dedicata.  
Lo Z80X comunica con l’esterno attraverso il display 7 segmenti a sei cifre presente sulla scheda. Per controllarlo si usano due bus. Il primo porta i segnali per i singoli segmenti di ogni cifra più il punto decimale, nSEGS, mentre il secondo è il comando degli anodi dei segmenti, nAN. Questo permette di avere una sola cifra accesa alla volta sfruttando la persistenza sulla retina così da risparmiare bus e corrente.  
Ci sono anche 4 LEDs collegati che però sono usati per funzioni di debug poiché comunicano lo stato dell’intero sistema. I LEDs segnalano uno stato di errore del controllore, il mancato riconoscimento di un comando via UART, lo stato del clock che viene fornito allo Z80X e infine l’ultimo è collegato all’uscita nHALT dello Z80X.  
L’unico ingresso diretto dalla scheda proviene da un bottone di reset che resetta il controllore e il sistema attorno allo Z80X ma non il microprocessore.

RUCS7

Figura XX – Schema del sistema di sviluppo RUCS7

Il sistema di sviluppo ho deciso di chiamarlo con l’acronimo RUCS7 che contiene tutte le iniziali dei sistemi presenti: Rom/Ram – Uart – Clock - Snapper – 7 segments.  
Questi sistemi permettono rispettivamente: il funzionamento dello Z80X, il controllo dell’intero sistema, la generazione di un segnale di clock variabile, il salvataggio dello stato della macchina e il collegamento dello Z80X con il display 7 segmenti.

Lo scopo di avere un sistema controllabile via UART è quello di permettere una programmazione veloce delle memorie assieme ad un facile ed interattivo debugging del sistema. Per cui il sistema permette di poter leggere e scrivere le due memorie indipendentemente dall’utilizzo che ne fa lo Z80X, di controllare il clock dello Z80X anche sino al livello di un’istruzione alla volta e di avere un tracciato dello stato dei pin d’uscita dello Z80X per ricostruire il suo comportamento durante l’esecuzione.  
Questi vantaggi sono difficilmente implementabili al difuori dell’FPGA, di conseguenza sono i punti di forza di quest’implementazione.