Roadblock for I2C communication with MPQ4210

viernes, 7 de junio de 2024 00:44

Check the level shifter behavior

1. Connect both the Rpi and the MPQ4210 to the level shifter. And add Pull-Up resistors on the Rpi side
2. Boot both devices
3. By now, as the I2C line is enabled you should be able to check the clock signal integrity on both sides with an oscilloscope (If you happen to have one)
    a. If you see a 100 kHz signal on both sides of the level shifter, one of 3.3V and the other of 5V. Level shifting should be okay
    b. If you do not happen to have an oscilloscope at hand you should do an isolated test of your level shifter, connect either side to VDD and you should see VDD on both sides, just that, one will be 3.3V and the other 5V.
4. You can also test this with the "**i2cdetect -y <I2CBus>**" test, if either 0x66 or 0x64 address pops up, this means that the RPI is able to communicate with the MPQ4210
5. If step 4 is successful you can proceed with the next section. Else, you should check the level shifter or the pull-up resistors.
    a. For pull-up resistors checking: Is there a correct resistance value for I2C pull-up resistors? - Electrical Engineering Stack Exchange or I2C Bus Pull-Up Resistor Calculation
    b. For level shifting refer to step 3
    c. After each adjustment you should be repeating step 4 to check if you can reach communication with the device.

Check basic communication with the MPQ4210

Once you have reached communication with the MPQ4210, you can check basic communication with the MPQ4210, for this you can.

1. Execute i2cget commands for each and every register from the MPQ4210 device, registers are:

Address	Register	Type	D7	D6	D5	D4	D3	D2	D1	D0	Reset State
0x00	REF_LSB	R/W	-	-	-	-	-	VREF_L			0000 0100
0x01	REF_MSB	R/W	VREF_H								0011 1110
0x02	Control 1	R/W	SR		DISCHG	Dither	PNG_Latch	Reserved (9)	GO_BIT	ENPWR	0100 0000
0x03	Control 2	R/W	FSW		-	BB_FSW	OCP_MODE		OVP_MODE		1000 0101
0x04	ILIM	R/W	-	-	-	-	Reserved	ILIM			0000 1001
0x05	Interrupt status	R/W	-	-	-	OTP	-	OVP	OCP	PNG	0000 0000
0x06	Interrupt mask	R/W	-	-	-	M_OTP	-	M_OVP	M_OCP	M_PNG	0000 0001

Command usage example for Control1 Reg: i2cget -y <I2CBus number> <Device address> 0x02

- Device address you will get on the step 4 of the previous section.
- If you get an error on the i2cget command it could mean a faulty i2c communication line. First thing I would check are the pull-up values

2. Try resetting the MPQ4210 if you can, that way you know which values you should be reading from each register, from the **Reset State** column on the table.
3. If you get all the reset values on registers 0x00-0x004 we're okay, and can continue to the next section.

Set output voltage

As per the MQP4210 datasheet:

$$(1)\ V_{REF} = \frac{R_2}{R_1 + R_2} V_{out}$$

From this equation you can get the Vref related to your desired Vout. Now, as you can see in the MPQ4210 datasheet

> Total 11 bits to set reference voltage. If V is an 11-bit, unsigned binary integer of VREF [10:0], then: V_{FB} (V) = V/1000.

Which means the value to be loaded on the VREF registers is exactly 1000 times the value you get from the equation or its closest integer.

Now, you have to take this into consideration, the loaded value cannot be greater than 2047 as it is limited by the 11 bits we have at hand.

So, for you to set up an output voltage you would have to follow the following steps:

1. Get Vref value with equation (1)
2. Multiply by 1000 to get the Vref value to be loaded
3. Do the decimal to binary converstion from that number
    a. The three less significant bits have to be loaded to REF_LSB register

       You can load the value with the following command: i2cset -y <I2CBus number> <Device address> 0x02 <hexadecimal representation of the three least significant bits>

    b. The eight bits remaining have to be loaded to REF_MSB register

       You can load the value with the following command: i2cset -y <I2CBus number> <Device address> 0x02 <hexadecimal representation of the eight most significant bits>

4. Do i2c reading operations on the REF_LSB and REF_MSB registers to check if the values you set are the ones which were set.
5. After this, you should set the GO_BIT for the Vref to be reached by the MPQ device you can do this by loading 0b01000010 (use the hexadecimal representation for the command) to the Control 1 register (0x02) with an i2cset command on the cmd.
6. Only after you have done this will you see a change in the Vref.
    a. You will only be able to see changes on Vout if power switching is enabled through ENPWR bit. You can enable it if needed by loading 0b01000001 but it has to be at least 200ms after you had set the GO_BIT in step 5.