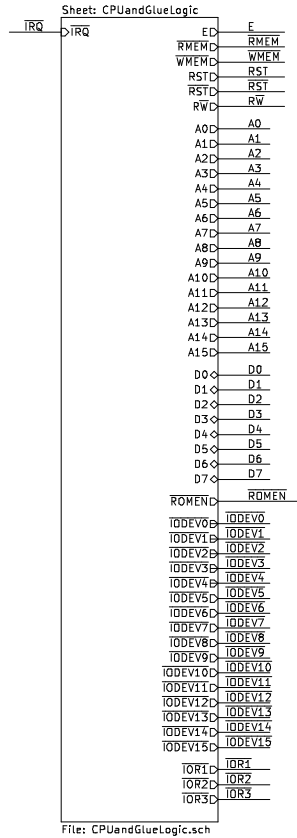
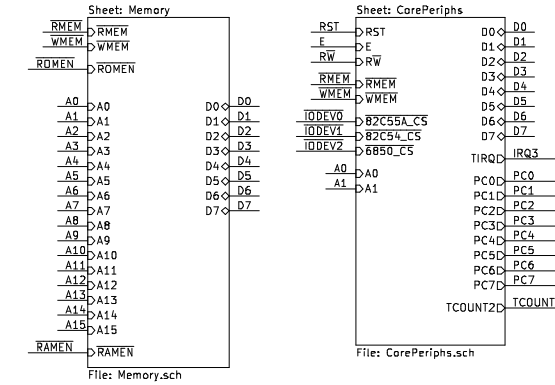


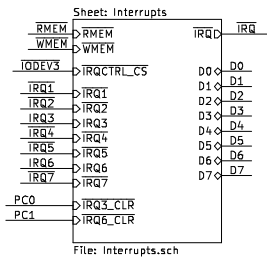
Module 1: core (CPU and glue logic)



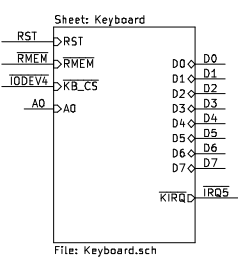
Module 2: memory and core peripherals



Module 3: interrupt controller



Module 4: keyboard controller



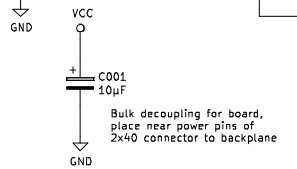
Sheet: Display



Backplane connector: one per board PCB
(appears here because all backplane signals exist on this sheet)

back row (backplane) front row (backplane)
bottom row (board) top row (board)
Conn_02x40_Odd_Even

back row (backplane)	front row (backplane)	bottom row (board)	top row (board)
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40
41	42	43	44
45	46	47	48
49	50	51	52
53	54	55	56
57	58	59	60
61	62	63	64
65	66	67	68
69	70	71	72
73	74	75	76
77	78	79	80

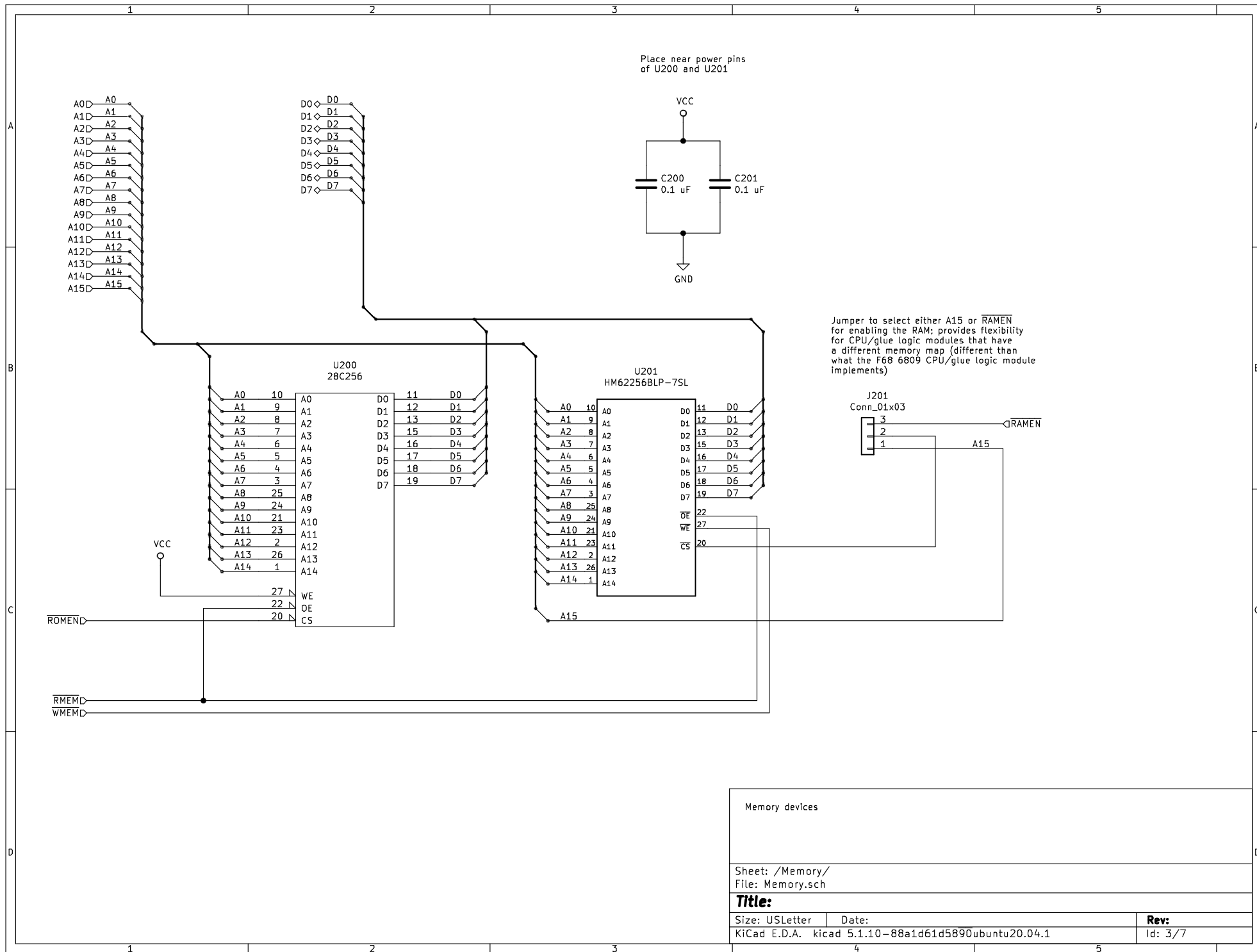


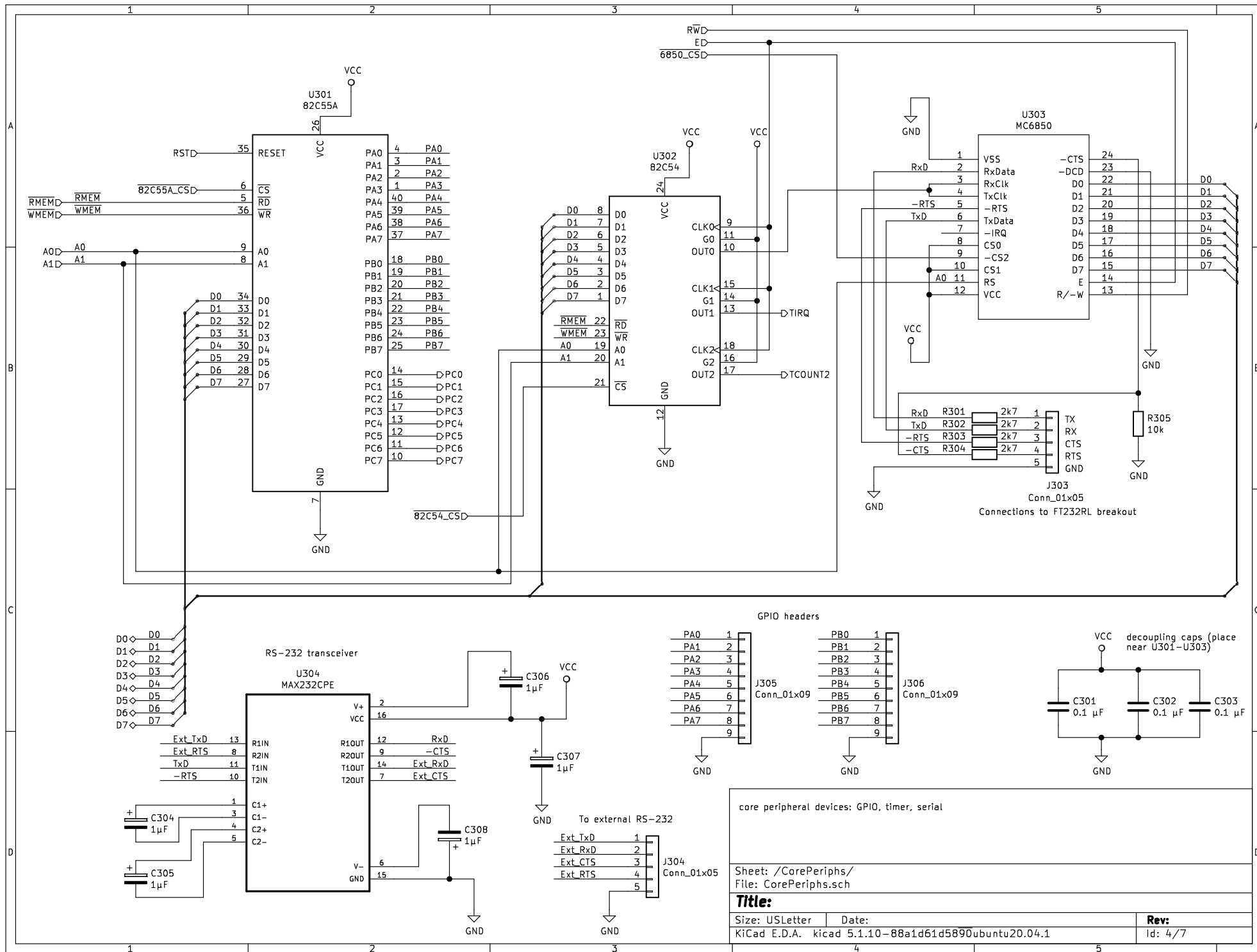
F68: A DIY 6809 microcomputer system

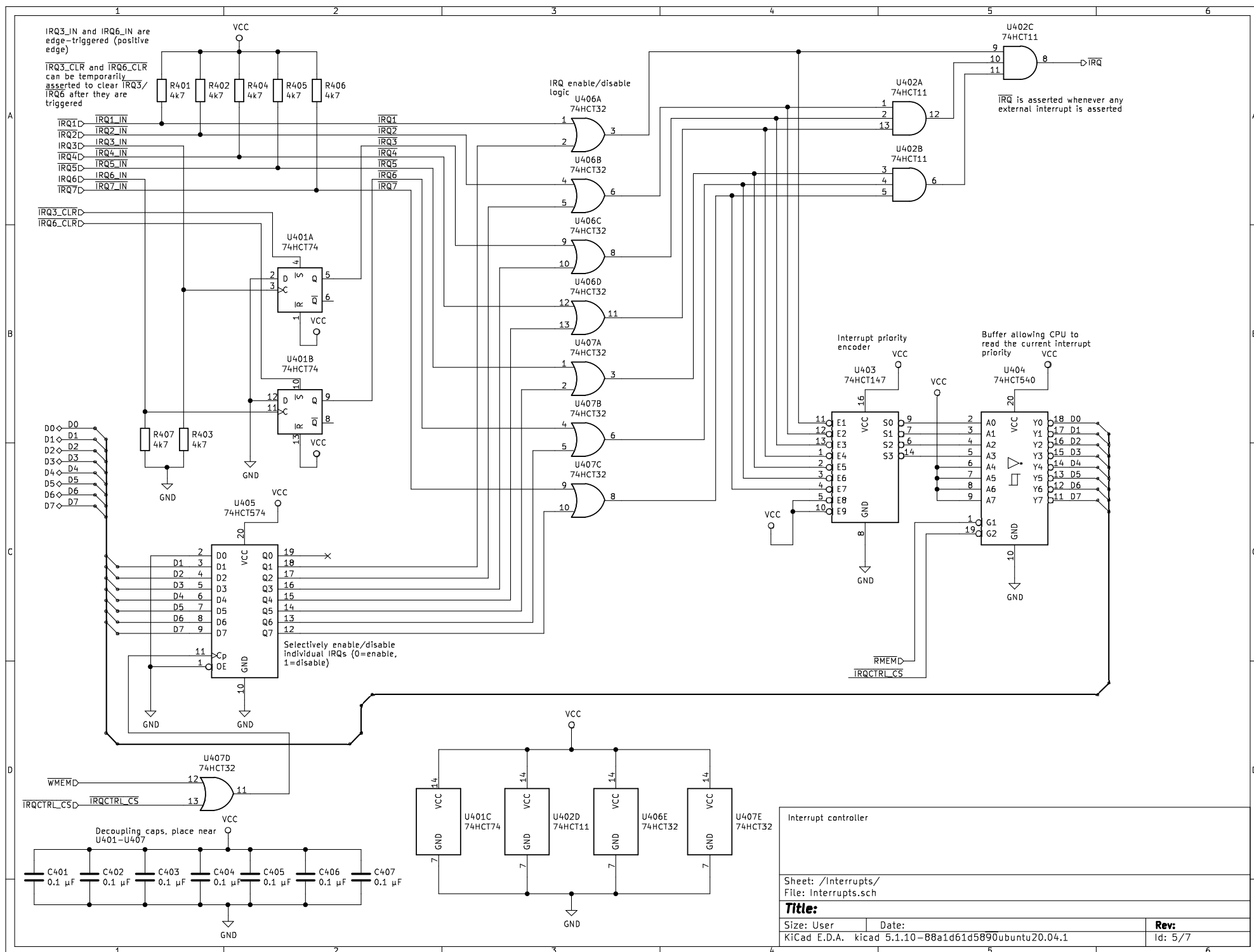
Sheet: /
File: F68.sch

Title:

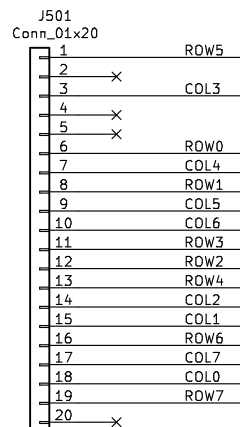
Size: User Date: Rev:
KiCad E.D.A. kicad 5.1.10-88a1d61d5890ubuntu20.04.1 Id: 1/7



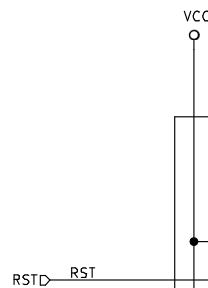
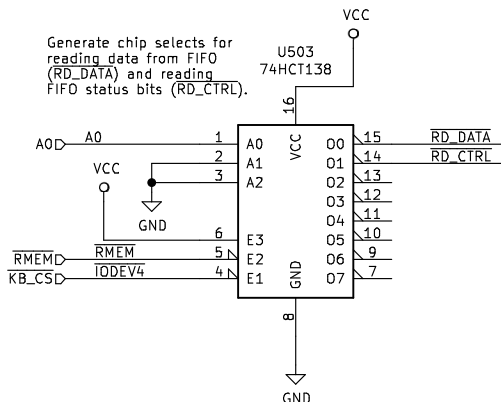




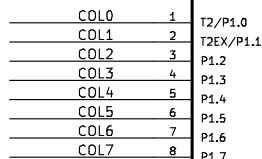
C16 keyboard connector, pin 2 is a key and should be removed



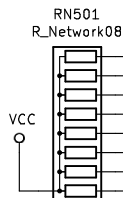
Generate chip selects for reading data from FIFO (RD_DATA) and reading FIFO status bits (RD_CTRL).



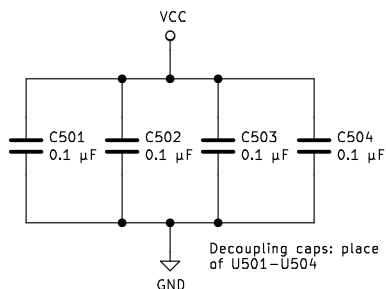
Port 1 actively drives the columns of the keyboard matrix: outputs are normally at VCC, but a column output is driven to GND when a scan of that column is in progress



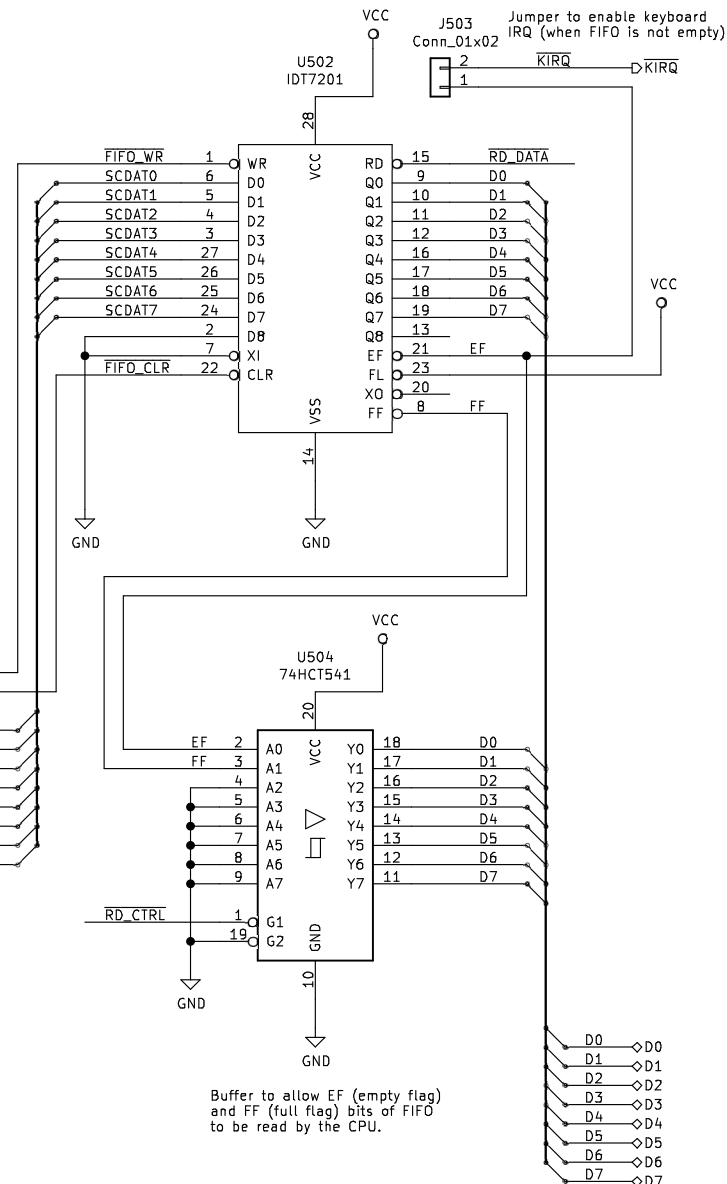
Port 0 reads the rows (with pull up resistors keeping the row inputs at VCC unless a keyswitch has made a connection to a column when the column scan is in progress)



The exact value of RN501 isn't critical: anything between 2k7 and 47k should work fine.



Microcontroller to scan keyboard matrix and send scan codes to FIFO. Will be a 89C51 or 87C51.



daveho hacks

Sheet: /Keyboard/
File: Keyboard.sch

Title: keyboard controller

Size: USLetter Date: KiCad E.D.A. kicad 5.1.10-88a1d61d5890ubuntu20.04.1

Rev: Id: 6/7

