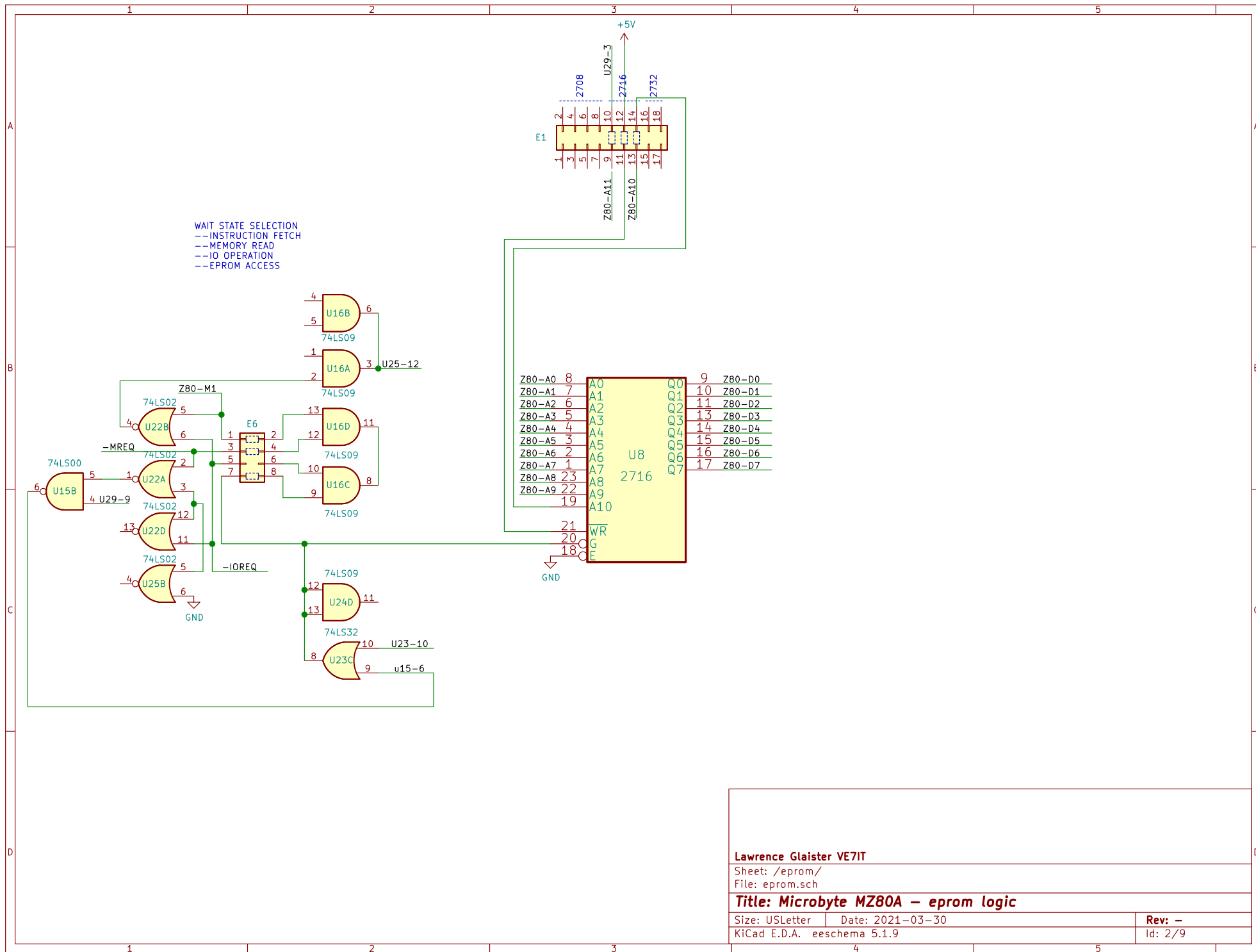


	1	2	3	4	5	
A	Sheet: eprom EPROM LOGIC File: eprom.sch	Sheet: serial Serial Port I/O and cables File: serial.sch				
B	Sheet: io I/O Decoding Parallel, CTC1, CTC2 File: io.sch	Sheet: jumpers Configuration Jumpers File: jumpers.sch				
C	Sheet: interrupts Interrupt Logic S100 BUS File: interrupts.sch	Sheet: untraced-parts Partial packages File: untraced-parts.sch		Notes: This reverse engineering effort was started so the CPU card could be used. No information seems to be available off the internet. To use the card we need to know how to set the jump start address, all the on board I/O port addresses, the eprom address etc. The schematics tend to focus on the critical infrastructure needed to configure the board. Items like power distribution, pullups etc have been largely ignored.		
D	Sheet: processor Z80 Processor File: processor.sch					
	Sheet: poweronreset Power On Reset File: poweronreset.sch					
	1	2	3	4	5	

partial schematics to help understand board operation	
Lawrence Glaister VE7IT	
Sheet: /	
File: mz80a-multi.sch	
Title: Microbyte MZ80A CPU Board	
Size: USLetter	Date: 2021-03-30
KiCad E.D.A. eeschema 5.1.9	Rev: -
	Id: 1/9



Lawrence Glaister VE7IT

Sheet: /eprom/

File: eprom.sch

Title: Microbyte MZ80A – eprom logic

Size: USLetter Date: 2021-03-30

KiCad E.D.A. eeschema 5.1.9

Rev: –

Id: 2/9

B

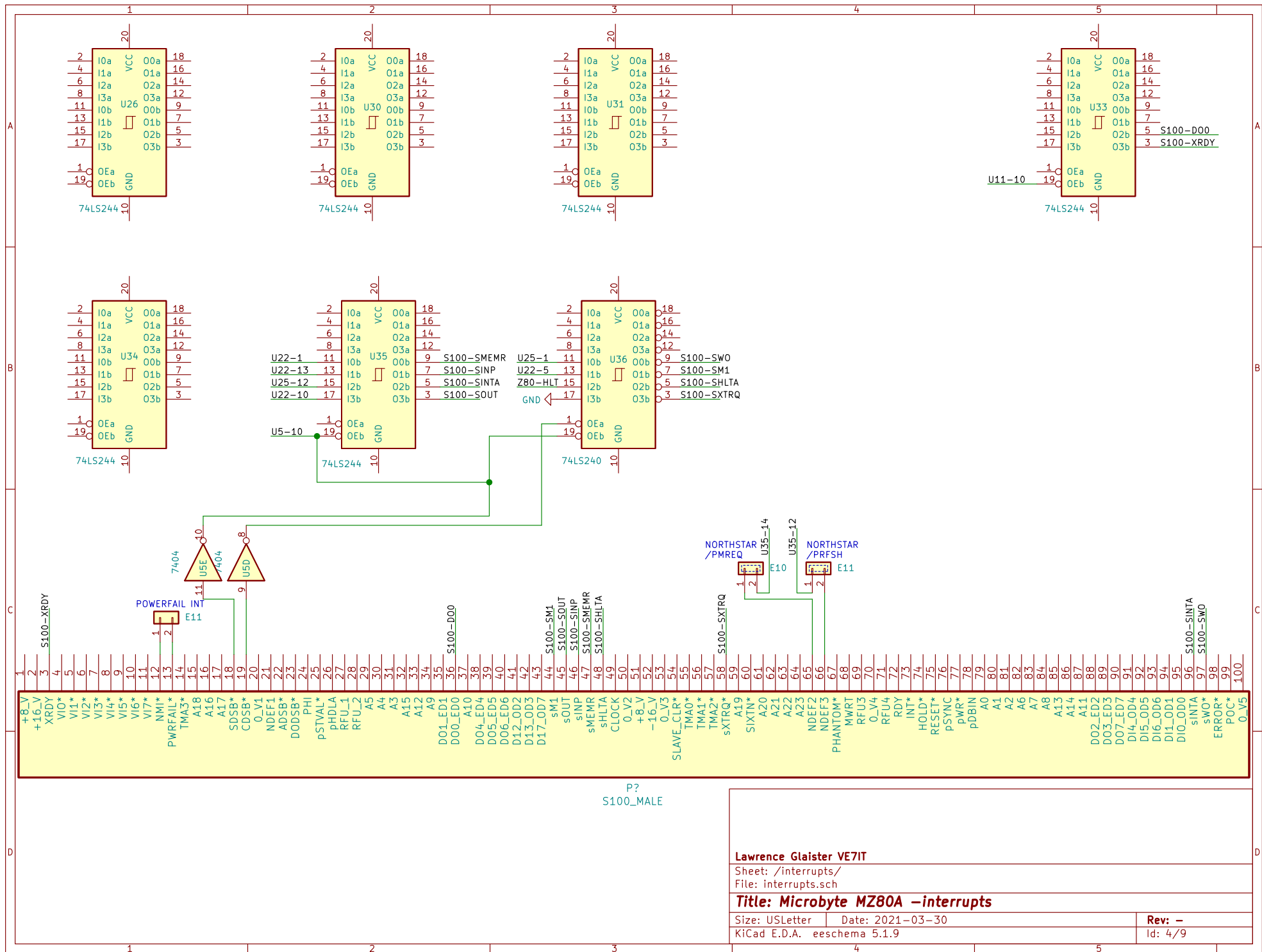


2



2

Id: 3/9



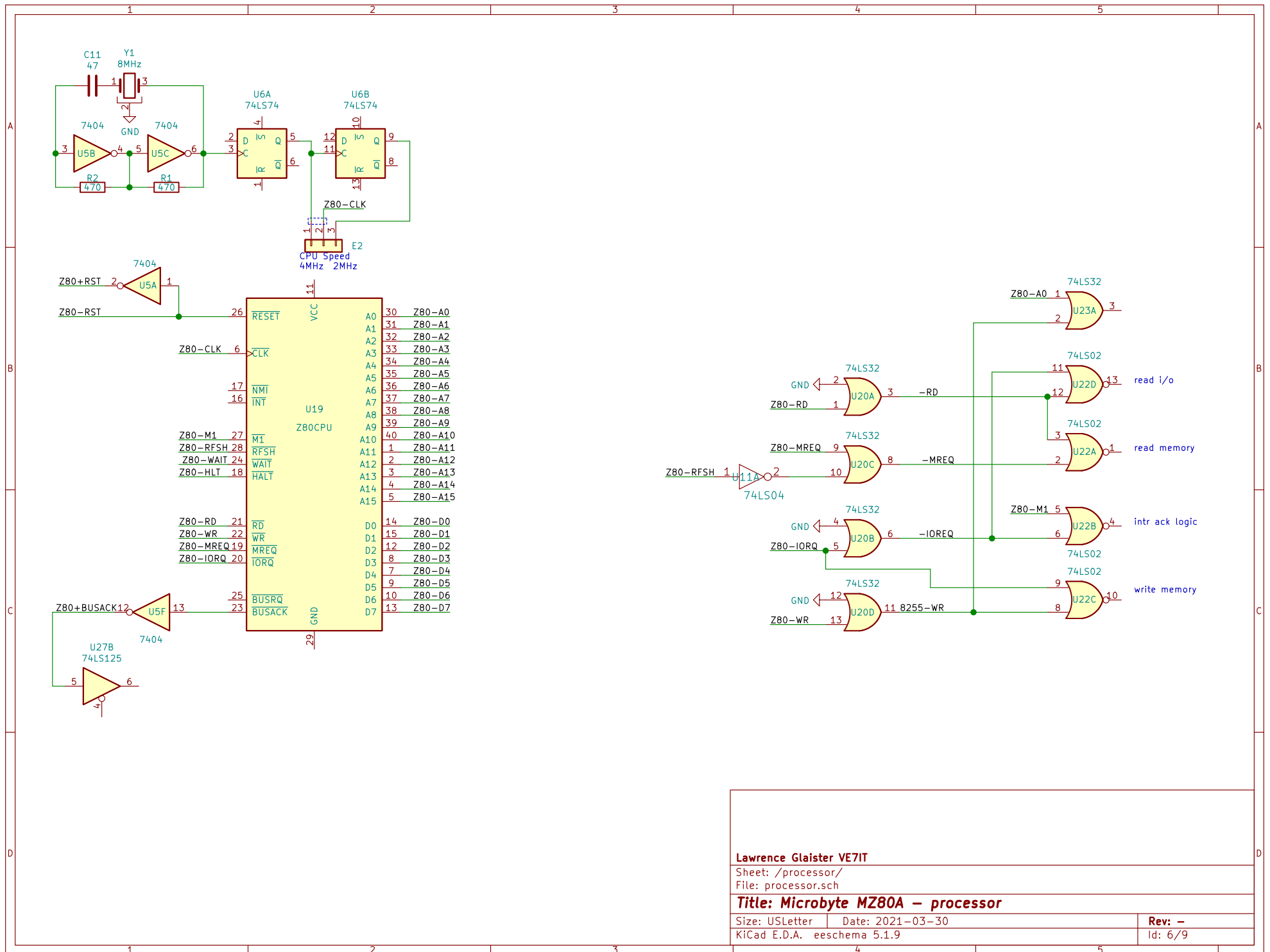
Lawrence Glaister VE7IT

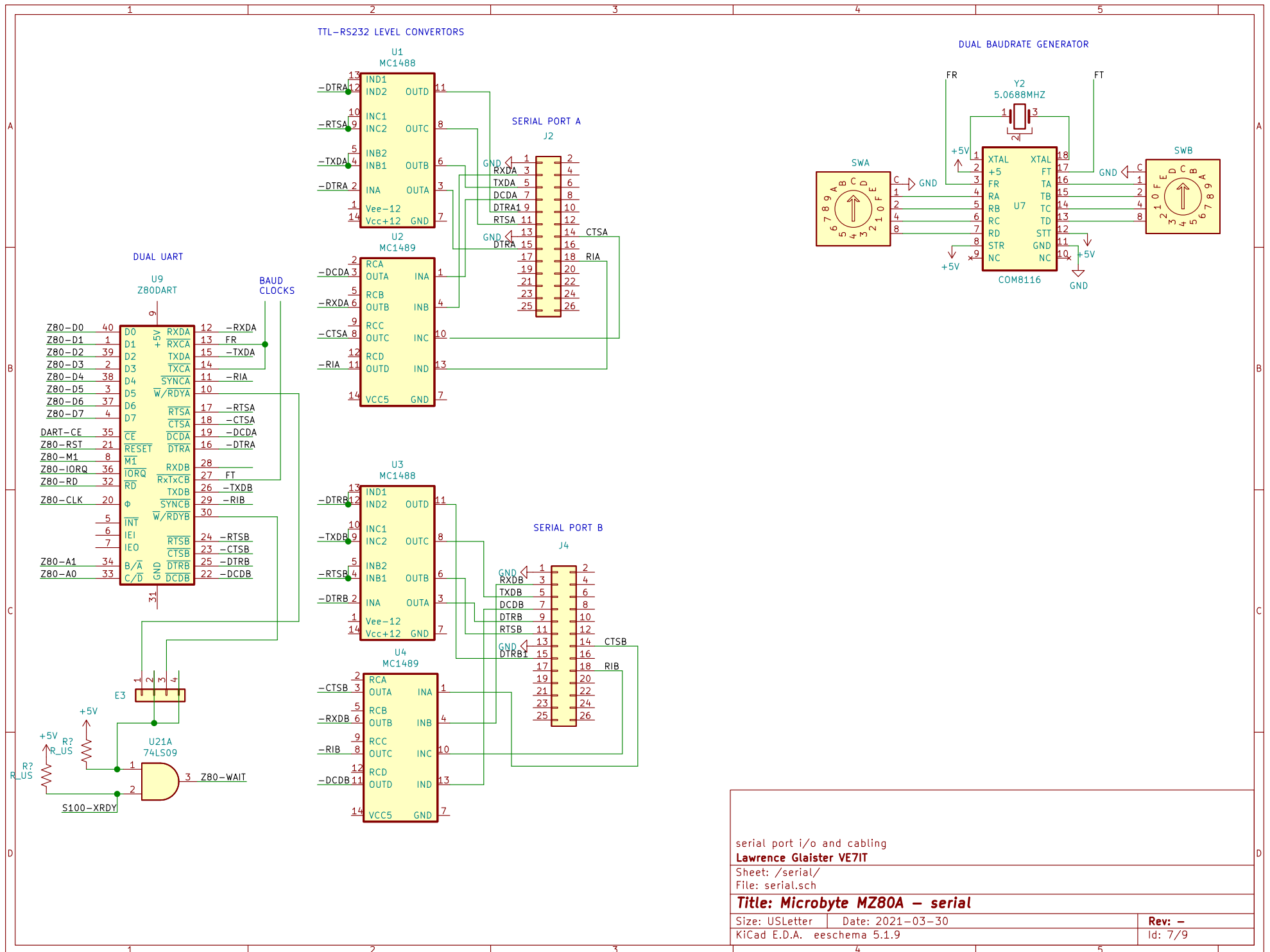
Sheet: /interrupts/
File: interrupts.sch

Title: Microbyte MZ80A -interrupts

Size: USLetter Date: 2021-03-30
KiCad E.D.A. eeschema 5.1.9

Rev: -
Id: 4/9





serial port i/o and cabling

Lawrence Glaister VE7IT

Sheet: /serial/

File: serial.sch

Title: Microbyte MZ80A – serial

Size: USLetter Date: 2021-03-30

KiCad E.D.A. eeschema 5.1.9

Rev: –

Id: 7/9

