8080

Hardware User's Manual

Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Trademark

The names used for identification only maybe registered trademarks of their respective companies.

Table of Contents

1. Introduction

The 8080 is a 4/8 channel Counter/Frequency module designed for the 8000 series modules and WinCon8000. With the Counter mode can be provided to measure quantities such as movement and flow quantity, and the Frequency mode can be provided to calculate the instantaneous differential of quantities such as rotating speed, frequency or flow rate ,etc. The key features of 8080 are as follows:

Operational mode: counter mode or frequency mode

Channel number:

- 8 channels for **Frequency** mode
- 8 channels for **Up-counting** mode
- 4 channels for **Up/Down counting** mode
- 4 channels for **Dir/Pulse counting** mode

Input Signal:

- Isolated or Non-isolated TTL selected by jumper

Input range for Frequency mode

- Isolated: 0 ~ 450KHz
- Non-isolated TTL: 0 ~ 450KHz

Input range for Counter mode

- IsolTated: 450KHz max
- Non-isolated TTL: 450KHz max

Logic high level

- Isolated: 4.5V ~ 30V
- Non-isolated TTL: 2 ~ 5V

Logic low level

- Isolated: 0V ~ 1V
- Non-isolated TTL: 0 ~ 0.8V

EEPROM: 128 bytes

Programmable built-in gate time: 0.33sec (Default)

Digital filter: 1μS ~ 32767μS programmable

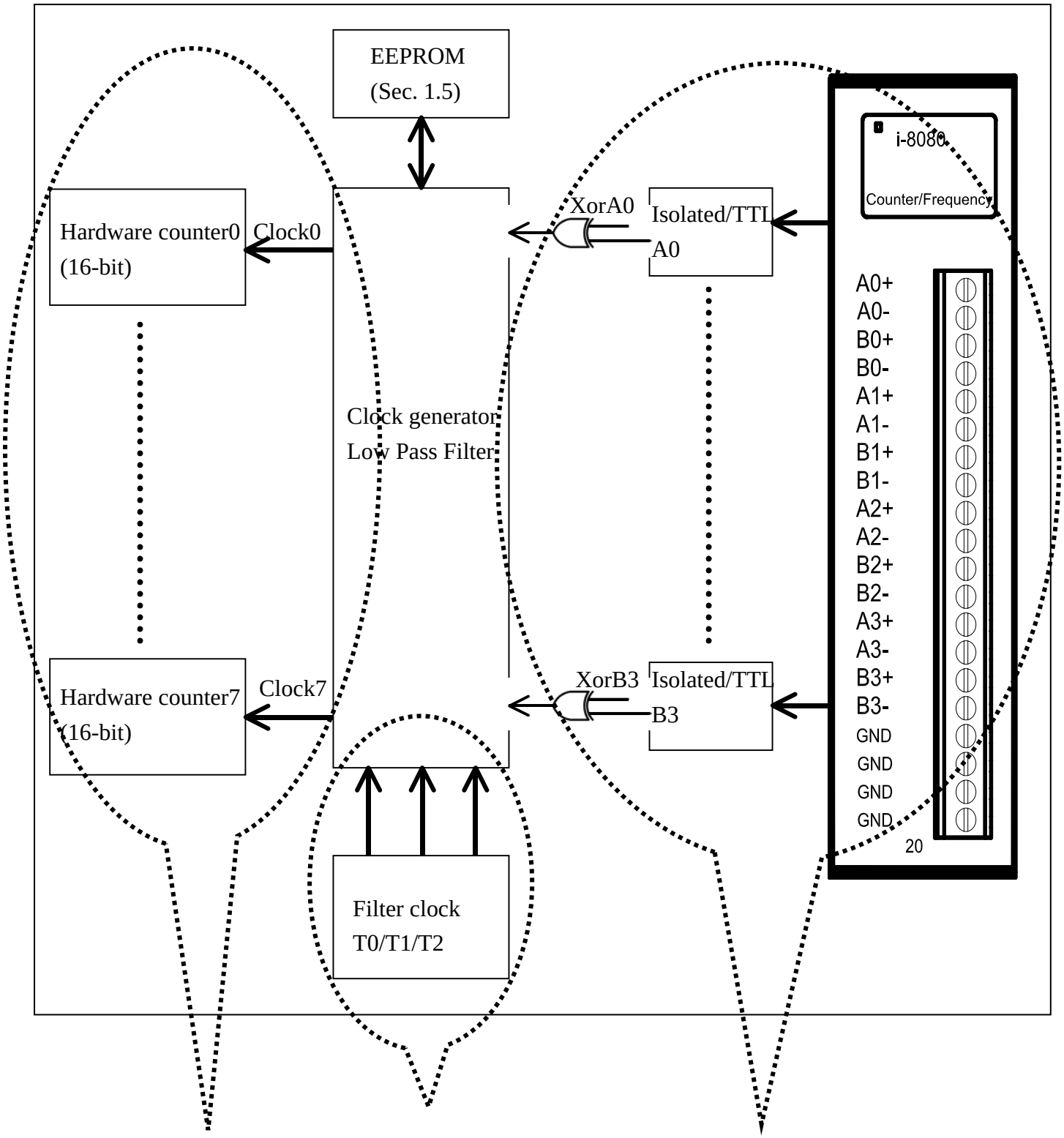
Minimum pulse width: 1μS (Frequency and Counter mode)

Isolated voltage: 3750 Vrms

Minimum Input Current : 2mA (Isolated)

Power consumption: 1W

Signal flow block diagram



1.1. Input signal model

1. Isolated Input (XOR=0)

The operational logic applied on the 8080 modules is the falling edge trigger (Normal High & active Low). The external signal is input into an 8080 module through the isolated mechanism, with the signal being reversed from the external signal. This internal signal is the suggested waveform, as it doesn't need to execute the XOR operation (XOR=0). The solution is shown in Figure 1-1 below.

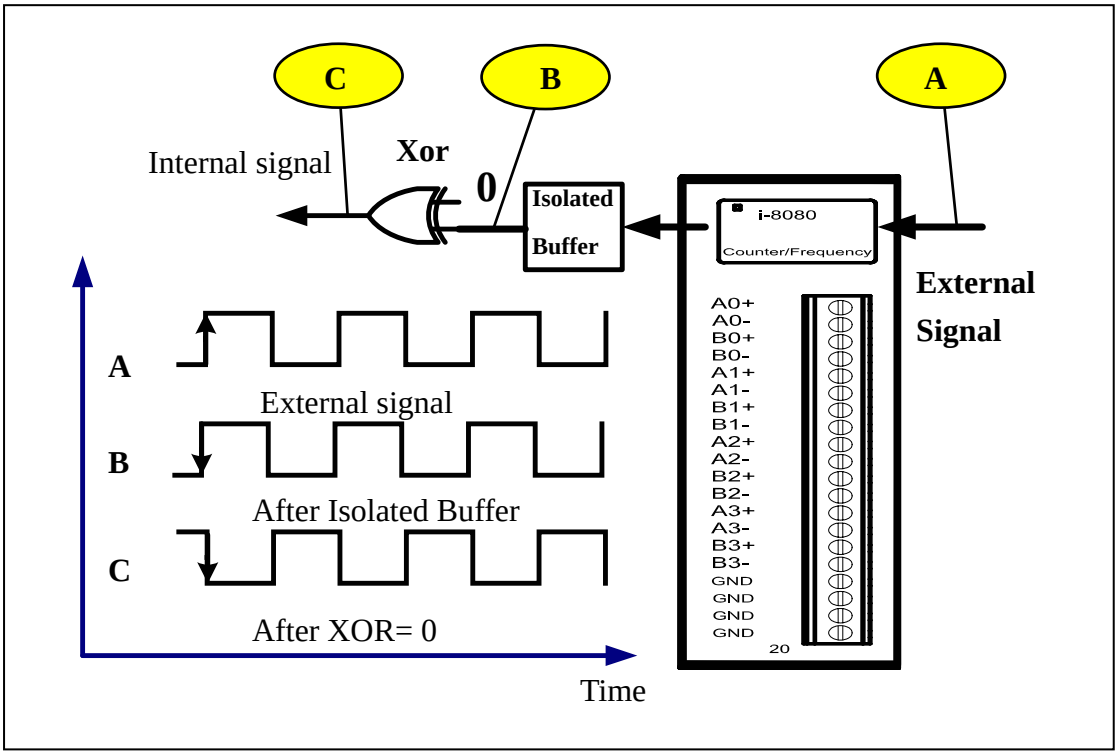


Figure 1-1 Isolated input

2. TTL Input (XOR=1)

When an external TTL signal is input into an 8080 module through the TTL mechanism, the signal will be the same as the external signal. This internal signal isn't the recommended waveform as it must execute the exclusive OR (XOR=1) operation. The solution is shown in Figure 1-2.

3. Always XOR=0

Regardless of whether the input signal is TTL or isolated, XOR is always set to 0, and the maximum count error can only be 1. XOR=0 can be used for all cases, if a 1-count error is acceptable.

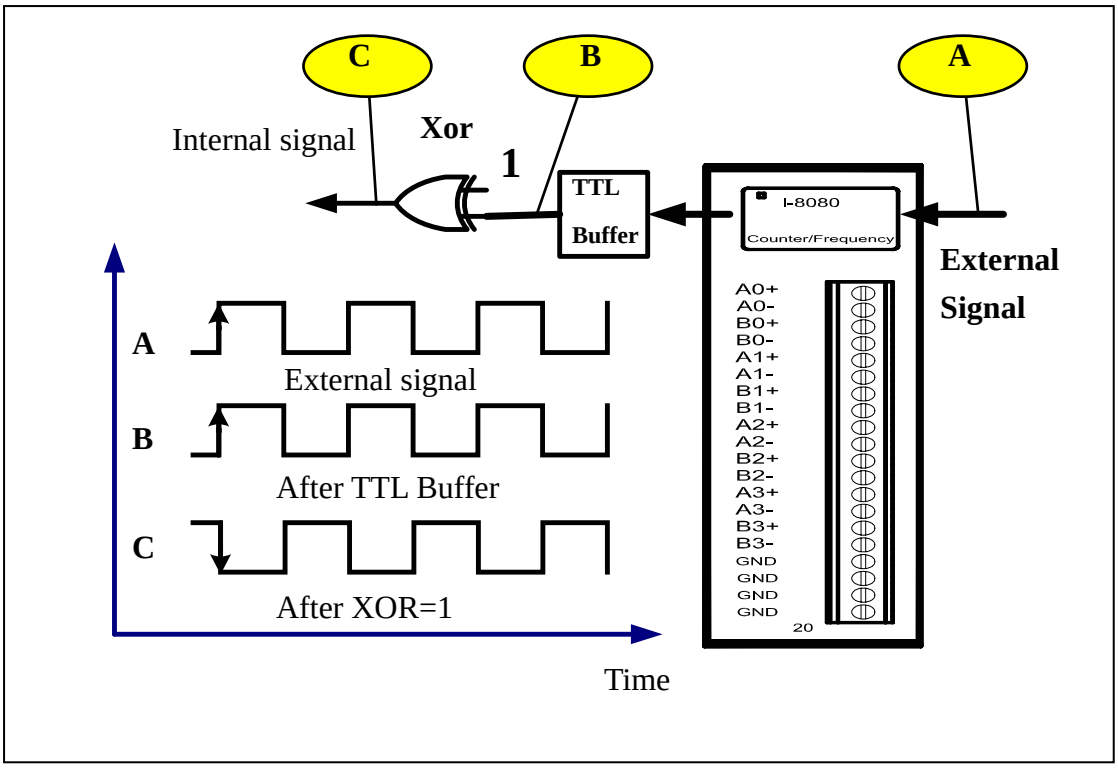


Figure 1-2 TTL input

Note:

1. When XOR=0 and the 8080 module status is OPEN status (i.e. no signals on the input terminal) ,regardless of whether you select the TTL or Isolated mode, the signal at the C point will always be 1. Similarly, if XOR=1 and the status is OPEN, then the signal at the C point will always be 0.
2. If the input signal is a pulse rather than a 50/50 duty cycle square waveform, then the 1-count error will not occur as the pulse width is shorter..

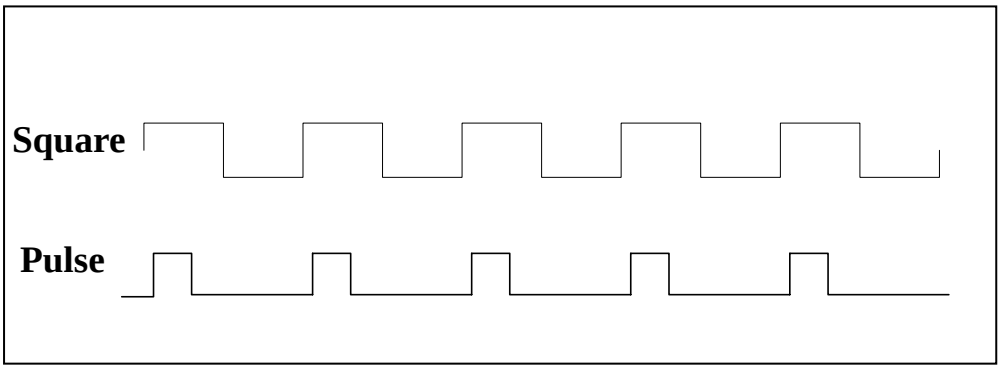


Figure 1-3 Square and pulse waveforms

1.2. Isolated or TTL Input

The input signal can be either isolated or TTL input. The isolated input block diagram is shown below:

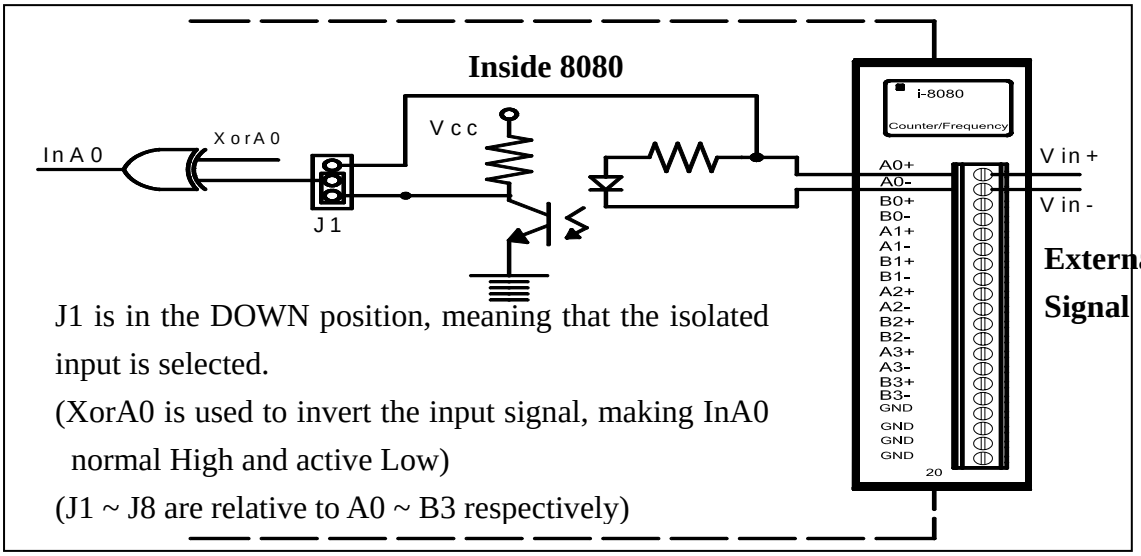


Figure 1-4 Isolated block diagram

The TTL input block diagram is as follows:

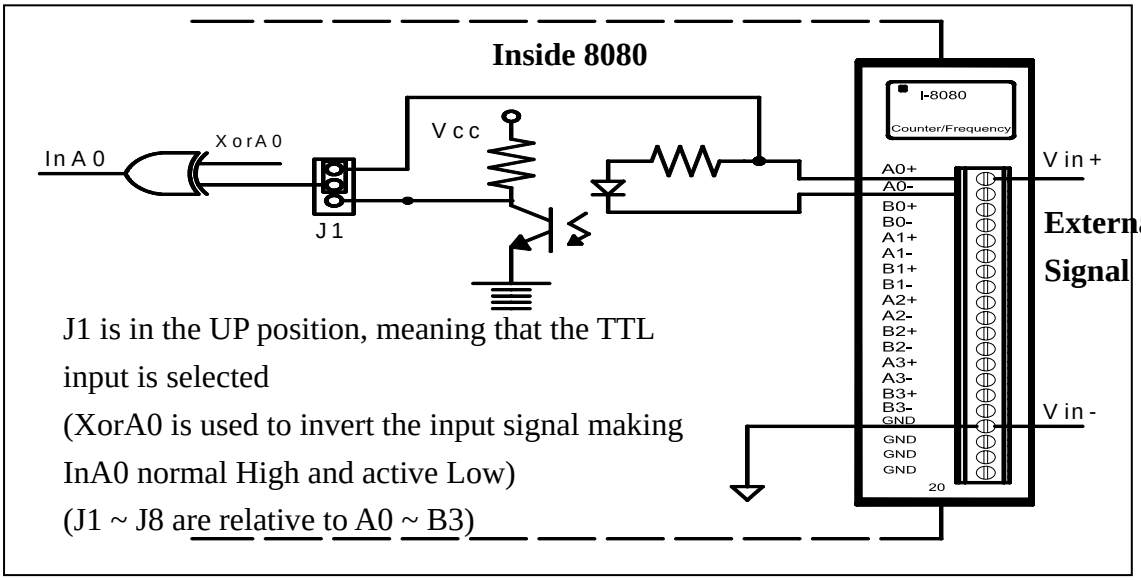
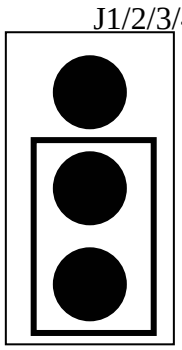


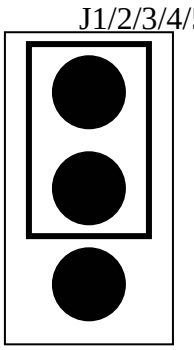
Figure 1-5 TTL block diagram

Isolated or TTL input is selected by using JP1 to JP8 as indicated below:

J1	Select A0
J2	Select B0
J3	Select A1
J4	Select B1
J5	Select A2
J6	Select B2
J7	Select A3
J8	Select B3



Select isolated input (default setting)



select TTL input

The default jumper settings are as follows:

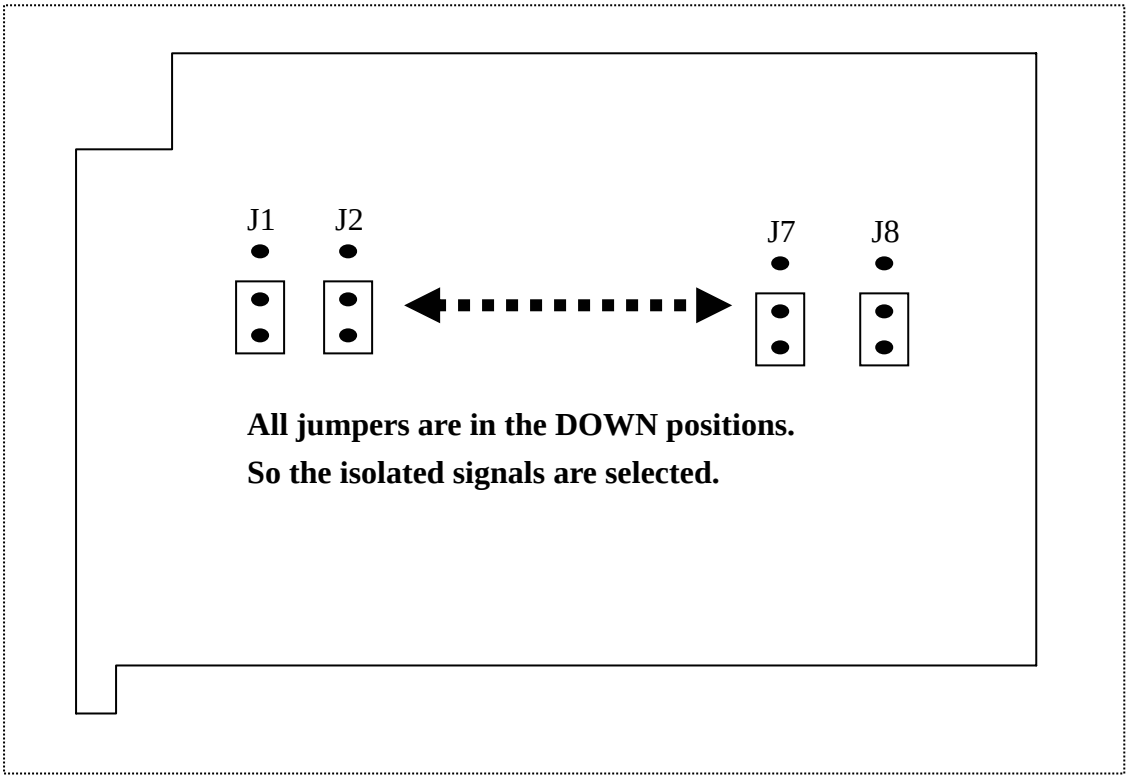


Figure 1-6 Isolated & TTL Jumper

Because JP1 to JP8 are independent, A0, B0, …, A3, B3 can be either isolated or TTL independently. That is to say, A0 can be isolated and B0 can be either isolated or TTL without limitation.

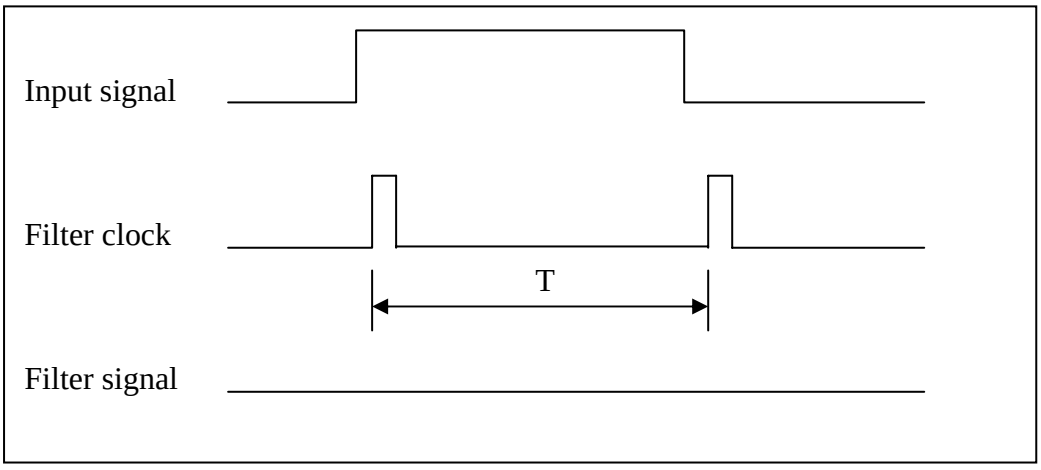
1.3. Digital Low Pass Filter

The 8080 has three independent 2nd-order digital noise filters, LP0, LP1 & LP2, to remove noises as follows:

Channel	Low Pass Filter
A0	Low Pass Filter 0 (Enabled or disabled)
B0	Low Pass Filter 0 (Enabled or disabled)
A1	Low Pass Filter 1 (Enabled or disabled)
B1	Low Pass Filter 1 (Enabled or disabled)
A2	Low Pass Filter 2 (Enabled or disabled)
B2	Low Pass Filter 2 (Enabled or disabled)
A3	Low Pass Filter 2 (Enabled or disabled)
B3	Low Pass Filter 2 (Enabled or disabled)

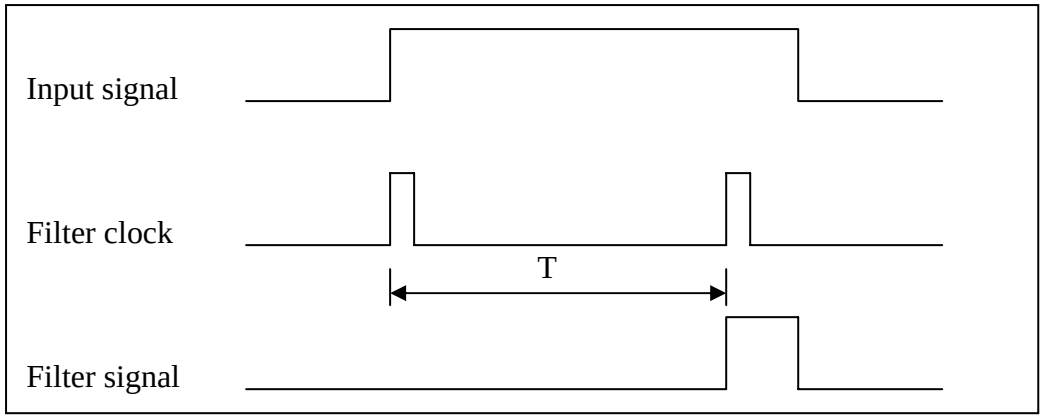
The Low Pass Filter can be either disabled or programmable from 2 μS to 65535 μS. The Low Pass Filter will apply to all working modes, counter or frequency. These 3 Low Pass Filters are disabled status in the default shipping. User defined program can be used to issue a command to enable or disable the filters.

Assume that the filter clock of the Low Pass Filter is set to T, this clock is used to sample the input signal. If one of the adjacent 2 samples is low, then the input signal will be removed as follows:



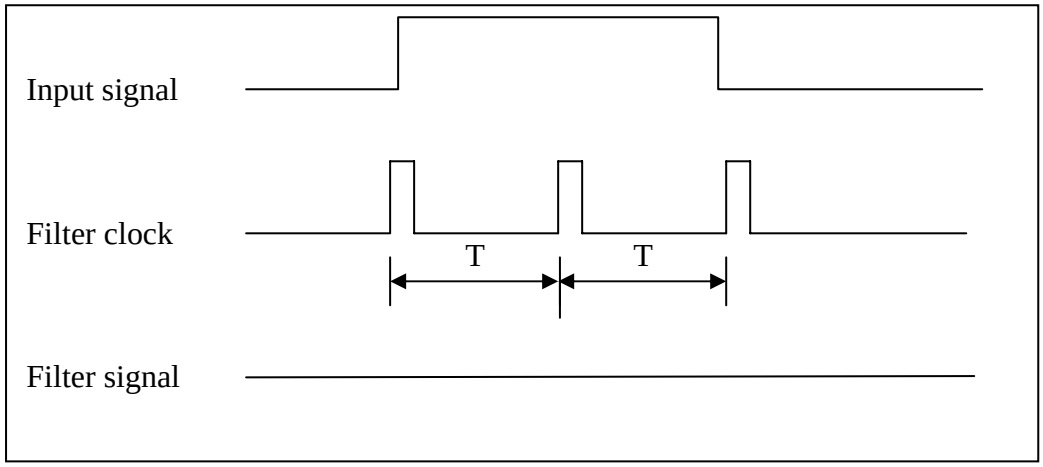
If the high width of the input signal is shorter then T, it will be filtered.

If the adjacent 2 samples are all HIGH, the input signal can pass as indicated below:



Note: the filter signal is shorter than the original input signal.

If the input signal is shorter than 2T, it may be filtered in the following manner:



The relationship between the input signal and the filter signal is as follows:

if (2T<input signal) → it will pass

if (T<=input signal<=2T) → it may be filtered or passed

if (input signal <T) → it will be filtered

The software driver, 'i8080_SetLowPassUs (int Slot, int Channel, unsigned int Us)', provides an parameter, Us which can be used to set the Low Pass Filter as follows:

Us=1 → 2T =1µS → T = 0.5µS → signal<=0.5µS will be removed

Us=2 → 2T= 2µS → T = 1µS → signal<=1µS will be removed

Us=N, N from 1 to 0x7fff → 2T = N µS → signal<=(N/2) µS will be removed

The Low Pass Filter range can be configured from 1µS to 32767µS. The high width of the signal < (Us/2) will be removed.

For example, if you use a function generator as signal source, the 500Hz signal & 50/50 duty cycle will generate a 1000µS high & 1000µS low as follows:

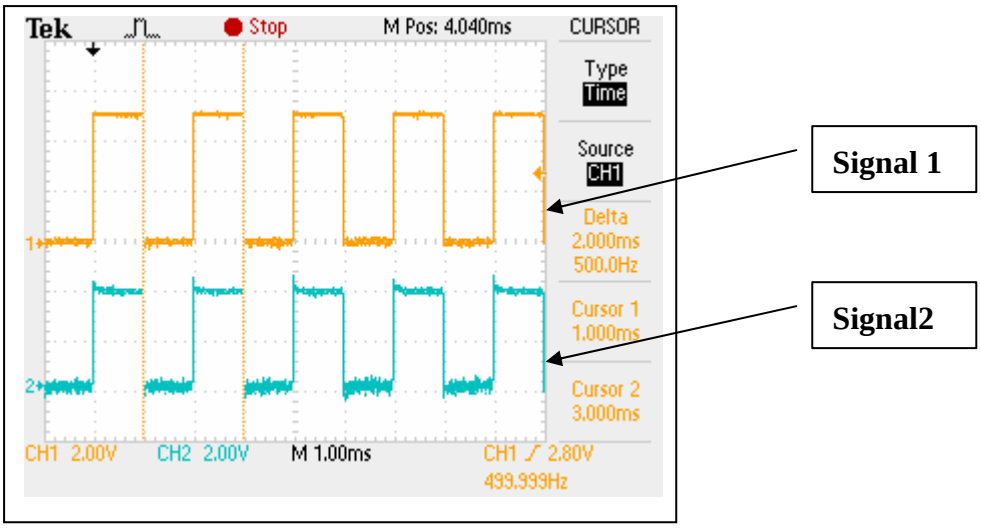


Figure 1-7 Input signal=500Hz & Low Pass Filter Disable

Signal 1 = input signal=500Hz, 50/50 duty cycle
Signal 2 = input signal after Xor and Low Pass Filter, now Xor=0 and Low Pass Filter is disable.
If the Low Pass Filter is disabled, signal 2 will be the same as signal 1 in the above diagram. If the Low Pass Filter is enabled, signal 2 will be shorter than signal 1 as shown below:

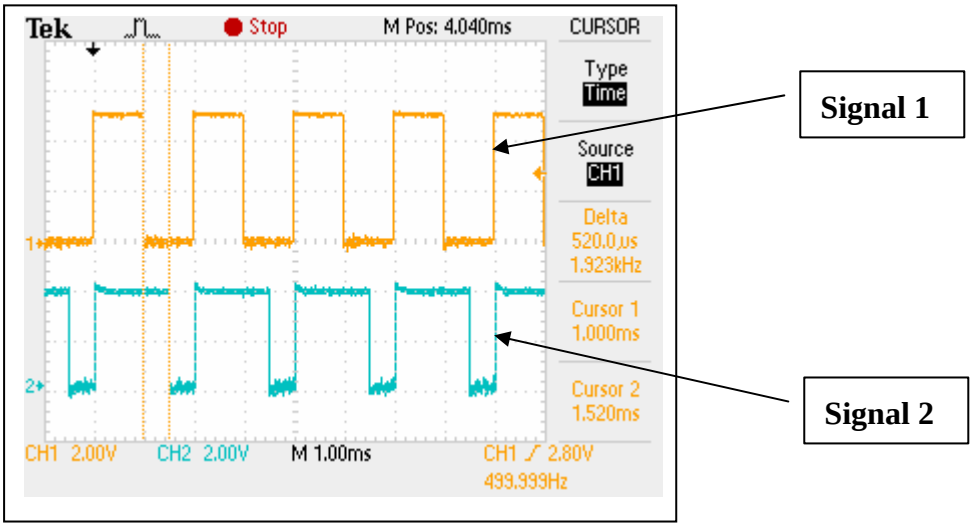


Figure 1-8 Input signal=500Hz & Low Pass Filter Enable=1µS

Signal 1 = input signal=500Hz, 50/50 duty cycle
Signal 2 = input signal after Xor and Low Pass Filter, now Xor=0 and the Low Pass Filter is enabled. Nearly all pulses are passed.

Now you can find that nearly all pulses are passed. If the input signal is increased to 600Hz, then some of the pulses are filtered as follows:

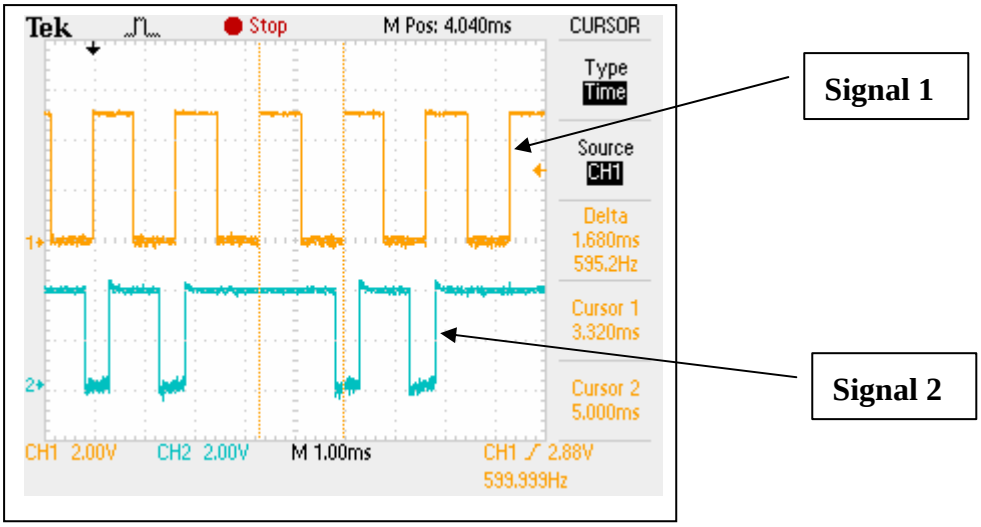


Figure 1-9 Input signal=600Hz & Low Pass Filter Enabled=1µS

Signal 1 = input signal=600Hz, 50/50 duty cycle.
Signal 2 = input signal after Xor and Low Pass Filter, now Xor =0 and Low Pass Filter is enabled.
Some pulses are filtered.
If the input signal is increased to 900Hz, then nearly all pulses are filtered as illustrated below:

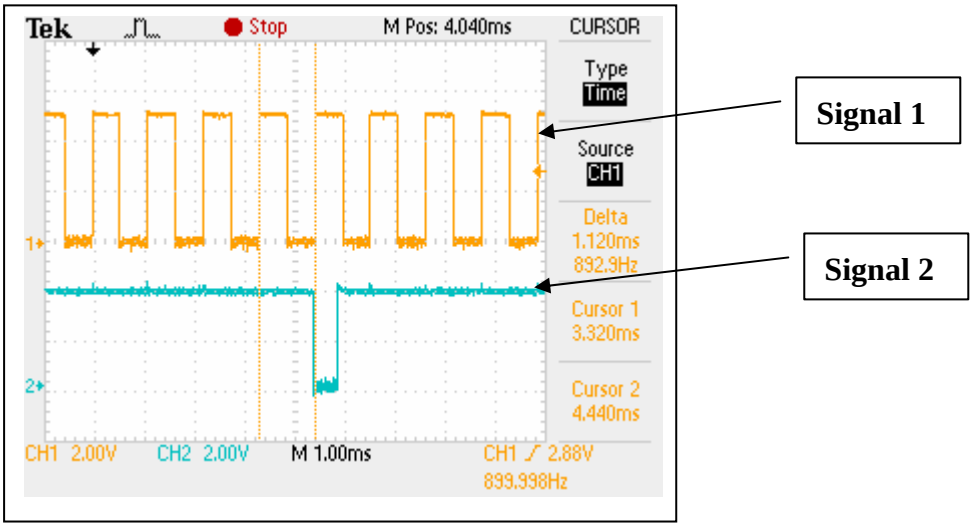


Figure 1-10 Input signal=900Hz & Low Pass Filter Enabled=1µS

Signal 1 = input signal=900Hz, 50/50 duty cycle
Signal 2 = input signal after Xor and Low Pass Filter, now Xor=0 and the Low Pass Filter is enabled. Nearly all pulses are filtered.

Because there are some frequency offset errors in the internal crystal, there may be some noises when the input signal width = Low Pass Filter/2 as follows:

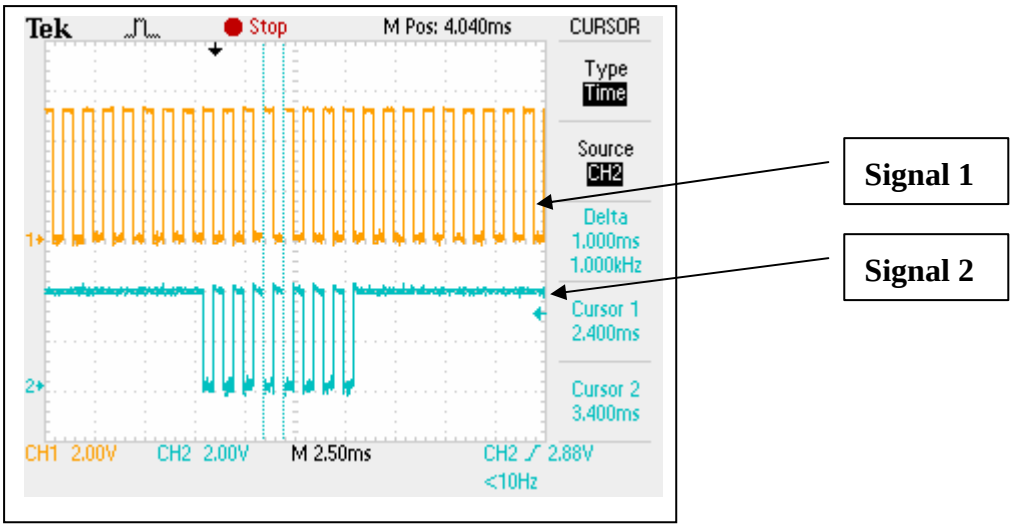


Figure 1-11 Input signal=1000Hz & Low Pass Filter Enabled=1µS

Signal 1 = input signal=1000Hz, 50/50 duty cycle → pulse width=500 µS
Signal 2 = input signal after Xor and Low Pass Filter, now Xor=0 and the Low Pass Filter is enabled.
Signal Pulse=500 µS=Low Pass Filter/2.
Nearly all pulses are filtered, but sometimes certain noises will not be filtered.
If the input signal is increased to 1100Hz, then all pulses will be filtered as shown in Figure 1-12:

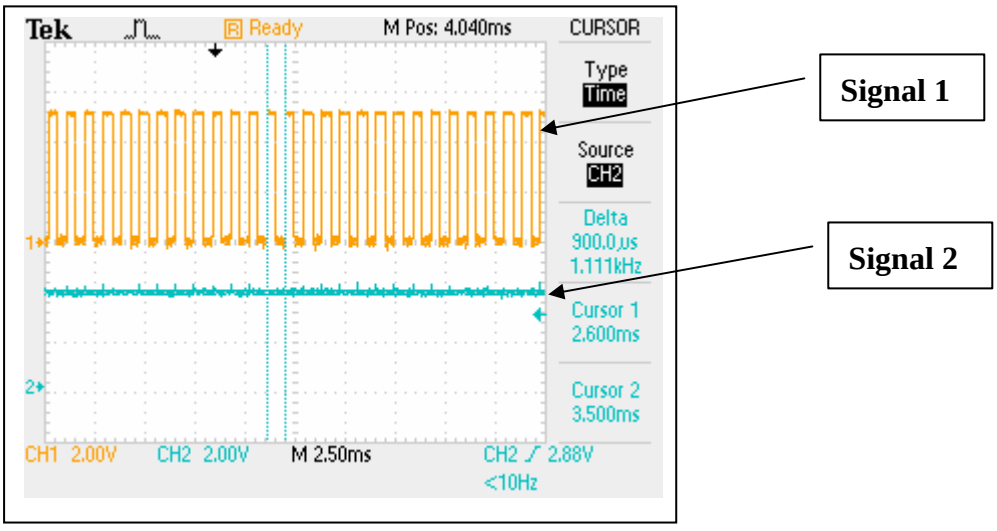


Figure 1-12 Input signal=1100Hz & Low Pass Filter Enabled=1µS

In summary, apply the minimum 1µS on Low Pass Filters.
The result of the signal being processed by the Low Pass Filter as follows:

Input signal frequency(Hz)	After Low Pass Filter processing	Reference
Input signal <500Hz (Low Pass Filter=1µS)	All signals will be passed	
Input signal =500Hz (Low Pass Filter=1µS)	All signals should be passed	**Figure 1-8**
Input signal =600Hz (Low Pass Filter=1µS)	Some signals will be filtered and some will be passed	**Figure 1-9**
Input signal =900Hz (Low Pass Filter=1µS)	Many signals will be filtered and few will be passed	**Figure 1-10**
Input signal =1000Hz (Low Pass Filter=1µS)	Nearly all signals are filtered	**Figure 1-11**
Input signal =1100Hz (>1k Hz) (Low Pass Filter=1µS)	All signals will be filtered	**Figure 1-12**

For the same reason, if the signal pulse=Low Pass Filter, certain pulses may be filtered. **Therefore, it is recommended to set the cycle time of Low Pass Filter about 5% less than the cycle time of input signal pulse** as shown below:
Input pulse =1 ms = 1000 µS → set Low Pass Filter <=950 µS
Input pulse = 100 µS → set Low Pass Filter <= 95 µS
The minimum Low Pass Filter = 1 µS → input signal < 475K, 50/50 duty cycle
As a result, the maximum speed of the 8080 is recommended to 450K, 50/50 duty cycle

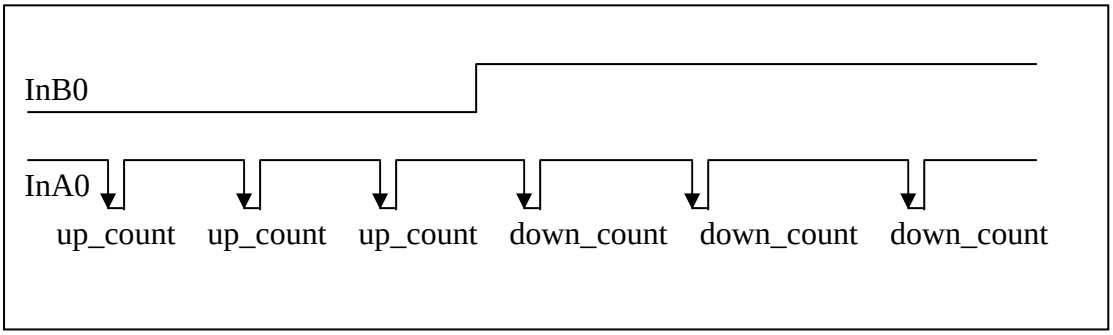
1.4. Operation Mode

Operation Mode	Description	Number of channels
00	Dir/Pulse counting mode	4 channels
01	Up/Down counting mode	4 channels
02	Frequency mode	8 channels
03	Up counting mode	8 channels

The input channels mapping table and working modes are indicated below:

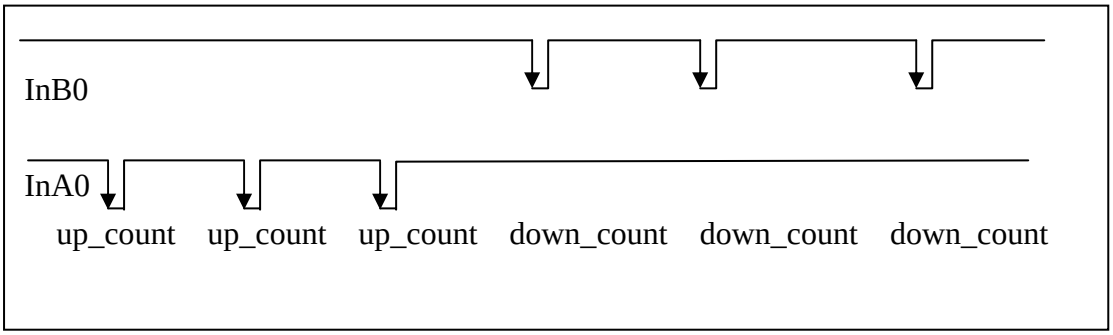
	Mode 00	Mode 01	Mode 02	Mode 03
A0 → InA0	Pulse 0	Up 0	Frequency 0	Up 0
B0 → InB0	Dir 0	Down 0	Frequency 1	Up 1
A1 → InA1	Pulse 1	Up 1	Frequency 2	Up 2
B1 → InB1	Dir 1	Down 1	Frequency 3	Up 3
A2 → InA2	Pulse 2	Up 2	Frequency 4	Up 4
B2 → InB2	Dir 2	Down 2	Frequency 5	Up 5
A3 → InA3	Pulse 3	Up 3	Frequency 6	Up 6
B3 → InB3	Dir 3	Down 3	Frequency 7	Up 7

The counter operation for mode 00 (Dir/Pulse mode) is as follows:



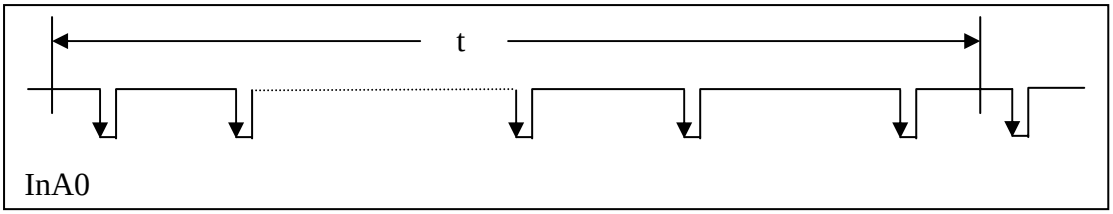
When InB0 is used as Dir, if InB0 is High, counter_0 will be increased by one for every falling edge of InA0. If InB0 is Low, counter_0 will be decreased by one for every falling edge of InA0.
InB1 and InA1 are Dir/Pulse pairs for counter_1
InB2 and InA2 are Dir/Pulse pairs for counter_2
InB3 and InA3 are Dir/Pulse pairs for counter_3
Note: A0 ~ 3 and B0 ~ 3 are input signal, InA0 ~ 3 and InB0 ~ 3 are input signals after Xor control. Refer to Sec. 1.1 for more information.

The counter operation for mode 01 (Up/Down mode) is as follows:



When InA0 is used as a UP_clock and InB0 is used as a DOWN_clock. The counter_0 will be increased by one for every falling edge of InA0 and decreased by one for every falling edge of InB0.
InA1 and InB1 are Up/Down pairs for counter_1
InA2 and InB2 are Up/Down pairs for counter_2
InA3 & InB3 are Up/Down pairs for counter_3
Note: A0 ~ 3 and B0 ~ 3 are input signals, InA0 ~ 3 and InB0 ~ 3 are input signals after Xor control. Refer to Sec. 1.2 for more information.

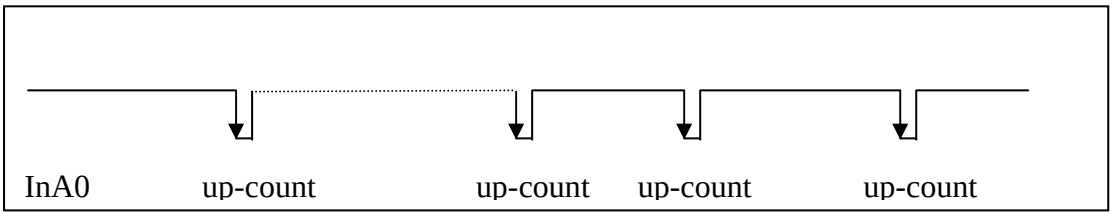
The frequency operation for mode 02 is as follows:



Counter_0 will be increased by one for every falling edge of InA0, the frequency of channel_0 = 1/ (t/number of count)
InB0 is the channel_1 frequency
InA1 is the channel_2 frequency
InB1 is the channel_3 frequency
InA2, InB2, InA3, InB3 are channels 4/5/6/7 for mode_02 only
Note 1: A0 ~ 3 and B0 ~ 3 are input signals, InA0 ~ 3 and InB0 ~ 3 are input signals after Xor control. Refer to Sec. 1.2 for more information.

2: t=0.1 second is the default setting. A user defined command can be used to change the value of t for special applications.

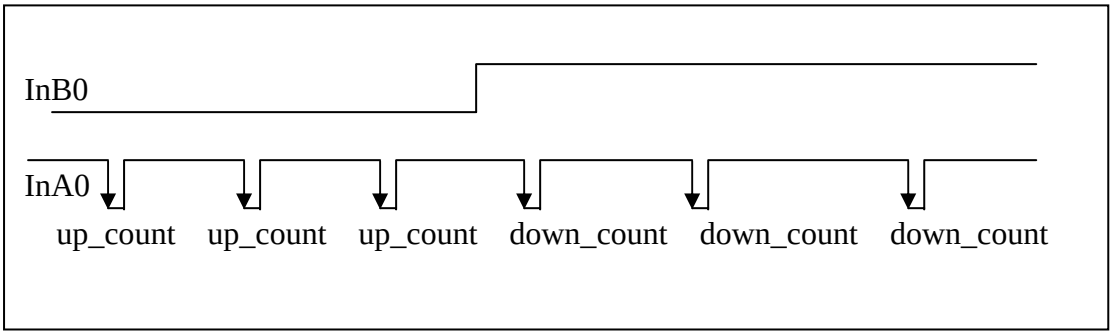
The counter operation for mode 03 is as follows:



Counter_0 will increment by one for every falling edge of InA0
InB0 is the channel_1 up counter
InA1 is the channel_2 up counter
InB1 is the channel_3 up counter
InA2, InB2, InA3, InB3 are Up counter, channel 4/5/6/7.
Note 1: A0 ~ 3 and B0 ~ 3 are input signals, InA0 ~ 3 and InB0 ~ 3 are input signals after Xor control. Refer to Sec. 1.2 for more information.

1.4.1. Mode 00: Dir/Pulse Counting

The counter operation for mode 00(Dir/Pulse mode) is shown below:



The Dir/Pulse input signals will be converted to up/down clocks and will be connected to Up/Down counter. The counter operation is given as follows:

00000000 → 00000001 → ……. → 7FFFFFFE → 7FFFFFFF

MSB=0 --> Up-counting, OverflowN = OverflowN + 1

00000000 → 80000001 → ……. → 8FFFFFFE → 8FFFFFFF

MSB=1 --> Down-counting, OverflowN = OverflowN+1

Channel	Counting Variable	Total Counting Value
Channel 0	Count0, Overflow0	Count0 + Overflow0 * 0x80000000
Channel 1	Count1, Overflow1	Count1 + Overflow1 * 0x80000000
Channel 2	Count2, Overflow2	Count2 + Overflow2 * 0x80000000
Channel 3	Count3, Overflow3	Count3 + Overflow3 * 0x80000000

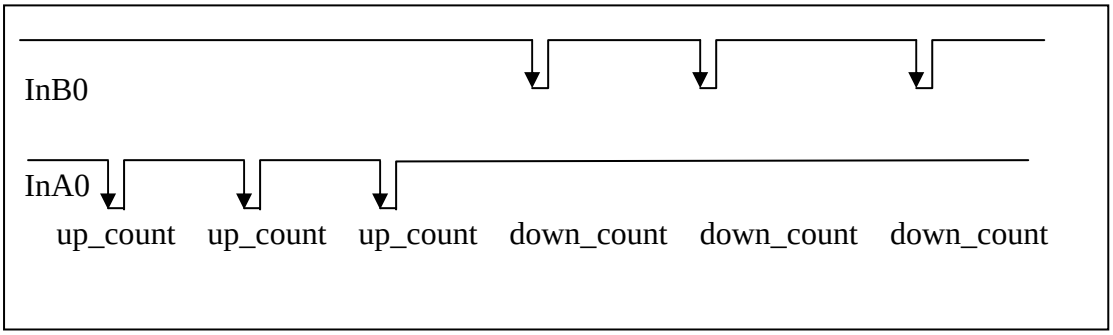
- CountN = current counter value for channel N, 32-bit wide, MSB=0 → up counting, MSB=1 → down counting
- OverflowN = the counting overflow number for channel N, 16-bit wide, from 0 to 0xFFFF
- Total Counting Value = 31-bit + 16-bit = 47-bit
  (31-bit denotes 15-bit software and 16-bit hardware counter)

Some examples are as follows:

OverflowN	CountN	Total Counting Value	Up/Down Counting
0	0	0	Initial Position
0	1	1	Up Counting, 1 count
0	0X80000001	1	Down Counting, 1 count
0	2	2	Up Counting, 2 counts
0	0X80000002	2	Down Counting, 2 counts
……..	…….	…….	…….
N	MSB=0	CountN+0X8000000*N	Up Counting
N	MSB=1	CountN & 0X7FFFFFFF + 0X8000000*N	Down Counting
……..	…….	…….	…….
0	0X7FFFFFFF	0X7FFFFFFF	Up Counting, 0X7FFFFFFF counts
1	0	0+0X80000000	Up Counting, 0X80000000 counts
0	0XFFFFFFFF	0X7FFFFFFF	Down Counting, 0X7FFFFFFF counts
1	0X80000000	0+0X80000000	Down Counting, 0X80000000 counts
1	1	1 + 0X80000000*1	Up Counting,, 0X80000001 counts
2	2	2 + 0X80000000*2	Up Counting, 0X100000002 counts
1	0X80000001	1 + 0X80000000*1	Down Counting, 0X80000001 counts
2	0X80000002	2 + 0X80000000*2	Down Counting, 0X100000002 counts

1.4.2. Mode 01: Up/Down Counting

The counter operation for mode 01 (Up/Down mode) is shown below:



The counter operation is as follows:

00000000 → 00000001 → ……. → 7FFFFFFE → 7FFFFFFF

MSB=0 --> Up-counting, OverflowN = OverflowN + 1

00000000 → 80000001 → ……. → FFFFFFFE → FFFFFFFF

MSB=1 --> Down-counting, OverflowN = OverflowN - 1

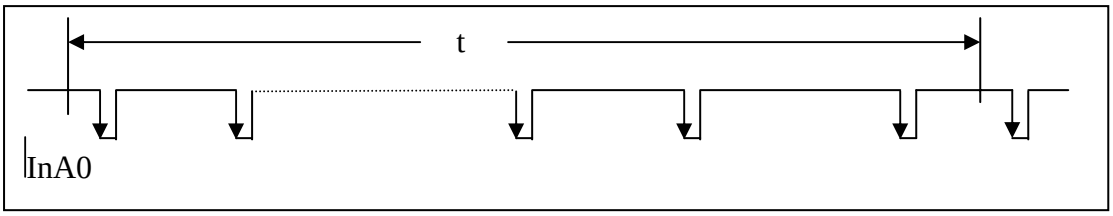
- CountN = current counter value for channel N, 32-bit wide, MSB=0 → up counting, MSB=1 → down counting
- OverflowN = The counting overflow number for channel N, 16-bit wide, from 0 to 0xFFFF
- Total Counting Value = 31-bit + 16-bit = 47-bit
  (31-bit denotes 15-bit software and 16-bit hardware counter)

Some examples are as a follows:

OverflowN	CountN	Total Counting Value	Up/Down Counting
0	0	0	Initial Position
0	1	1	Up Counting, 1 count
0	0X80000001	1	Down Counting, 1 count
0	2	2	Up Counting, 2 counts
0	0X80000002	2	Down Counting, 2 counts
……..	…….	…….	…….
N	MSB=0	CountN+0X8000000*N	Up Counting
N	MSB=1	CountN & 0X7FFFFFFF + 0X8000000*N	Down Counting
……..	…….	…….	…….
0	0X7FFFFFFF	0X7FFFFFFF	Up Counting, 0X7FFFFFFF counts
1	0	0+0X80000000	Up Counting, 0X80000000 counts
0	0XFFFFFFFF	0X7FFFFFFF	Down Counting, 0X7FFFFFFF counts
1	0X80000000	0+0X80000000	Down Counting, 0X80000000 counts
1	1	1 + 0X80000000*1	Up Counting, 0X80000001 counts
2	2	2 + 0X80000000*2	Up Counting, 0X100000002 counts
1	0X80000001	1 + 0X80000000*1	Down Counting, 0X80000001 counts
2	0X80000002	2 + 0X80000000*2	Down Counting, 0X100000002 counts

1.4.3. Mode 02: Frequency Mode

The frequency operation for mode 02 is shown below:



Frequency = 1 / (t/count)

Assume t = 0.1 seconds,

If count = 1 → frequency = 1/(0.1/1) = 10 Hz

If count = 10 → frequency = 1/(0.1/10) = 100 Hz

All frequency channels will be updated every 0.1 seconds for t= 0.1 seconds.

The software driver provides three ways to adjust t.

They are Auto select, Low and High Frequency. (The default is Auto select)

The default configuration data is as follows:

AutoTT = 330millisecond (ms), AutoVV = 10;

LowTT = 1000millisecond(ms), AutoVV = 3;

HighTT = 100millisecond(ms), AutoVV = 30;

Please refer to the “I8080 software user’s manual (C language)” for more details.

The Frequency Measurement Algorithm

step 1 : start timer

step 2 : read counts

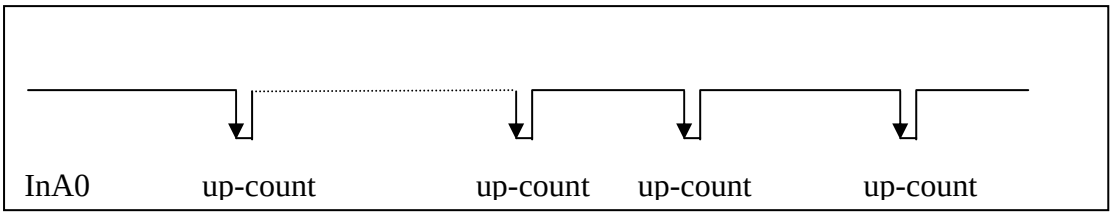
step 3 : if count > ModeVV then update frequency

step 4 : if timer > ModeTT then go to step 1

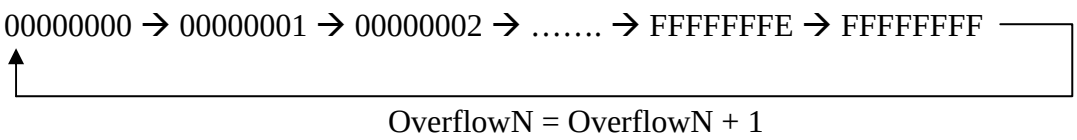
step 5 : go to step 2

1.4.4. Mode 03: Up Counting

The counter operation for mode 03 is shown below:



The counter operation is as follows:



Channel	Counting Variable	Total Counting Value
Channel 0	Count0, Overflow0	Count0 + Overflow0 * 0X100000000
Channel 1	Count1, Overflow1	Count1 + Overflow1 * 0X100000000
Channel 2	Count2, Overflow2	Count2 + Overflow2 * 0X100000000
Channel 3	Count3, Overflow3	Count3 + Overflow3 * 0X100000000
Channel 4	Count4, Overflow4	Count4 + Overflow4 * 0X100000000
Channel 5	Count5, Overflow5	Count5 + Overflow5 * 0X100000000
Channel 6	Count6, Overflow6	Count6 + Overflow6 * 0X100000000
Channel 7	Count7, Overflow7	Count7 + Overflow7 * 0X100000000

- CountN = current counter value for channel N, 32-bit wide, initial 0
- OverflowN = The counting overflow number for channel N, 16-bit wide, from 0 to 0XFFFF
- Total Counting Value = 32bit + 16-bit = 48bit

(32bit denotes 16-bit software and 16-bit hardware counter, No MSB bit)

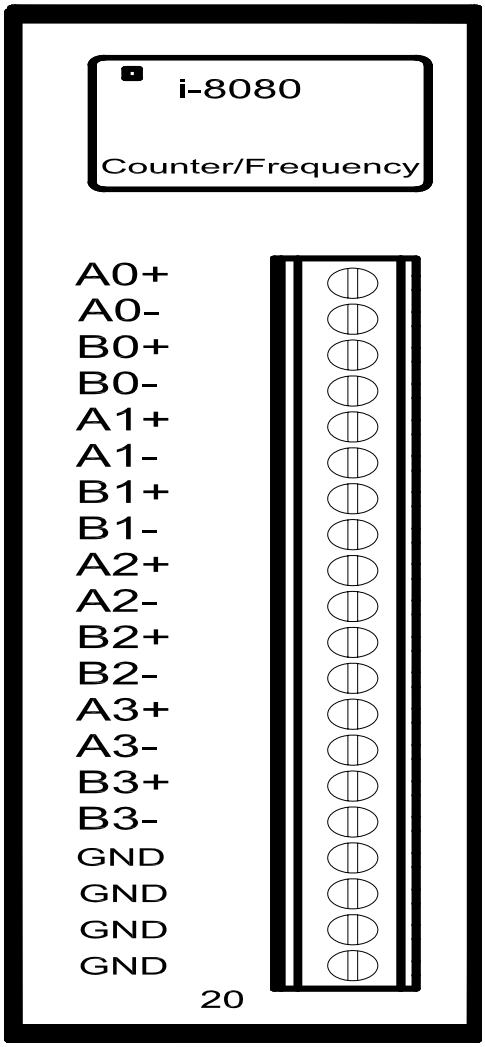
Some examples are as follows:

OverflowN	CountN	Total Up Counting Value
0	0	0, initial value
0	1	1
1	0	0X100000000
1	1	0X100000001
2	2	0X200000002

1.5. EEPROM

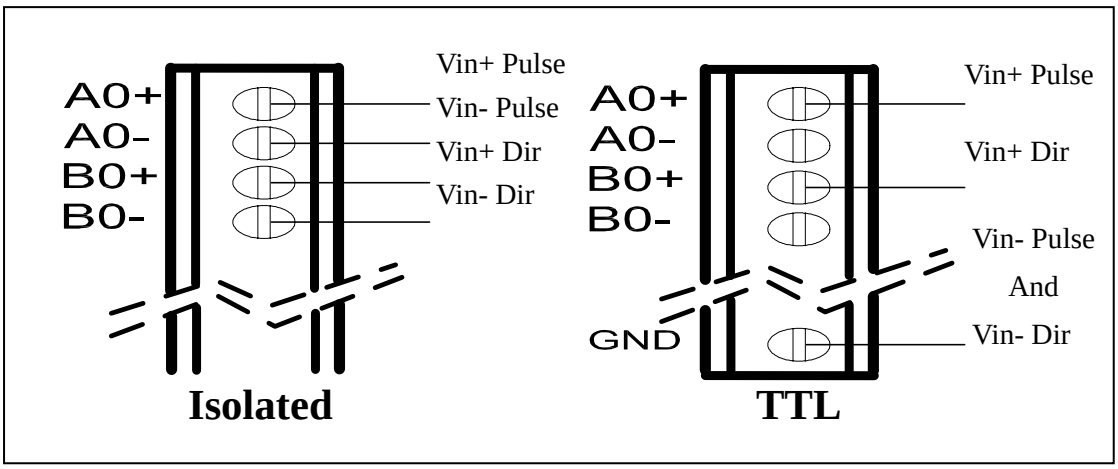
The 8080 is equipped with 128bytes of EEPROM, allowing the user to save both system and channel configuration data. The limitation of EEPROM can be rewrite 1,000,000 times.

2. Pin Assignments

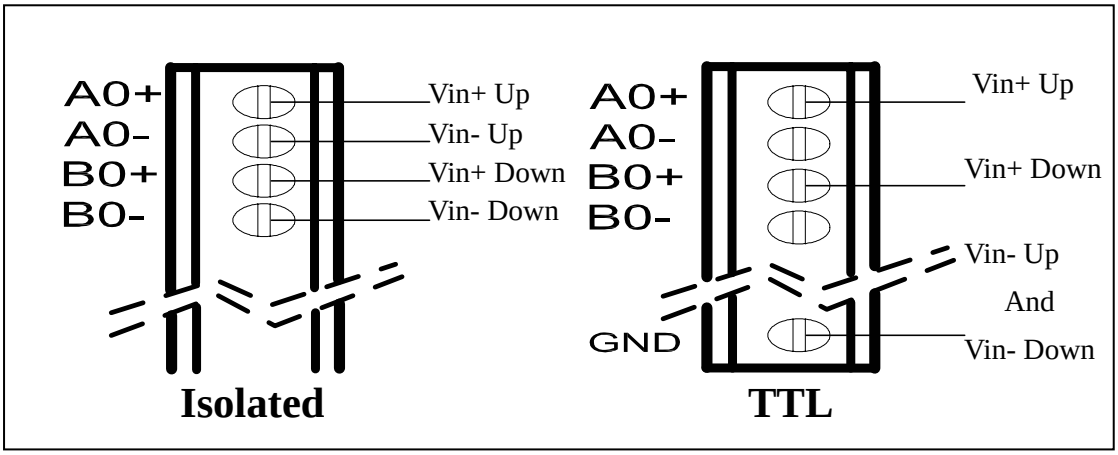


2.1. I-8080 Application Wiring

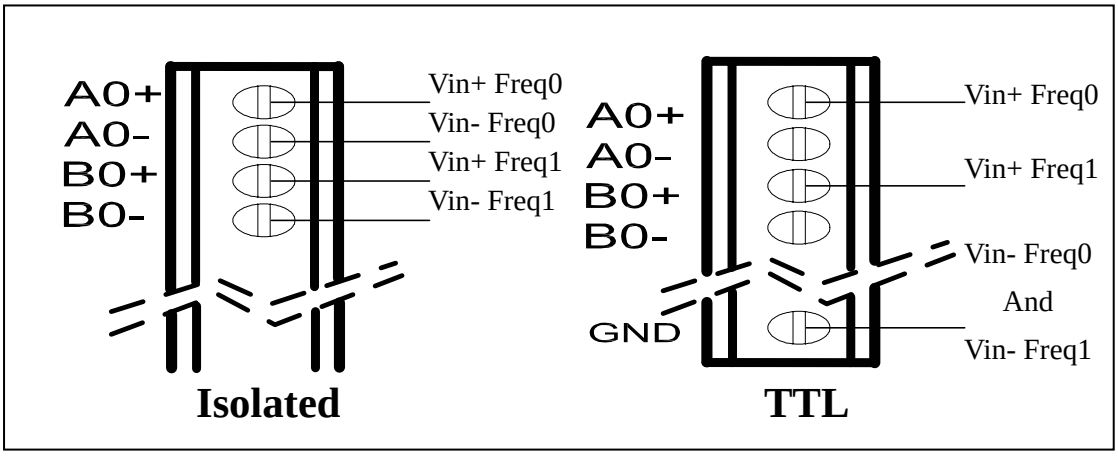
Mode0: Dir/Pulse Counter Mode



Mode1: Up/Down Counter Mode



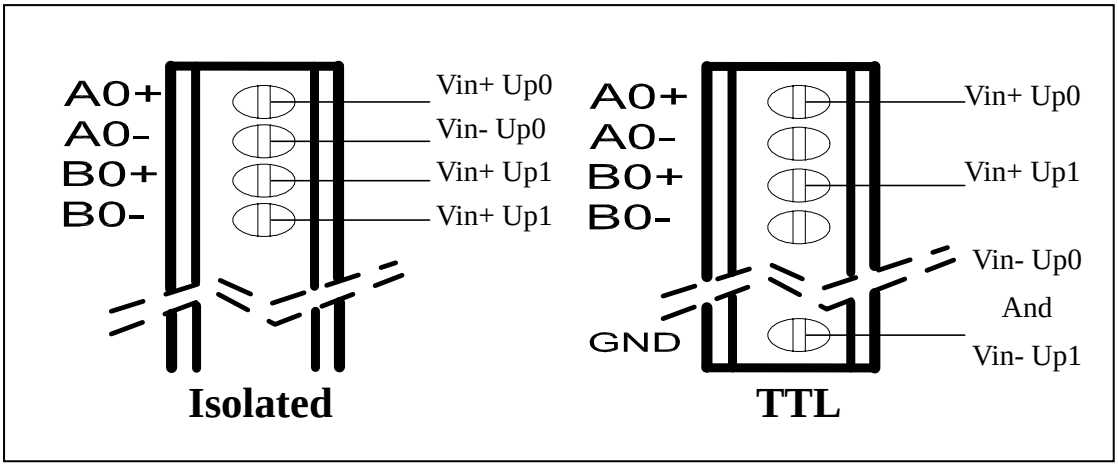
Mode2: Frequency Mode



Freq0: The first channel of Frequency mode (8 channels, channel0~7)

Freq1: The second channel of Frequency mode

Mode3: Up Counter Mode



Up0: The first channel of Up Counter mode (8 channels, channel0~7)

Up1: The second channel of Up Counter mode