32-Bit Register

Michael Rudolf

Table of Contents

Purpose

This register is for my experimental CPU and is **completely untested** so far. I **do not recommend using this PCB** without thoroughly double-checking everything. I am not legally liable if this PCB does not work (which I don't think it will), should you choose to manufacture it. However, if you still wish to proceed, it's released under the MIT license.

The information in this data sheet is purely speculative.

Output enable

There are two options for sending the stored data to the data bus, either using just one bit or using multiple. While the first option might primarily be used to enable/disable by hand, the other one is for addressing registers by numbers.

Option 1: Just one bit (OE)

If OE is tied low, the register will output its contents to the data bus. OE is connected like any bit of the OE-Bus, resulting in no bit in the OE-Bus being allowed to be tied high or it will not output.

Option 2: Output enable bus (OE-Bus)

If all of the output enable bus' pins are tied low (and also OE), the register will output its contents. This behaviour can be used to input the negated version of a register's addressing number to address it.

Input enable

There are two options for storing the data from the data bus, either by using just one bit or using multiple. While the first option might primarily be used to enable/disable by hand, the other one is for addressing registers by numbers.

Option 1: Just one bit (STO)

If STO is tied low, the register will store the data from the data bus. STO is connected like any bit of the STO-Bus, resulting in no bit in the STO-Bus being allowed to be tied high or it will not activate.

Option 2: Store data enable bus (STO-Bus)

If all of the STO-Bus' pins are tied low (and also STO), the register will start storing the contents. This behaviour can be used to input the negated version of a register's addressing number to address it.

Note

From a functionality standpoint, the addressing is similar to the addressing in the output section. Internally, the buses need to be negated for input enable, as the SN74HC273's clock pin is active-high while this is not needed for the output enable due to the buffer's pins being active-low.

Data Bus

Physical Location & Silkscreen

The data bus is located near the center of the board horizontally and takes up about 60% of the board vertically. Groups of 8 bits are marked on the silkscreen as individual buses with the least-significant-bit being marked as 1 and the most-significant-bit being marked as 8.

Usage

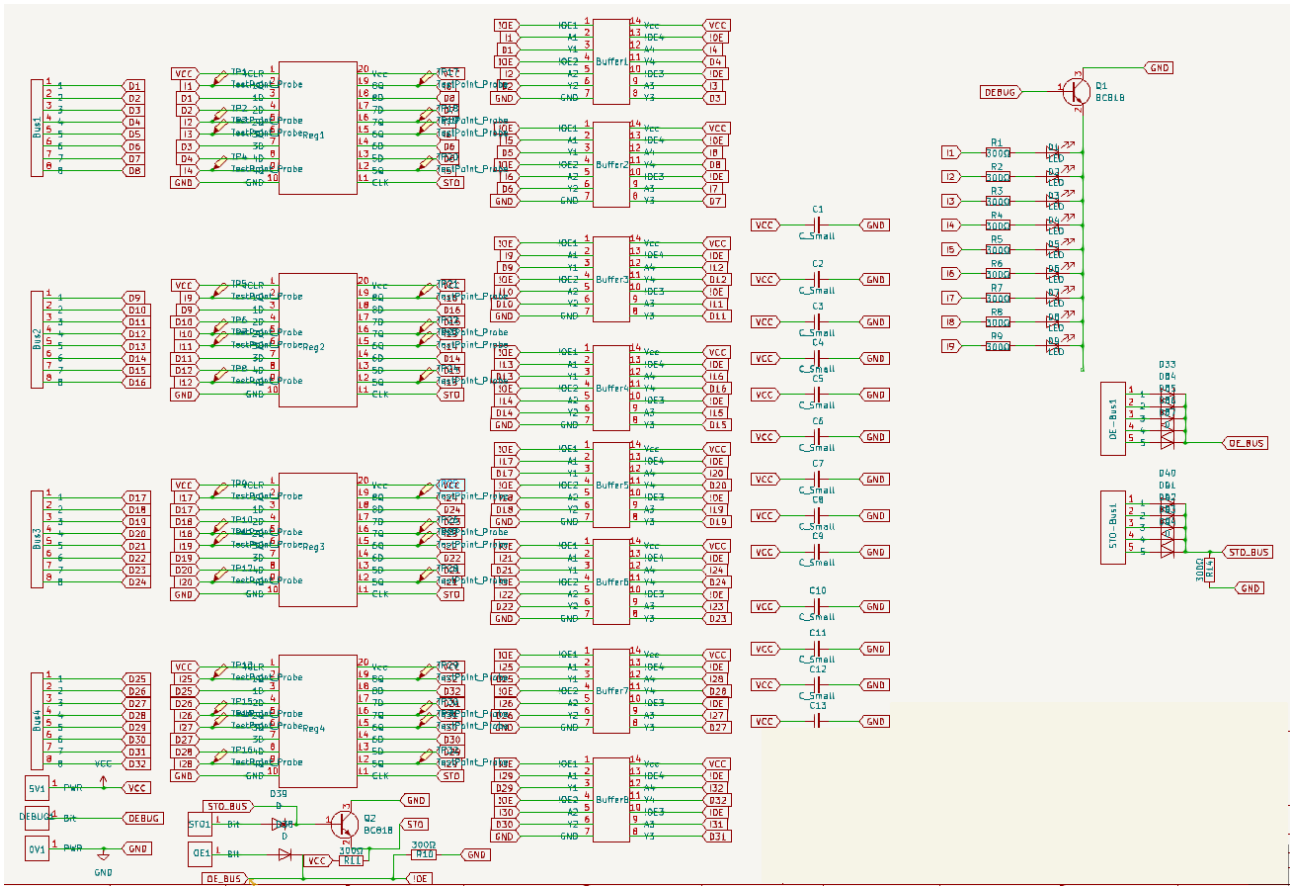
This bus is for sending and receiving 32 bits of data. Multiple registers can be used on the same bus, but only one should output at a time. More than one register sending data might damage some integrated circuits.

Probing

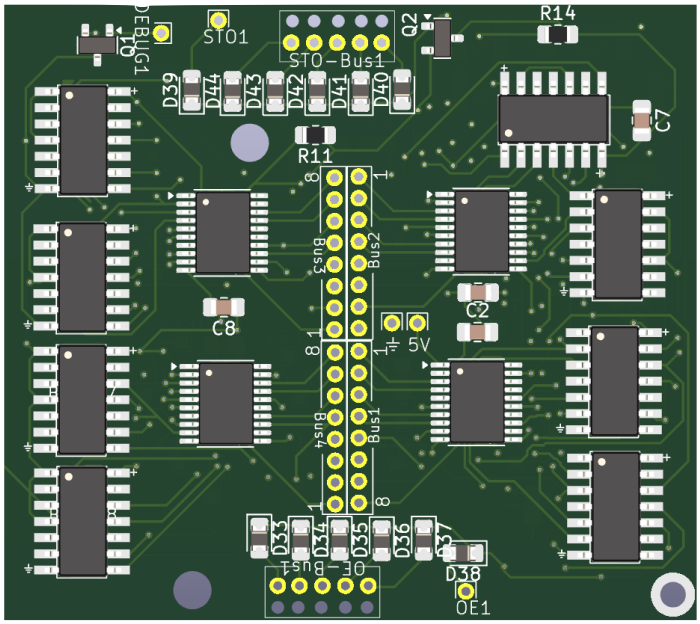
There is one probing pad per bit outputting this bit's current value. Those pads are marked TP_x. X can be converted to the bit number using the table below (x: pad number, b: corresponding bit number).

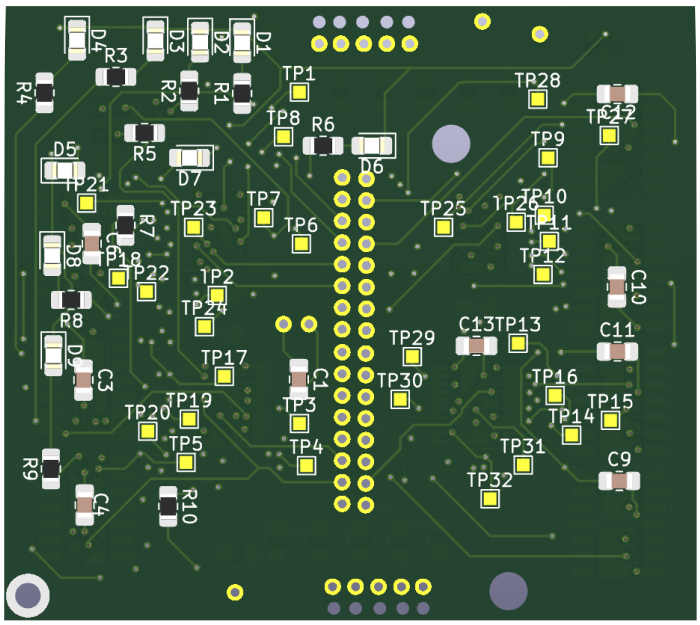
x	b
1	1
2	2
3	3
4	4
5	9
6	10
7	11
8	12
9	17
10	18
11	19
12	20
13	25
14	26
15	27
16	28
17	8
18	7
19	6
20	5
21	16
22	15
23	14
24	13
25	24
26	23
27	22
28	21
29	32
30	31
31	30
32	29

Schematic



3D Images


DEBUG1
STO1
STO-Bus1
Q2
R14
D39
D44
D43
D42
D41
D40
C7
R11
Bus3
Bus2
Bus4
Bus1
C8
C2
5V
D33
D34
D35
D36
C37
OE-Bus1
D38
OE1


D4
D3
D2
D1
R3
R4
R2
R1
TP1
TP8
R6
TP28
C12
TP27
D5
R5
D7
D6
TP9
TP21
TP23
TP7
TP6
TP25
TP26
TP10
TP11
R7
C6
D8
C18
TP22
TP2
TP12
C10
R8
TP24
C13
TP13
C11
TP29
TP17
TP30
TP16
TP15
TP19
TP20
TP3
TP14
TP5
TP4
TP31
C9
TP32
R9
C4
R10