九齊科技股份有限公司

Nyquest Technology Co., Ltd.

USER MANUAL

NY8L Series

8-bit 65C02 MCU with LCD Driver, 13~32 I/O & '1-Ch Speech + Dual-Tone'

Version 1.3

Mar. 8, 2019

NYQUEST TECHNOLOGY CO. reserves the right to change this document without prior notice. Information provided by NYQUEST is believed to be accurate and reliable. However, NYQUEST makes no warranty for any errors which may appear in this document. Contact NYQUEST to obtain the latest version of device specifications before placing your orders. No responsibility is assumed by NYQUEST for any infringement of patent or other rights of third parties which may result from its use. In addition, NYQUEST products are not authorized for use as critical components in life support devices/systems or aviation devices/systems, where a malfunction or failure of the product may reasonably be expected to result in significant injury to the user, without the express written approval of NYQUEST.

Revision History

Version	*Date*	*Description*	*Modified Page*
1.0	2017/08/31	Formal release.	-
1.1	2017/11/28	Delete KSB wake-up source for Halt mode.	17, 20, 56
1.2	2018/05/31	SFR rename: 1. Update DC characteristics. 2. PXD rename to PX. 3. TM0 rename to TM0D. 4. TM1 rename to TM1D. 5. TM2 rename to TM2D. 6. ADCSEL rename to ADCMD. 7. ADCCTRL rename to ADCC. 8. ADCO rename to ADCD. 9. LCDPWR rename to LCDPC. 10.LCD rename to LCDC. 11.SPIC rename to SPIMD. 12. AUDFUNC rename to AUD. 13. IOC rename to IRC. 14. PAD rename to PA. 15. PBD rename to PB. 16. PCD rename to PC. 17. PDD rename to PD. 18. ON/OFF rename to Enable/Disable. 19. PL ON/OFF rename to Pull-Low Resistor Enable/Disable.	14, 15 20, 53, 55 29, 47, 48 29, 30, 47, 48 30, 47 31, 65, 66 32, 65, 66 32, 64, 65, 66 33, 36 33, 35, 39 33, 58, 59 34, 46 34, 60, 63 34, 52, 53, 55 34, 53, 55 35, 53, 55, 56 35, 53, 55 22, 29, 30, 33, 34, 36, 45, 46, 48, 57, 58 35, 53
1.3	2019/03/08	1. Modify NY8L040A LCD share pin to COM4~11 2. Modify “1.6.3 DC Characteristics” 3. Modify “8.3.2 stop mode1” & Figure 8-5 4. Modify “Chapter 9.Application Circuits”	12 15 61, 62 68~71

Table of Contents

Chapter 1. Introduction

1.1 General Description

The NY8L IC is a high-performance 8-bit 65C02 micro-controller with LCD driver and sound processor, three sets of 8-bit timer/counter, 13~32 general I/Os. For LCD driver, it applies for the most common-used LCD panel and functions as a key strobe for further application.

There are 3 channels that are configured as 1-channel speech and dual-tone melody (by body), and all of them can be auto-played back simultaneously. Not only high fidelity ADPCM speech synthesis algorithm built-in, but also wide range sampling rate up to 16KHz@4MHz and different volume level are supported. Those combinations create remarkable high-quality voice. NY8L provides two kinds of audio outputs with fine resolution (by body), one is 10-bit current-type D/A converter (DAC) and the other is 9-bit Pulse-Width-Modulation (PWM).

The NY8L IC provides 4 channels high-precision 8-bit analog-to-digital converter (ADC) (by body), which is suitable for any analog interface detection and measurement applications.

The CISC MCU architecture is very easy to program and control, various applications can be easily implemented. Furthermore, in addition to the Slow mode, it offers the Standby mode and Halt mode (or Sleep mode) to minimize power dissipation.

1.2 Features

- Wide operating voltage range: 1.1V ~ 3.6V / 2.0V ~ 5.5V. (by body)
  - 1.1V ~ 3.6V @ System clock ≤ 500KHz (NY8L005A, NY8L010A, NY8L020A).
  - 2.0V ~ 3.6V @ System clock ≤ 4MHz (NY8L005A, NY8L010A, NY8L020A).
  - 2.0V ~ 5.5V @ System clock ≤ 4MHz (NY8L030A, NY8L040A, NY8L050A).
  - 3.0V ~ 5.5V @ System clock ≤ 8MHz (NY8L030A, NY8L040A, NY8L050A).
- 4KB~128KB ROM (by body), program and voice data share the same ROM space.
- 64B~320B RAM. (by body)
- Dual-clock oscillation: System clock can switch between high oscillation and low oscillation.
  - High oscillation: IOSC8M/IOSC4M/IOSC2M/IOSC500K.
  - Low oscillation: IOSC32K/XTAL32K.
- Built-in RC oscillation is accurate with +/- 1.5% frequency deviation.
- Four kinds of operation mode to reduce system power consumption:
  - Normal mode, Slow mode, Standby mode and Halt mode.
- At Normal mode, CPU clock is software programmable.
  - 1/1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128 of high oscillator (F_{FAOS}) frequency.

- Slow mode to operate with low power consumption (+/-3% accuracy).
- Three 8-bit timers for 1-channel speech and 2-channel tone or other applications such as RFC.
- support most of LCD panel types:
  - 1/2, 1/3, 1/4 bias (by body).
  - 1/2, 1/3, 1/4, 1/5, 1/6, 1/7, 1/8, 1/9, 1/10, 1/11, 1/12 duty (by body).
- Charge pump for the LCD display power.
- LED sink/drive configuration supported through COM & SEG (by option).
- Matrix key strobe supported.
- Serial peripheral interface (SPI) master mode for serial Flash/SRAM memory
- Electroluminescent (EL) back light driver block supported with various frequency and duty.
- Resistance to Frequency Converter (RFC) for the detection of humidity, temperature or other applications.
- Low voltage reset (by body), watch-dog reset (by option) and external reset pin (by option) are all supported to protect the system.
- Flexible I/Os maximum with optional function: floating input, pull-low input, CMOS output, open-drain output.
- Infrared (IR) output: optional carrier frequency and optional data high/low output supported.
- Maximum of 3 channels can play simultaneously (by body); channel0/1 can be assigned as tone, channel2 can be assigned as speech.
- New high fidelity ADPCM speech synthesis algorithm.
- High quality direct-drive 9-bit PWM and 10-bit D/A converter audio output. (by body)
- 8 or 16 steps volume control for audio output.
- 4 channels high-precision 8-bit analog-to-digital converter (ADC). (by body)
- 7~8 interrupt modes supported (by body).

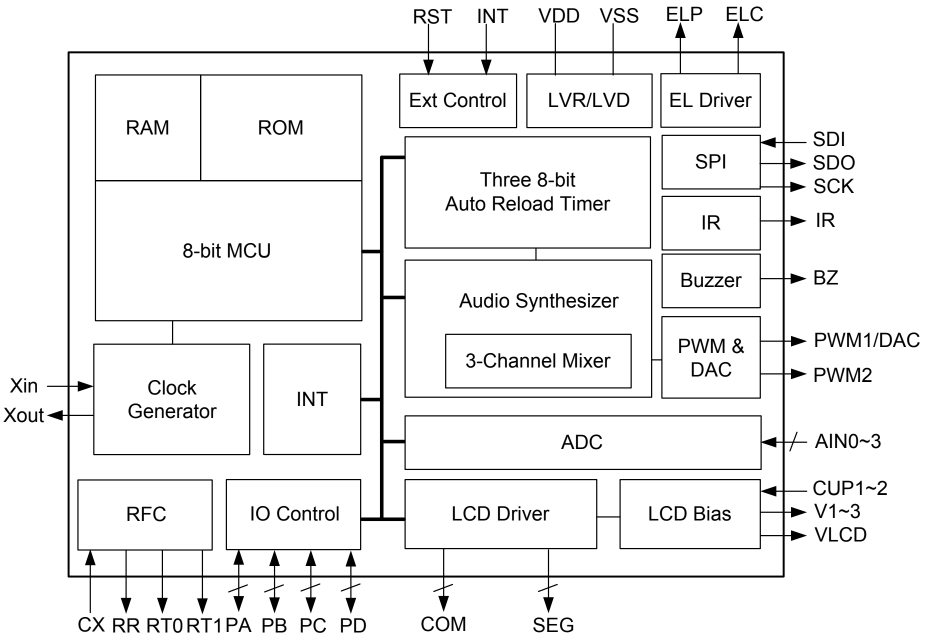
1.3 Product List

P/N	Voltage (V)	Voice Duration[1] @6KHz(sec)	ROM (bits)	RAM (bits)	I/O	LCD Dots[2] (COM x SEG)	Buzzer	PWM	DAC	ADC
NY8L005A	1.1 ~ 3.6	1.0	4K x 8	64 x 8	13	4 x 10	1	-	-	-
NY8L010A	1.1 ~ 3.6	2.3	8K x 8	64 x 8	16	6 x 23	1	-	-	-
NY8L020A	1.1 ~ 3.6	4.7	16K x 8	128 x 8	20	8 x 31	1	-	-	-
NY8L030A	2.0 ~ 5.5	10	32K x 8	128 x 8	20	12 x 33	-	9-bit	-	-
NY8L040A	2.0 ~ 5.5	20.7	64K x 8	192 x 8	24	12 x 43	-	9-bit	10-bit	4-ch
NY8L050A	2.0 ~ 5.5	42	128K x 8	320 x 8	32	12 x 55	-	9-bit	10-bit	4-ch

Note: 1. The voice duration is calculated at 6KHz by 4-bit ADPCM algorithm.

2. The LCD dots are calculated with maximal common pins, which can be shared with segment pins by options.

1.4 Block Diagram



1.5 Pad Description

1.5.1 Pad Description for NY8L005A

Pad Name	ATTR.	Description
VDD	P	Positive supply power.
VSS	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX	I/O	Bit 4 for Port A (can be used as key strobe input), or input of RFC function.
PA5/RR	I/O	Bit 5 for Port A (can be used as key strobe input), or output of RFC function.
PA6/RT0	I/O	Bit 6 for Port A (can be used as key strobe input), or output of RFC function.
PA7/RT1	I/O	Bit 7 for Port A (can be used as key strobe input), or output of RFC function.
PB0/BZ	I/O	Bit 0 for Port B, or buzzer output.
PB1/IR	I/O	Bit 1 for Port B, or IR output.
SEG0/PB2	I/O	LCD segment 0, or Bit 2 for Port B.
SEG1/PB3	I/O	LCD segment 1, or Bit 3 for Port B.
SEG2/PB4	I/O	LCD segment 2, or Bit 4 for Port B.
SEG3~9	O	LCD segment 3~9 (SEG3~8 can be used as key strobe output).
COM0~3	O	LCD common 0~3 (COM2~3 can be used as SEG11~10).
V1, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：31 Pins

1.5.2 Pad Description for NY8L010A

Pad Name	ATTR.	Description
VDD	P	Positive supply power.
VSS	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX	I/O	Bit 4 for Port A (can be used as key strobe input), or input of RFC function.
PA5/RR	I/O	Bit 5 for Port A (can be used as key strobe input), or output of RFC function.
PA6/RT0	I/O	Bit 6 for Port A (can be used as key strobe input), or output of RFC function.
PA7/RT1	I/O	Bit 7 for Port A (can be used as key strobe input), or output of RFC function.
PB0/BZ	I/O	Bit 0 for Port B, or buzzer output.
PB1	I/O	Bit 1 for Port B.
SEG0/PB2/ELP	I/O	LCD segment 0, Bit 2 for Port B, or charging signal of EL driver.
SEG1/PB3/ELC	I/O	LCD segment 1, Bit 3 for Port B, or discharging signal of EL driver.
SEG2/PB4/IR	I/O	LCD segment 2, Bit 4 for Port B, or IR output.
SEG3/PB5/SCK	I/O	LCD segment 3, Bit 5 for Port B, or clock output of SPI.
SEG4/PB6/SDI	I/O	LCD segment 4, Bit 6 for Port B, or data input (MISO) of SPI.
SEG5/PB7/SDO	I/O	LCD segment 5, Bit 7 for Port B, or data output (MOSI) of SPI.
SEG6~22	O	LCD segment 6~22 (SEG6~17 can be used as key strobe output).
COM0~5	O	LCD common 0~5 (COM2~5 can be used as SEG26~23).
V1~2, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：47 Pins

1.5.3 Pad Description for NY8L020A

Pad Name	ATTR.	Description
VDD	P	Positive supply power.
VSS	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX	I/O	Bit 4 for Port A, or input of RFC function.
PA5/RR	I/O	Bit 5 for Port A, or output of RFC function.
PA6/RT0	I/O	Bit 6 for Port A, or output of RFC function.
PA7/RT1	I/O	Bit 7 for Port A, or output of RFC function.
PB0/BZ	I/O	Bit 0 for Port B, or buzzer output.
PB1	I/O	Bit 1 for Port B.
SEG0/PB2/ELP	I/O	LCD segment 0, Bit 2 for Port B, or charging signal of EL driver.
SEG1/PB3/ELC	I/O	LCD segment 1, Bit 3 for Port B, or discharging signal of EL driver.
SEG2/PB4/IR	I/O	LCD segment 2, Bit 4 for Port B, or IR output.
SEG3/PB5/SCK	I/O	LCD segment 3, Bit 5 for Port B, or clock output of SPI.
SEG4/PB6/SDI	I/O	LCD segment 4, Bit 6 for Port B, or data input (MISO) of SPI.
SEG5/PB7/SDO	I/O	LCD segment 5, Bit 7 for Port B, or data output (MOSI) of SPI.
SEG6~9/PC0~3	I/O	LCD segment 6~9, or Bit 0~3 for Port C (can be used as key strobe input).
SEG10~30	O	LCD segment 10~30 (SEG10~25 can be used as key strobe output).
COM0~7	O	LCD common 0~7 (COM2~7 can be used as SEG36~31).
V1~2, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：57 Pins

1.5.4 Pad Description for NY8L030A

Pad Name	ATTR.	Description
VDD1~2	P	Positive supply power.
VSS1~2	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX	I/O	Bit 4 for Port A, or input of RFC function.
PA5/RR	I/O	Bit 5 for Port A, or output of RFC function.
PA6/RT0	I/O	Bit 6 for Port A, or output of RFC function.
PA7/RT1	I/O	Bit 7 for Port A, or output of RFC function.
PB0/PWM2	I/O	Bit 0 for Port B, or PWM2 output.
PB1/PWM1	I/O	Bit 1 for Port B, or PWM1 output.
SEG0/PB2/ELP	I/O	LCD segment 0, Bit 2 for Port B, or charging signal of EL driver.
SEG1/PB3/ELC	I/O	LCD segment 1, Bit 3 for Port B, or discharging signal of EL driver.
SEG2/PB4/IR	I/O	LCD segment 2, Bit 4 for Port B, or IR output.
SEG3/PB5/SCK	I/O	LCD segment 3, Bit 5 for Port B, or clock output of SPI.
SEG4/PB66/SDI	I/O	LCD segment 4, Bit 6 for Port B, or data input (MISO) of SPI.
SEG5/PB7/SDO	I/O	LCD segment 5, Bit 7 for Port B, or data output (MOSI) of SPI.
SEG6~9/PC0~3	I/O	LCD segment 6~9, or Bit 0~3 for Port C (can be used as key strobe input).
SEG10~32	O	LCD segment 10~32 (SEG10~25 can be used as key strobe output).
COM0~11	O	LCD common 0~11 (COM5~11 can be used as SEG39~33).
V1~3, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：66 Pins

1.5.5 Pad Description for NY8L040A

Pad Name	ATTR.	Description
VDD1~2	P	Positive supply power.
VSS1~2	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX/AIN3	I/O	Bit 4 for Port A, input of RFC function, or analog input 3 for ADC.
PA5/RR/AIN2	I/O	Bit 5 for Port A, output of RFC function, or analog input 2 for ADC.
PA6/RT0/AIN1	I/O	Bit 6 for Port A, output of RFC function, or analog input 1 for ADC.
PA7/RT1/AIN0	I/O	Bit 7 for Port A, output of RFC function, or analog input 0 for ADC.
PB0/PWM2	I/O	Bit 0 for Port B, PWM2 output.
PB1/PWM1/DAC	I/O	Bit 1 for Port B, or PWM1 output, or DAC output.
SEG0/PB2/ELP	I/O	LCD segment 0, Bit 2 for Port B, or charging signal of EL driver.
SEG1/PB3/ELC	I/O	LCD segment 1, Bit 3 for Port B, or discharging signal of EL driver.
SEG2/PB4/IR	I/O	LCD segment 2, Bit 4 for Port B, or IR output.
SEG3/PB5/SCK	I/O	LCD segment 3, Bit 5 for Port B, or clock output of SPI.
SEG4/PB6/SDI	I/O	LCD segment 4, Bit 6 for Port B, or data input (MISO) of SPI.
SEG5/PB7/SDO	I/O	LCD segment 5, Bit 7 for Port B, or data output (MOSI) of SPI.
SEG6~13/PC0~7	I/O	LCD segment 6~13, or Bit 0~7 for Port C (can be used as key strobe input).
SEG14~42	O	LCD segment 14~42 (SEG14~29 can be used as key strobe output).
COM0~11	O	LCD common 0~11 (COM4~11 can be used as SEG50~43).
V1~3, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：76 Pins

1.5.6 Pad Description for NY8L050A

Pad Name	ATTR.	Description
VDD1~3	P	Positive supply power.
VSS1~3	P	Negative supply power.
PA0/Xin	I/O	Bit 0 for Port A, or input of XTAL32K.
PA1/Xout	I/O	Bit 1 for Port A, or output of XTAL32K.
PA2/INT	I/O	Bit 2 for Port A, or external interrupt input.
PA3/RST	I/O	Bit 3 for Port A, or external reset input.
PA4/CX/AIN3	I/O	Bit 4 for Port A, input of RFC function, or analog input 3 for ADC.
PA5/RR/AIN2	I/O	Bit 5 for Port A, output of RFC function, or analog input 2 for ADC.
PA6/RT0/AIN1	I/O	Bit 6 for Port A, output of RFC function, or analog input 1 for ADC.
PA7/RT1/AIN0	I/O	Bit 7 for Port A, output of RFC function, or analog input 0 for ADC.
PB0/PWM2	I/O	Bit 0 for Port B, PWM2 output.
PB1/PWM1/DAC	I/O	Bit 1 for Port B, or PWM1 output, or DAC output.
SEG0/PB2/ELP	I/O	LCD segment 0, Bit 2 for Port B, or charging signal of EL driver.
SEG1/PB3/ELC	I/O	LCD segment 1, Bit 3 for Port B, or discharging signal of EL driver.
SEG2/PB4/IR	I/O	LCD segment 2, Bit 4 for Port B, or IR output.
SEG3/PB5/SCK	I/O	LCD segment 3, Bit 5 for Port B, or clock output of SPI.
SEG4/PB6/SDI	I/O	LCD segment 4, Bit 6 for Port B, or data input (MISO) of SPI.
SEG5/PB7/SDO	I/O	LCD segment 5, Bit 7 for Port B, or data output (MOSI) of SPI.
SEG6~13/PC0~7	I/O	LCD segment 6~13, or Bit 0~7 for Port C.
SEG14~21/PD0~7	I/O	LCD segment 14~21, or Bit 0~7 for Port D (can be used as key strobe input).
SEG22~54	O	LCD segment 22~54 (SEG22~37 can be used as key strobe output).
COM0~11	O	LCD common 0~11 (COM3~11 can be used as SEG63~55).
V1~3, VLCD	P	LCD supply power.
CUP1~2	I/O	Auxiliary capacitor pins for voltage pumping.

Total：90 Pins

1.6 Electrical Characteristics

The following lists the electrical characteristics. All the product’s properties must refer to each part’s data sheet.

1.6.1 Absolute Maximum Rating

Symbol	Parameter	Rated Value	Unit
$V_{DD} – V_{SS}$	Supply voltage (NY8L005A~NY8L020A)	-0.5 ~ +4.0	V
	Supply voltage (NY8L030A~NY8L050A)	-0.5 ~ +6.0	V
V_{IN}	Input voltage	VSS-0.3V ~ VDD+0.3	V
T_{OP}	Operating Temperature	0 ~ +70	°C
T_{ST}	Storage Temperature	-25 ~ +85	°C

1.6.2 DC Characteristics (Only for NY8L005A~ NY8L020A)

Symbol	Parameter		VDD	Min.	Typ.	Max.	Unit	Test Condition
V_{DD}	Operating voltage			1.1	1.5	3.6	V	F_{CPU} = 500KHz
				2.0	3	3.6		F_{CPU} = 4MHz
I_{HALT}	Supply Current	Halt mode	1.5		0.1	0.5	uA	Sleep, no load
			3		0.1	0.5		
I_{SB1}		Standby mode1	1.5		1.3		uA	CPU off, IOSC32KHz on, LCD off, Reg off, no load
			3		2.2			
I_{SB2}		Standby mode2	1.5		1.4		uA	CPU off, IOSC32KHz on, LCD on, Reg off, no load
			3		2.3			
I_{SB3}		Standby mode3	1.5		3.8		uA	CPU off, IOSC32KHz on, LCD on, Reg on, no load
			3		5			
I_{SL}		Slow mode	1.5		6		uA	F_{CPU} = IOSC32KHz, no load
			3		20			
I_{OP}		Normal mode	1.5		90		uA	F_{CPU} = 500KHz, no load
			3		300			
			3		1.0		mA	F_{CPU} = 4MHz, no load
I_{IH}	Input current (Internal pull-low)	Weak (1M ohms)	1.5		0.5		uA	V_{IN} = VDD
			3		3			
		Strong (100K ohms)	1.5		5			
			3		30			
I_{OH}	Output high current (PA/B/C, SEG/COM@LED mode)		1.5		-2		mA	V_{OH} = 1.0V
			3		-9			V_{OH} = 2.0V
I_{OL1}	Output low current (PA/B/C)		1.5		4		mA	V_{OL} = 0.5V
			3		18			V_{OL} = 1.0V
I_{OL2}	Output low current (SEG/COM@LED mode)		1.5		2		mA	V_{OL} = 0.5V
			3		9			V_{OL} = 1.0V
ΔF/F	Frequency deviation by voltage drop(500KHz)		1.5		-0.5		%	$\frac{\text{Fosc(1.5V) - Fosc(1.2V)}}{\text{Fosc(1.5v)}}$
	Frequency deviation by voltage drop(4MHz)		3		-0.5			$\frac{\text{Fosc(3.0V) - Fosc(2.4V)}}{\text{Fosc(3.0v)}}$
ΔF/F	Frequency lot deviation (500KHz)		1.5	-1.5		1.5	%	$\frac{\text{Fosc(1.5V) - 500KHz}}{\text{500KHz}}$
	Frequency lot deviation (4MHz)		3	-1.5		1.5	%	$\frac{\text{Fosc(3.0v) - 4MHz}}{\text{4MHz}}$
Fosc	Oscillation Frequency		--	0.48	0.5	0.52	MHz	V_{DD} = 1.1~3.6V
				1.95	2	2.05		
				3.9	4	4.1		

1.6.3 DC Characteristics (Only for NY8L030A~ NY8L050A)

Symbol	Parameter		VDD	Min.	Typ.	Max.	Unit	Test Condition
V_{DD}	Operating voltage			2.0	3.0	5.5	V	F_{CPU} = 4MHz
				3.0	4.5	5.5		F_{CPU} = 8MHz
I_{HALT}	Supply Current	Halt mode	3		0.1	0.5	uA	Sleep, no load
			4.5		0.1	0.5		
I_{SB1}		Standby mode1	3		1.4		uA	CPU off, IOSC32KHz on, LCD off, Reg off, no load
			4.5		2.8			
I_{SB2}		Standby mode2	3		1.5		uA	CPU off, IOSC32KHz on, LCD on, Reg off, no load
			4.5		3			
I_{SB3}		Standby mode3	3		5		uA	CPU off, IOSC32KHz on, LCD on, Reg on, no load
			4.5		8			
I_{SL}		Slow mode	3		15		uA	F_{CPU}=IOSC32KHz, PWM output off, no load
			4.5		30			
I_{OP}		Normal mode	3		0.8		mA	F_{CPU}=4MHz, PWM output off, no load
			4.5		1.8			
I_{IH}	Input current (Internal pull-low)	Weak (1M ohms)	3		3		uA	V_{IN} = VDD
			4.5		9			
		Strong (100K ohms)	3		30			
			4.5		90			
I_{OH}	Output high current (PA/B/C/D, SEG/COM@LED mode)		3		-9		mA	V_{OH} = 2.0V
			4.5		-12			V_{OH} = 3.5V
I_{OL1}	Output low current (PA/B/C/D)		3		18		mA	V_{OL} = 1.0V
			4.5		25			V_{OL} = 1.0V
I_{OL2}	Output low current (SEG/COM@LED mode)		3		9		mA	V_{OL} = 1.0V
			4.5		12.5			V_{OL} = 1.0V
I_{PWM}	PWM output current		3		180		mA	Load = 8 ohms
			4.5		280			
I_{DAC}	DAC output current		3		-1.4		mA	Half scale
			4.5		-1.6			
ΔF/F	Frequency deviation by voltage drop		3		-0.5		%	$\frac{Fosc(3.0V) - Fosc(2.4V)}{Fosc(3.0v)}$
			4.5		-0.5			$\frac{Fosc(4.5V) - Fosc(3.0V)}{Fosc(4.5v)}$
ΔF/F	Frequency lot deviation (4MHz)		3	-1.5		1.5	%	$\frac{Fosc(3.0V) - 4MHz}{4MHz}$
	Frequency lot deviation (8MHz)		4.5	-1.5		1.5	%	$\frac{Fosc(4.5V) - 8MHz}{8MHz}$
Fosc	Oscillation Frequency		--	0.48	0.5	0.52	MHz	V_{DD} = 2.0~5.5V
				1.95	2	2.05		
				3.9	4	4.1		
				7.8	8	8.2		

Chapter 2. Operation Modes

2.1 Clock Source

Because NY8L is a dual-clock IC, there are fast oscillator (F_{FAOS}) and slow oscillator (F_{SLOW}) that can be selected as system oscillation (F_{CPU}). The fast oscillator which could be used as F_{FAOS} is internal high RC oscillator: IOSC8M/IOSC4M/IOSC2M/IOSC500K. The slow oscillators which could be used as F_{SLOW} are internal low RC oscillator (IOSC32K) or external low crystal oscillator (XTAL32K). Users can choose the clock sources by programming its option based on the application.

To utilize the precise timing application, two pins (Xin & Xout) are needed to connect with external crystal module and set the corresponding option for 32KHz crystal. To match the high-speed application, it provides up to 8MHz for F_{CPU} and no additional pins are needed.

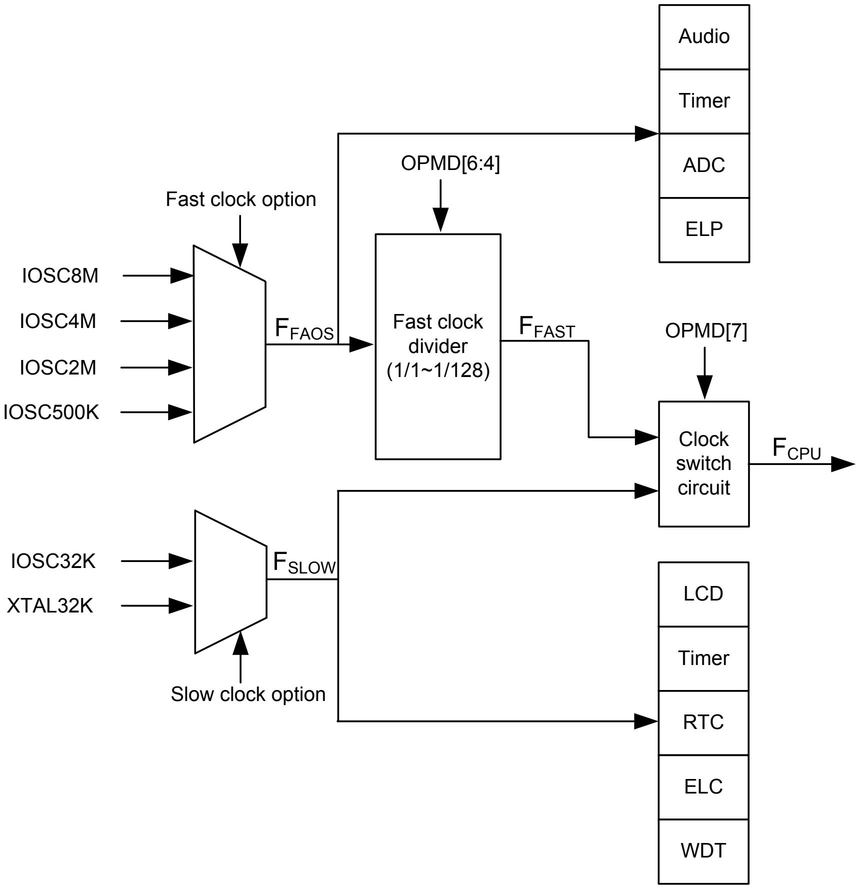


Figure 2-1 Oscillation Configuration

The fast frequency (F_{FAST}) is divided from F_{FAOS} through control register OPMD[6:4]: 1/1~1/128. The F_{SLOW} is the slow frequency and the source of Real-Time-Clock (RTC). The RTC is the clock source of 14-bit divider, ranges from 16KHz to 2Hz, and it generates multiple clocks to apply to the LCD module, LCD power charge pump block, watch dog timer, or EL block, etc. Therefore user is suggested to enable the RTC (OPMD[3]) when CPU is busy. Besides, the RTC counter can be cleared by writing 0 to OPMD[2], and which is always read as high.

Mode	Normal mode	Slow mode	Standby mode	Halt mode
F_{CPU}	ON (F_{FAST})	ON (F_{SLOW})	OFF	OFF
F_{FAOS}	ON	OFF	OFF	OFF
F_{FAST}	$F_{FAOS}/2^N$	OFF	OFF	OFF
F_{SLOW}	ON/OFF	ON	ON	OFF
RTC	ON/OFF	ON	ON	OFF
LCD	ON/OFF	ON/OFF	ON/OFF	OFF
Wake-up Source	--	--	- Key change - Timer2/Timer1/ Timer0 Interrupt (based on F_{SLOW}) - FT/ST Interrupt - External Interrupt - KSB Interrupt - KSB wake-up	- Key change - External Interrupt

NY8L provide four kinds of operating mode to tailor all kinds of application and save power consumptions. These operating modes are Normal mode, Slow mode, Standby mode and Halt mode. Normal mode is designated for high-speed operating mode. Slow mode is designated for low-speed mode in order to save power consumption. At Standby mode, NY8L will stop almost all operations except Timer2 /Timer1 /Timer0 /FT /ST (based on F_{SLOW}) in order to wake up periodically. At Halt mode, NY8L will sleep until key change or external interrupt occurs. User can set the control register OPMD to swap Normal/Slow mode and the control register SLP to enter Standby/Halt mode. The block diagram of four operating modes is described in Figure 2-2.

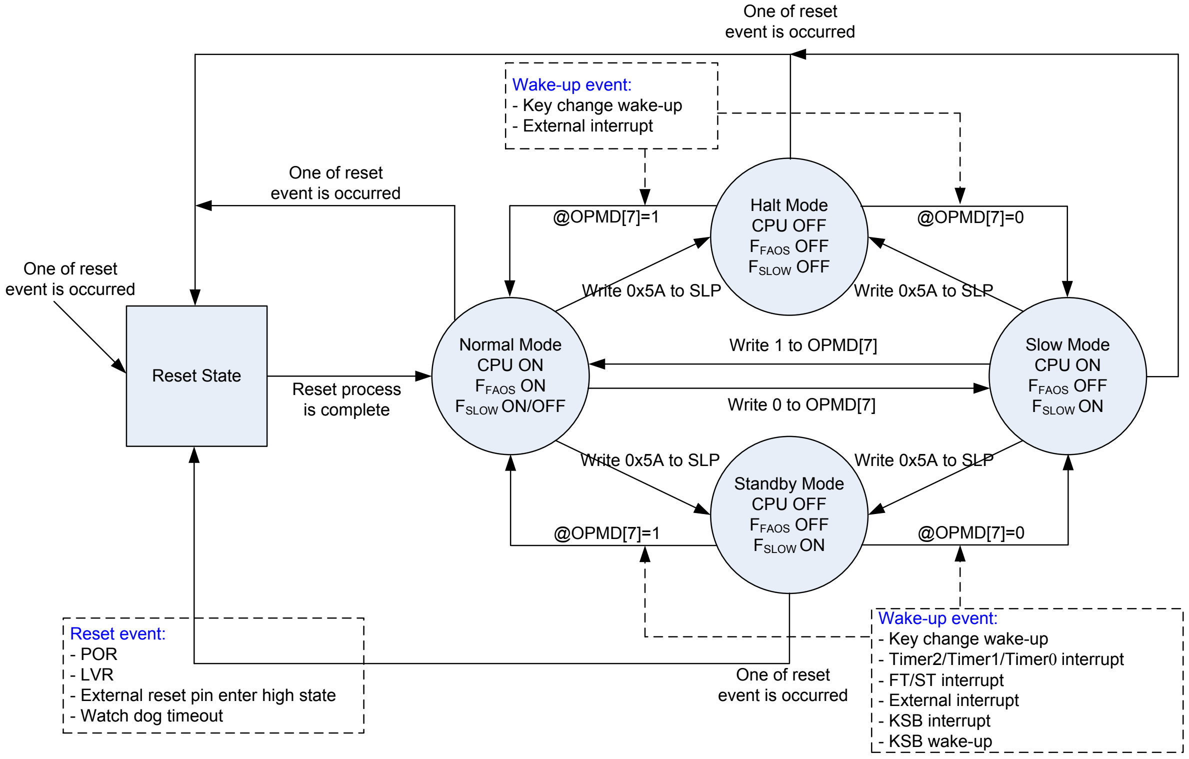


Figure 2-2: Four Operating Modes

2.2 Normal Mode

After any reset event is occurred and reset process is complete, NY8L will enter Normal mode. At Normal mode, F_{FAOS} is selected as system oscillation in order to provide highest performance and its power consumption will be the largest among four operating modes.

- Instruction execution is based on F_{FAST} and all peripheral modules may be active according to corresponding module enable bit.
- F_{FAST} is divided from the F_{FAOS} (1/1 ~ 1/128).
- F_{SLOW} is enabled, or disabled according to application.
- IC can switch to Slow mode by writing 0 to the bit7 of control register OPMD ($17[7]).
- IC can switch to Standby mode or Halt mode by programming control register SLP ($15) with 0x5A.

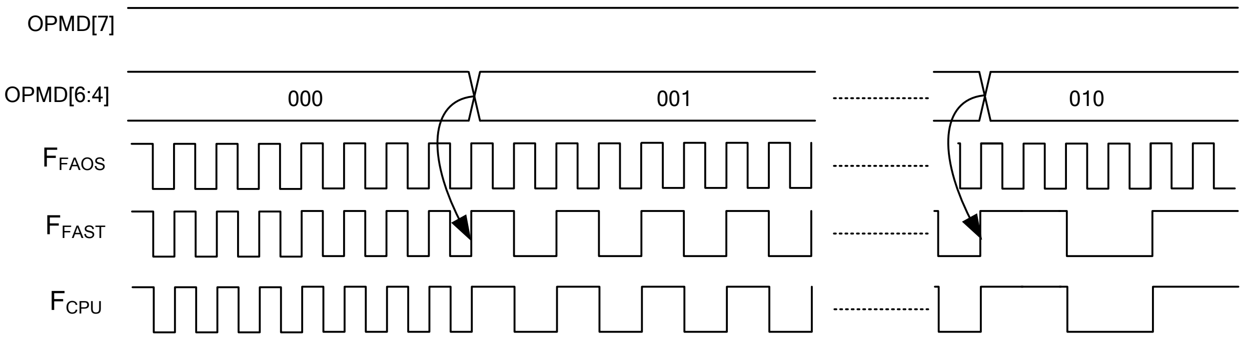


Figure 2-3: The procedure of dividing CPU clock

Users have to set the control register OPMD to the relative setting as the following table.

Addr.	Name	R/W	Bit	Data	Description	Default
$17	OPMD	W	[2]	0	Write 0 to clear RTC counter (read as high)	x
		R/W	[3]	1/0	RTC Enable/Disable	Enable
			[6:4]	000	F_{FAST} = F_{FAOS}/1	F_{FAOS} /1
				001	F_{FAST} = F_{FAOS}/2	
				010	F_{FAST} = F_{FAOS}/4	
				011	F_{FAST} = F_{FAOS}/8	
				100	F_{FAST} = F_{FAOS}/16	
				101	F_{FAST} = F_{FAOS}/32	
				110	F_{FAST} = F_{FAOS}/64	
				111	F_{FAST} = F_{FAOS}/128	
			[7]	1/0	F_{CPU} = F_{FAST}/F_{SLOW} (32KHz)	F_{FAST}

2.3 Slow Mode

NY8L will enter Slow mode by writing 0 to the bit7 of control register OPMD ($17[7]). At Slow mode, F_{SLOW} is selected as system oscillation in order to save power consumption but still keep IC running. However, the F_{FAOS} should be turned off permanently if the mode won't be swapped to Normal mode. When switching to Normal mode, the fast clock source must wait about 512 cycles (~128us@4MHz) for being stable. It is strongly recommended that IC should switch to Slow mode after F_{SLOW} being stable (~120us@IOSC32K or ~30ms@XTAL32K). Once setting control register SLP ($15) with 0x5A, it will turn to Standby mode or Halt mode, and the F_{CPU} will be stopped until the wake-up signal occurs.

- Instruction execution is based on F_{SLOW} and all peripheral modules may be active according to corresponding module enable bit.
- F_{FAOS} can be turned off by writing 0 to OPMD[7].
- IC can switch to Standby mode or Halt mode by programming control register SLP ($15) with 0x5A.
- IC can switch to Normal mode by writing 1 to OPMD[7].

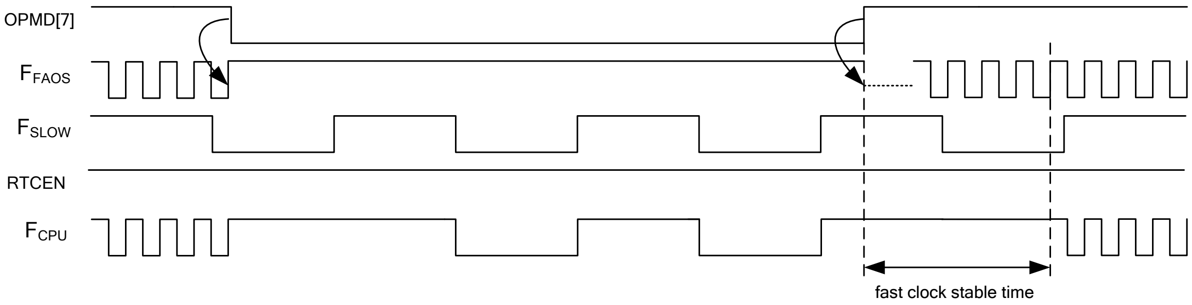


Figure 2-4: The procedure of switching operation modes

2.4 Standby Mode

By setting control register SLP ($15) with 0x5A, the operation mode will turn to Standby mode if RTC is enabled (OPMD[3] = 1), while the previous mode is either Normal mode or Slow mode. At Standby mode, the F_{FAOS} is shut down and the F_{SLOW} is kept to supply the clock for LCD display, etc.

NY8L supports three wake-up methods to leave out of Standby mode, the difference between I/O pads and its data registers (key change), the occurrence of each interrupt, and KSB wake-up. So before entering Standby mode, users have to keep in mind to store the current input port statuses into port registers, and clear key strobe input (KI) data register (PX). If the system is waked up, the succeeding instructions after writing SLP register will be executed after the clock source stable time. The stable time of fast clock source should wait about 512 cycles (~128us@4MHz), and the stable time of IOSC32K is about 4 cycles (~120us@32KHz), the stable time of XTAL32K is about 1024 cycles (~30ms@32KHz).

If the IC is waked up from Standby mode by a reset pin, it goes into reset procedure.

- Instruction execution is stop and some peripheral modules may be active according to corresponding module enable bit.
- F_{FAOS} can be shut down by writing 0x5A to register SLP ($15).
- The F_{SLOW} is still active and running.
- IC can being waked up from Standby mode if any of (a) key change wake-up (refer to 7.1 I/O Ports), (b) Timer2/Timer1/Timer0 (based on F_{SLOW}) interrupt, (c) FT/ST interrupt, (d) external interrupt, (e) KSB interrupt or (f) KSB wake-up (refer to 7.2 Key Strobe) is happened.
- After being waked up from Standby mode, IC will return to Normal mode if OPMD[7] = 1, or Slow mode if OPMD[7] = 0.

The relative control registers are shown as the following tables

Addr.	Name	R/W	Bit	Data	Description	Default
$15	SLP	W	[7:0]		Write 0x5A to sleep	xx

2.5 Halt mode

By setting control register SLP ($15) with 0x5A, the operation mode will turn to Halt mode if RTC is disabled (OPMD[3] = 0), while the previous mode is either Normal mode or Slow mode. Halt mode is also known as Sleep mode. As implied by the name, the IC falls asleep and the system clock is completely turned off, so all the IC functions are halted and it minimizes the power consumption.

At Halt mode, both of the F_{FAOS} and the F_{SLOW} are shut down and waked up by key change or external interrupt. So before entering Halt mode, users have to keep in mind to store the current input port statuses into port registers. For avoiding awaking Halt mode wrongly, key strobe (KI) must be set as output, and the data register (PX) must be cleared to low. If the system is waked up, the succeeding instructions after writing SLP register will be executed after the clock source stable time.

If the IC is waked up from the standby mode by a reset pin, it goes into the reset procedure.

- Instruction execution is stop and all peripheral modules are disabled.
- F_{FAOS} and F_{SLOW} are both disabled automatically.
- IC can being waked up from Halt mode if any of (a) Key change wake-up (refer to 7.1 I/O Ports) or (b) external interrupt is happened.
- After being waked up from Halt mode, IC will return to Normal mode if OPMD[7] = 1, or Slow mode if OPMD[7] = 0.

The relative control registers are shown as the following tables:

Addr.	Name	R/W	Bit	Data	Description	Default
$15	SLP	W	[7:0]		Write 0x5A to sleep	xx

Chapter 3. System Control

3.1 Reset System

For the NY8L IC, the reset procedure needs at least 125ms to deal with initialization process. In addition, 4 conditions will cause the reset procedure to be triggered, described in next sections. The reset initialization procedure is shown in Figure 3-1.

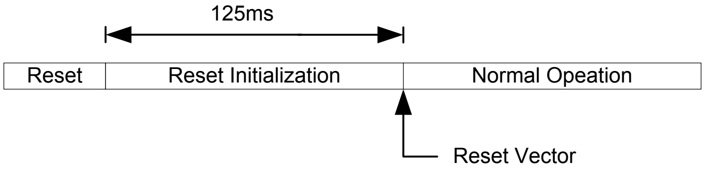


Figure 3-1: The reset initialization procedure

3.1.1 Power-On Reset (POR)

After power-on, the power-on reset initialization will automatically be set out. After the system leaves the reset initialization procedure, it enters the normal operation and the program counter (PC) starts at the reset vector.

3.1.2 Low Voltage Reset (LVR) (by body)

When the system enters the normal operation, the power supply voltage must be kept in an effective working voltage range. When the power supply voltage is lower than the effective operating voltage range, the system can’t work properly. When the detector detects a harmful low voltage supply, it will cause a low voltage reset.

3.1.3 External Reset Pin (by option)

The external reset pin is always pulled-low with strong or weak resistor controlled by option. Generally, when the reset pin rises to high, it generates an external reset.

3.1.4 Watch-Dog Timer Reset (WDTR) (by option)

To recover from program malfunction, the NY8L IC supports an embedded watch-dog timer reset by option. The WDTR function is based on Real-Time-Clock, and always works with the program executing. Users have to clear the WDT ($16) periodically to prevent from timing up with a reset generation. Typically, the minimum time-up period of the WDT is about 1.5s.

Users can write 0xA5 to the control register WDGC to clear WDT, The relative control registers are shown as the following tables.

Addr.	Name	R/W	Bit	Data	Description	Default
$16	WDGC	W	[7:0]		Write 0xA5 to clear watchdog timer	xx

3.2 Low Voltage Detector (LVD) (by body)

To monitor the voltage supply, the NY8L IC also provides low voltage detector (LVD) function. The LVD has 8 levels from 2.2V to 3.6V. If LVD is enabled, user can read back LVD status, which will go high when the power supply is lower than LVD level. The setting of LVD is shown as below.

Addr.	Name	R/W	Bit	Data	Description	Default
$1C	LVD	R/W	[3:0]	0100	LVD level = 2.2V	2.2V
				0101	LVD level = 2.4V	
				0110	LVD level = 2.6V	
				0111	LVD level = 2.8V	
				1x00	LVD level = 3.0V	
				1x01	LVD level = 3.2V	
				1x10	LVD level = 3.4V	
				1x11	LVD level = 3.6V	
		R/W	[4]	1/0	LVD Enable/Disable	Disable
		R	[7]	1/0	LVD status: VDD<LVD level/VDD>LVD level	x

3.3 Interrupts

The interrupt event can be a fixed interval of the Timer2/Timer1/Timer0, the fast real timer (FT), the slow real timer (ST), a random period triggered by the external interrupt pin (INT), the occurrence of analog to digital converter (ADC), or the occurrence of key strobe (KSB). The Timer2/Timer1/Timer0 can also be selected as one of the sample rate for audio playing, and the KSBF arises as a key strobe pulse occurs. There are two real timers (FT & ST) in the NY8L IC, which function as long as it isn't in the halt mode. NY8L provide 8 fixed intervals from the real timer for FT, ranged from 16Hz to 16KHz, and it provide 8 fixed intervals from the real timer for ST, ranged from 256Hz to 2Hz. The interrupt events have to be cleared by users after entering the interrupt routine.

While any of hardware interrupts is occurred, the corresponding bit of interrupt flag register INTF will be set to 1. This bit will not be clear until users write 0 to this bit. Therefore user can obtain information of which event causes hardware interrupt by polling register INTF even if interrupt enable flag register IEF is Disable. The detailed settings of interrupt mode and flag are shown as the following tables.

Addr.	Name	R/W	Bit	Data	Description	Default
$0C	IEF	R/W	[0]	1/0	Timer2 Interrupt Enable/Disable	Disable
			[1]	1/0	Timer1 Interrupt Enable/Disable	Disable
			[2]	1/0	Timer0 Interrupt Enable/Disable (or Timer0 stop@Timer0 stop enable)	Disable
			[3]	1/0	FT Interrupt Enable/Disable	Disable
			[4]	1/0	ST Interrupt Enable/Disable	Disable
			[5]	1/0	EXT Interrupt Enable/Disable	Disable
			[6]	1/0	ADC Interrupt Enable/Disable	Disable
			[7]	1/0	Key Strobe Interrupt Enable/Disable	Disable

Addr.	Name	R/W	Bit	Data	Description	Default
$0D	INTF	R	[0]	1/0	Read Timer2 Interrupt Flag	0
			[1]	1/0	Read Timer1 Interrupt Flag	0
			[2]	1/0	Read Timer0 Interrupt Flag (or Timer0 stop@Timer0 stop enable)	0
			[3]	1/0	Read FT Interrupt Flag	0
			[4]	1/0	Read ST Interrupt Flag	0
			[5]	1/0	Read EXT Interrupt Flag	0
			[6]	1/0	Read ADC Interrupt Flag	0
			[7]	1/0	Read Key Strobe Interrupt Flag	0
		W	[0]	0	Clear Timer2 Interrupt Flag	0
			[1]	0	Clear Timer1 Interrupt Flag	0
			[2]	0	Clear Timer0 Interrupt Flag (or Timer0 stop@Timer0 stop enable)	0
			[3]	0	Clear FT Interrupt Flag	0
			[4]	0	Clear ST Interrupt Flag	0
			[5]	0	Clear EXT Interrupt Flag	0
			[6]	0	Clear ADC Interrupt Flag	0
			[7]	0	Clear Key Strobe Interrupt Flag	0
$0E	NMI	R/W	[0]	1/0	Timer2 Interrupt Flag be NMI/IRQ	IRQ
			[1]	1/0	Timer1 Interrupt Flag be NMI/IRQ	IRQ
			[2]	1/0	Timer0 Interrupt Flag be NMI/IRQ (or Timer0 stop@Timer0 stop enable)	IRQ
			[3]	1/0	FT Interrupt Flag be NMI/IRQ	IRQ
			[4]	1/0	ST Interrupt Flag be NMI/IRQ	IRQ
			[5]	1/0	EXT Interrupt Flag be NMI/IRQ	IRQ
			[6]	1/0	ADC Interrupt Flag be NMI/IRQ	IRQ
			[7]	1/0	Key Strobe Interrupt Flag be NMI/IRQ	IRQ
$0F	RTC	R/W	[2:0]	000	FT Interrupt = RT[10] (16Hz, F_{SLOW}/2048)	RT[4]
				001	FT Interrupt = RT[8] (64Hz, F_{SLOW}/512)	
				010	FT Interrupt = RT[6] (256Hz, F_{SLOW}/128)	
				011	FT Interrupt = RT[4] (1KHz, F_{SLOW}/32)	
				100	FT Interrupt = RT[3] (2KHz, F_{SLOW}/16)	
				101	FT Interrupt = RT[2] (4KHz, F_{SLOW}/8)	
				110	FT Interrupt = RT[1] (8KHz, F_{SLOW}/4)	
				111	FT Interrupt = RT[0] (16KHz, F_{SLOW}/2)	
			[5:3]	000	ST Interrupt = RT[13] (2Hz, F_{SLOW}/16384)	RT[13]
				001	ST Interrupt = RT[12] (4Hz, F_{SLOW}/8192)	
				010	ST Interrupt = RT[11] (8Hz, F_{SLOW}/4096)	
				011	ST Interrupt = RT[10] (16Hz, F_{SLOW}/2048)	
				100	ST Interrupt = RT[9] (32Hz, F_{SLOW}/1024)	
				101	ST Interrupt = RT[8] (64Hz, F_{SLOW}/512)	
				110	ST Interrupt = RT[7] (128Hz, F_{SLOW}/256)	
				111	ST Interrupt = RT[6] (256Hz, F_{SLOW}/128)	
			[6]	1/0	EXT Interrupt takes place at Rising/Falling edge	Rising

Note: It is strongly recommended to set Timer2, Timer1, Timer0, FT, ST, external interrupt, ADC, or key strobe control register before enabling interrupt, otherwise interrupt may be falsely triggered.

For example, if a Timer2 interrupt occurs, the IC pushes the program counter (PC) to stack (STK), and jump to the interrupt vector ($7E0) automatically. In the interrupt sub-routine, user must store the accumulator (ACC), register X/Y (X/Y), and draw the Y, X, and ACC back before Timer2 sub-routine being finished. With the return instruction executes, the interrupt routine is finished. The IC pops STK back to the PC, and back to the original track of the program. The addresses for each interrupt mode are described in below table.

Addr.	Interrupt Vector	IRQ Priority
$7E0	Timer2 Interrupt	1 (highest)
$7E2	Timer1 Interrupt	2
$7E4	Timer0 Interrupt	3
$7E6	FT Interrupt	4
$7E8	ST Interrupt	5
$7EA	EXT Interrupt	6
$7EC	ADC Interrupt	7
$7EE	KSB Interrupt	8 (lowest)

The Timer2 interrupt sub-routine is shown below.

```
V-IRQ-Timer2:
PHA
PHX
PHY
……
PLY
PLX
PLA

RTI
```

The NY8L IC supports two types of interrupt mode, IRQ and NMI (Non-Maskable Interrupt). The IRQ mode is sensed by **level-trigger** event, which means the continued interrupt event will be held until the current is finished. Even if two interrupt events come up simultaneously, the priority of each interrupt decides that the event with higher priority defined by IC itself will be enabled. The NMI mode is sensed by **edge-trigger** event, if an event is coming up with the others at the same time. There is the only one with highest priority defined by software will be enabled and the others may be falsely omitted, **so it is strongly recommended to enable ONLY ONE of NMI ($0E).**

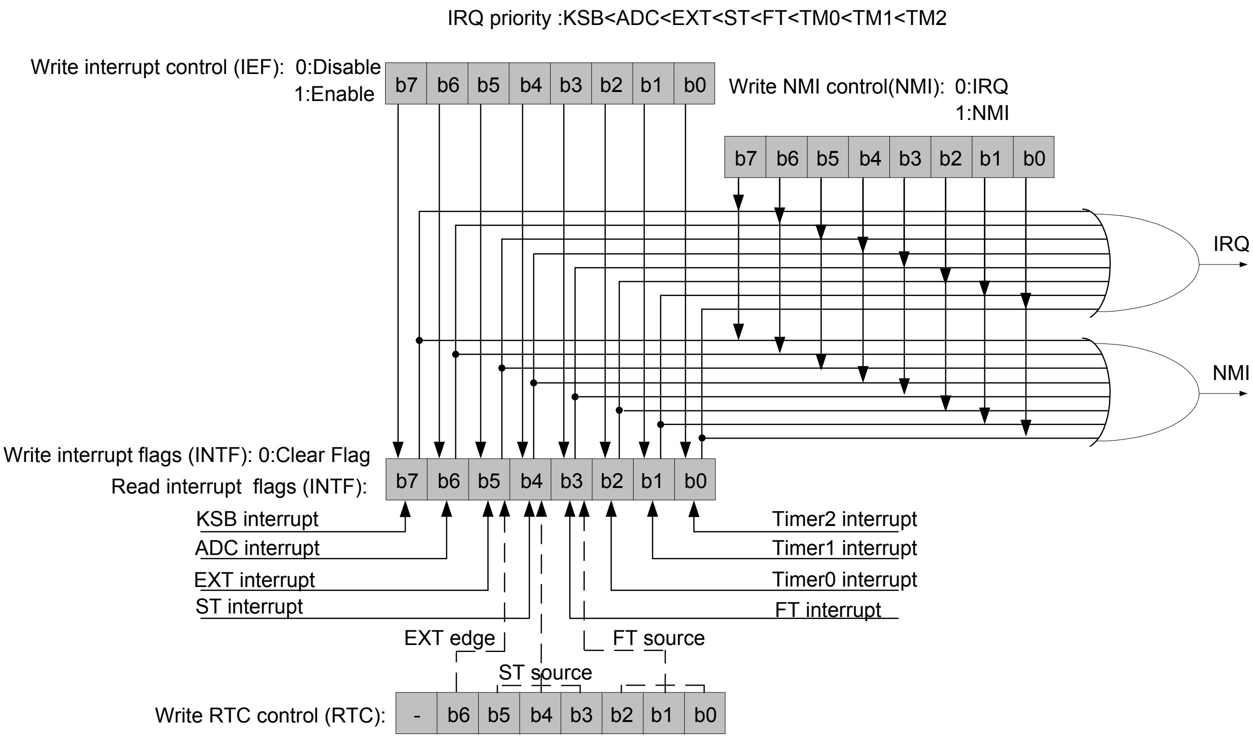


Figure 3-2: The structure of IRQ & NMI

Chapter 4. Address Mapping

In the NY8L IC, SFR contains 64 bytes, RAM contains 64~320 bytes for zero page register (ZP) and stack (STK). Moreover, there are 4K~128K bytes ROM for program data. The following three sections describe the detail about special function register (SFR), RAM and ROM configuration of the NY8L005A~NY8L050A.

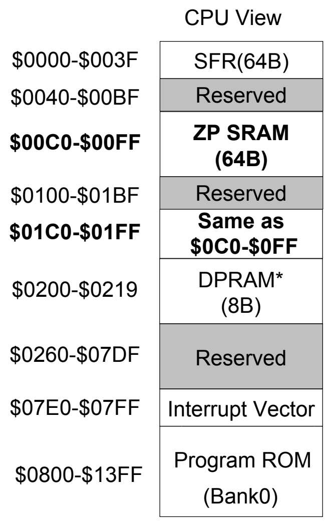


Figure 4-1: The structure of address mapping for NY8L005A

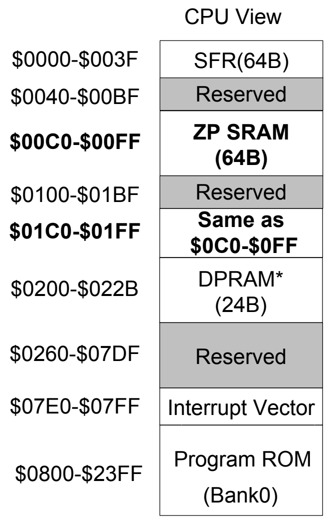
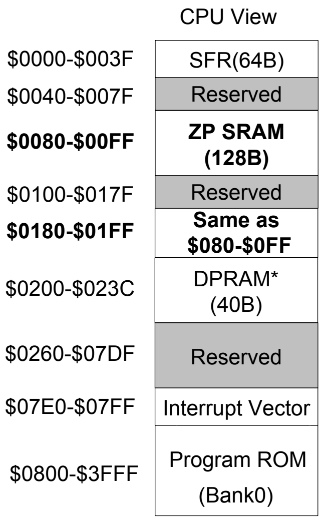
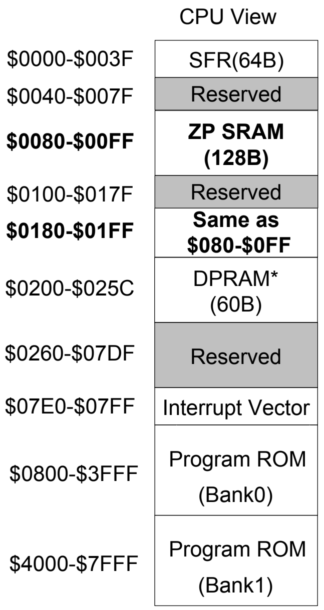


Figure 4-2: The structure of address mapping for NY8L010A



	CPU View
$0000-$003F	SFR(64B)
$0040-$007F	Reserved
$0080-$00FF	**ZP SRAM (128B)**
$0100-$017F	Reserved
$0180-$01FF	**Same as $080-$0FF**
$0200-$023C	DPRAM* (40B)
$0260-$07DF	Reserved
$07E0-$07FF	Interrupt Vector
$0800-$3FFF	Program ROM (Bank0)

Figure 4-3: The structure of address mapping for NY8L020A



	CPU View
$0000-$003F	SFR(64B)
$0040-$007F	Reserved
$0080-$00FF	**ZP SRAM (128B)**
$0100-$017F	Reserved
$0180-$01FF	**Same as $080-$0FF**
$0200-$025C	DPRAM* (60B)
$0260-$07DF	Reserved
$07E0-$07FF	Interrupt Vector
$0800-$3FFF	Program ROM (Bank0)
$4000-$7FFF	Program ROM (Bank1)

Figure 4-4: The structure of address mapping for NY8L030A

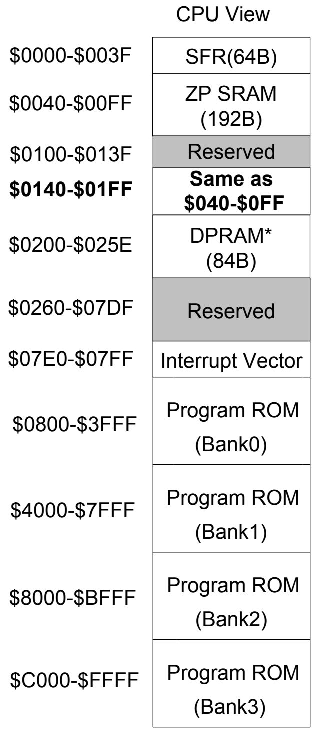


Figure 4-5: The structure of address mapping for NY8L040A

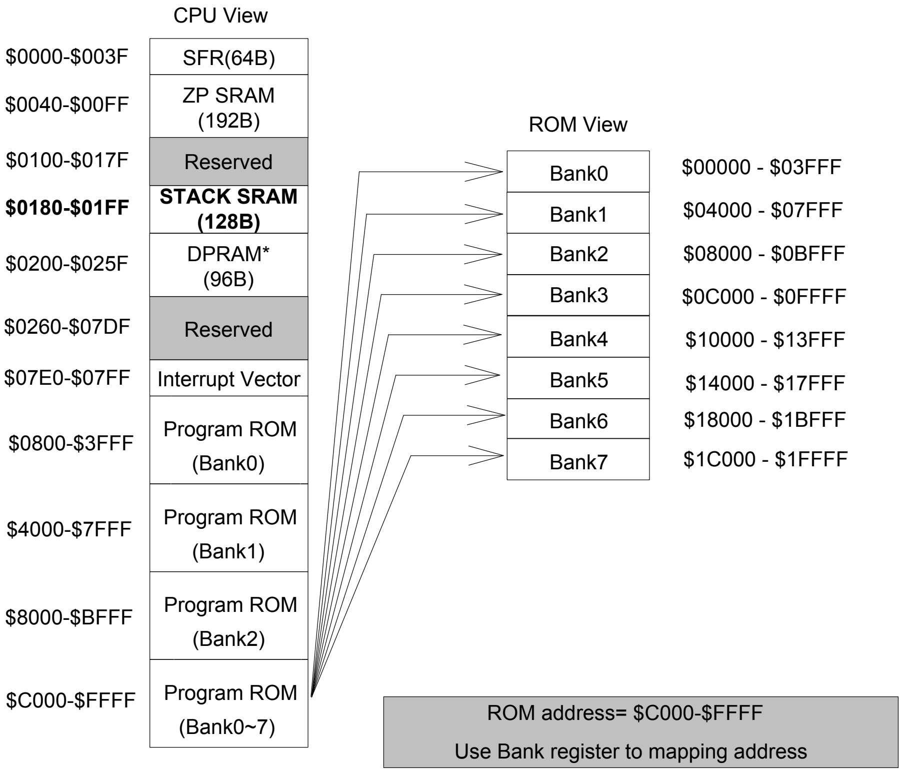


Figure 4-6: The structure of address mapping for NY8L050A

*Please refer to chapter 5.2 for details of LCD RAM address mapping.

4.1 Control Register Description

The special function register (SFR) is assigned to use the dedicated address ranged $0000 to $003B. Users can program these registers to fit their applications. The SFR table is shown below.

Addr.	Name	R/W	Bit	Data	Description	Default
$00	TM0D	R	[7:0]		Read the Timer0 counting data[7:0]	xx
		W	[7:0]		Preload Timer0 data[7:0]	xx
$01	TM0C	R/W	[3:0]	000x	Timer0 clock = CX	BT[0]
				001x	Timer0 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer0 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer0 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer0 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer0 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer0 clock = BT[6] (F_{FAOS}/128)	
				1001	Timer0 clock = BT[5] (F_{FAOS}/64)	
				1010	Timer0 clock = BT[4] (F_{FAOS}/32)	
				1011	Timer0 clock = BT[3] (F_{FAOS}/16)	
				1100	Timer0 clock = BT[2] (F_{FAOS}/8)	
				1101	Timer0 clock = BT[1] (F_{FAOS}/4)	
				1110	Timer0 clock = BT[0] (F_{FAOS}/2)	
				1111	Timer0 clock = F_{FAOS}	
			[4]	1/0	Timer0 Reload/One shot	1
			[5]	1/0	Tone0 Enable/Disable	Enable
			[7:6]	00	Timer0 clock stop mode OFF	OFF
				01	Timer0 clock stopped by Timer2 overflow	
				10	Timer0 clock stopped by a full cycle of CX	
				11	Timer0 clock stopped by a full cycle of Timer2 clock	
$02	TM0EN	R/W	[0]	1/0	Timer0 Enable/Disable	Disable
		R/W	[2:1]	0x	Tone0 4 points patch	00
				10	Tone0 8 points patch	
				11	Tone0 16 points patch	
$03	ENV0	R/W	[7:0]		Envelope of channel 0	00
$04	TM1D	R	[7:0]		Read the Timer1 counting data[7:0]	xx
		W	[7:0]		Preload Timer1 data[7:0]	xx
$05	TM1C	R/W	[3:0]	000x	Timer1 clock = TM0D[7]	BT[0]
				001x	Timer1 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer1 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer1 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer1 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer1 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer1 clock = BT[6] (F_{FAOS}/128)	
				1001	Timer1 clock = BT[5] (F_{FAOS}/64)	
				1010	Timer1 clock = BT[4] (F_{FAOS}/32)	
				1011	Timer1 clock = BT[3] (F_{FAOS}/16)	
				1100	Timer1 clock = BT[2] (F_{FAOS}/8)	
				1101	Timer1 clock = BT[1] (F_{FAOS}/4)	

Addr.	Name	R/W	Bit	Data	Description	Default
				1110	Timer1 clock = BT[0] (F_{FAOS}/2)	
				1111	Timer1 clock = F_{FAOS}	
			[4]	1/0	Timer1 Reload/One shot	1
$06	TM1EN	R/W	[0]	1/0	Timer1 Enable/Disable	Disable
		R/W	[2:1]	0x	Tone1 4 points patch	00
				10	Tone1 8 points patch	
				11	Tone1 16 points patch	
$07	ENV1	R/W	[7:0]		Envelope of channel 1	00
$08	TM2D	R	[7:0]		Read the Timer2 counting data[7:0]	xx
		W	[7:0]		Preload Timer2 data[7:0]	xx
$09	TM2C	R/W	[3:0]	000x	Timer2 clock = TM1D[7]	BT[2]
				001x	Timer2 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer2 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer2 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer2 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer2 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer2 clock = BT[8] (F_{FAOS}/512)	
				1001	Timer2 clock = BT[7] (F_{FAOS}/256)	
				1010	Timer2 clock = BT[6] (F_{FAOS}/128)	
				1011	Timer2 clock = BT[5] (F_{FAOS}/64)	
				1100	Timer2 clock = BT[4] (F_{FAOS}/32)	
				1101	Timer2 clock = BT[3] (F_{FAOS}/16)	
				1110	Timer2 clock = BT[2] (F_{FAOS}/8)	
				1111	Timer2 clock = BT[1] (F_{FAOS}/4)	
			[4]	1/0	Timer2 Reload/One shot	1
$0A	TM2EN	R/W	[0]	1/0	Timer2 Enable/Disable	Disable
$0C	IEF	R/W	[0]	1/0	Timer2 Interrupt Enable/Disable	Disable
			[1]	1/0	Timer1 Interrupt Enable/Disable	Disable
			[2]	1/0	Timer0 Interrupt Enable/Disable (or Timer0 stop@Timer0 stop enable)	Disable
			[3]	1/0	FT Interrupt Enable/Disable	Disable
			[4]	1/0	ST Interrupt Enable/Disable	Disable
			[5]	1/0	EXT Interrupt Enable/Disable	Disable
			[6]	1/0	ADC Interrupt Enable/Disable	Disable
			[7]	1/0	Key Strobe Interrupt Enable/Disable	Disable
$0D	INTF	R	[0]	1/0	Read Timer2 Interrupt Flag	0
			[1]	1/0	Read Timer1 Interrupt Flag	0
			[2]	1/0	Read Timer0 Interrupt Flag (or Timer0 stop@Timer0 stop enable)	0
			[3]	1/0	Read FT Interrupt Flag	0
			[4]	1/0	Read ST Interrupt Flag	0
			[5]	1/0	Read EXT Interrupt Flag	0
			[6]	1/0	Read ADC Interrupt Flag	0
			[7]	1/0	Read Key Strobe Interrupt Flag	0
		W	[0]	0	Clear Timer2 Interrupt Flag	0
			[1]	0	Clear Timer1 Interrupt Flag	0

Addr.	Name	R/W	Bit	Data	Description	Default
			[2]	0	Clear Timer0 Interrupt Flag (or Timer0 stop@Timer0 stop enable)	0
			[3]	0	Clear FT Interrupt Flag	0
			[4]	0	Clear ST Interrupt Flag	0
			[5]	0	Clear EXT Interrupt Flag	0
			[6]	0	Clear ADC Interrupt Flag	0
			[7]	0	Clear Key Strobe Interrupt Flag	0
$0E	NMI	R/W	[0]	1/0	Timer2 Interrupt Flag be NMI/IRQ	IRQ
			[1]	1/0	Timer1 Interrupt Flag be NMI/IRQ	IRQ
			[2]	1/0	Timer0 Interrupt Flag be NMI/IRQ (or Timer0 stop@Timer0 stop enable)	IRQ
			[3]	1/0	FT Interrupt Flag be NMI/IRQ	IRQ
			[4]	1/0	ST Interrupt Flag be NMI/IRQ	IRQ
			[5]	1/0	EXT Interrupt Flag be NMI/IRQ	IRQ
			[6]	1/0	ADC Interrupt Flag be NMI/IRQ	IRQ
			[7]	1/0	Key Strobe Interrupt Flag be NMI/IRQ	IRQ
$0F	RTC	R/W	[2:0]	000	FT Interrupt = RT[10] (16Hz, F_{SLOW}/2048)	RT[4]
				001	FT Interrupt = RT[8] (64Hz, F_{SLOW}/512)	
				010	FT Interrupt = RT[6] (256Hz, F_{SLOW}/128)	
				011	FT Interrupt = RT[4] (1KHz, F_{SLOW}/32)	
				100	FT Interrupt = RT[3] (2KHz, F_{SLOW}/16)	
				101	FT Interrupt = RT[2] (4KHz, F_{SLOW}/8)	
				110	FT Interrupt = RT[1] (8KHz, F_{SLOW}/4)	
				111	FT Interrupt = RT[0] (16KHz, F_{SLOW}/2)	
			[5:3]	000	ST Interrupt = RT[13] (2Hz, F_{SLOW}/16384)	RT[13]
				001	ST Interrupt = RT[12] (4Hz, F_{SLOW}/8192)	
				010	ST Interrupt = RT[11] (8Hz, F_{SLOW}/4096)	
				011	ST Interrupt = RT[10] (16Hz, F_{SLOW}/2048)	
				100	ST Interrupt = RT[9] (32Hz, F_{SLOW}/1024)	
				101	ST Interrupt = RT[8] (64Hz, F_{SLOW}/512)	
				110	ST Interrupt = RT[7] (128Hz, F_{SLOW}/256)	
				111	ST Interrupt = RT[6] (256Hz, F_{SLOW}/128)	
			[6]	1/0	EXT Interrupt takes place at Rising/Falling edge	Rising
$10	ADCMD	R/W	[0]	1/0	ADC Enable/Disable	Disable
			[2:1]	00	ADC input from channel 0 (AIN0)	00
				01	ADC input from channel 1 (AIN1)	
				10	ADC input from channel 2 (AIN2)	
				11	ADC input from channel 3 (AIN3)	
			[4:3]	00	ADCLK = F_{FAOS}/1	00
				01	ADCLK = F_{FAOS}/2	
				10	ADCLK = F_{FAOS}/4	
				11	ADCLK = F_{FAOS}/8	
			[6:5]	00	SHCLK = ADCLK*1	00
				01	SHCLK = ADCLK*2	
				10	SHCLK = ADCLK*4	
				11	SHCLK = ADCLK*8	

Addr.	Name	R/W	Bit	Data	Description	Default
$11	ADCC	R	[0]	1/0	ADC conversion status: ready/NOT ready	1
		R/W	[7]	1/0	ADC conversion start/reset	0
$12	ADCD	R	[7:0]		Voltage ADC data output buffer	xx
$14	KSB	R/W	[3:0]		Key Strobe output segment select (SEGi~SEG(i+15)) (i = 0~48) (by body)	0000
			[4]	1/0	All/One segments selected as key strobe output	One
			[5]	0	Key Strobe Interrupt = key strobe occurs	0
				1	Key Strobe Interrupt = each key strobe scanning cycle	
			[7:6]	00	Key Strobe scan rate = RT[6] (256Hz, F_{SLOW}/128)	RT[6]
				01	Key Strobe scan rate = RT[5] (512Hz, F_{SLOW}/64)	
				10	Key Strobe scan rate = RT[4] (1KHz, F_{SLOW}/32)	
				11	Key Strobe scan rate = RT[3] (2KHz, F_{SLOW}/16)	
$15	SLP	W	[7:0]		Write 0x5A to sleep	xx
$16	WDGC	W	[7:0]		Write 0xA5 to clear watchdog timer	xx
$17	OPMD	W	[2]	0	Write 0 to clear RTC counter (read as high)	x
		R/W	[3]	1/0	RTC Enable/Disable	Enable
			[6:4]	000	F_{FAST} = F_{FAOS}/1	F_{FAOS} /1
				001	F_{FAST} = F_{FAOS}/2	
				010	F_{FAST} = F_{FAOS}/4	
				011	F_{FAST} = F_{FAOS}/8	
				100	F_{FAST} = F_{FAOS}/16	
				101	F_{FAST} = F_{FAOS}/32	
				110	F_{FAST} = F_{FAOS}/64	
				111	F_{FAST} = F_{FAOS}/128	
			[7]	1/0	F_{CPU} = F_{FAST}/F_{SLOW} (32KHz)	F_{FAST}
$18	ELFQ	R/W	[2:0]	000	ELP frequency = BT[9] (F_{FAOS}/1024)	BT[7]
				001	ELP frequency = BT[8] (F_{FAOS}/512)	
				010	ELP frequency = BT[7] (F_{FAOS}/256)	
				011	ELP frequency = BT[6] (F_{FAOS}/128)	
				100	ELP frequency = BT[5] (F_{FAOS}/64)	
				101	ELP frequency = BT[4] (F_{FAOS}/32)	
				110	ELP frequency = BT[3] (F_{FAOS}/16)	
				111	ELP frequency = BT[2] (F_{FAOS}/8)	
			[4:3]	00	ELC frequency = RT[7] (128Hz, F_{SLOW}/256)	RT[5]
				01	ELC frequency = RT[6] (256Hz, F_{SLOW}/128)	
				10	ELC frequency = RT[5] (512Hz, F_{SLOW}/64)	
				11	ELC frequency = RT[4] (1KHz, F_{SLOW}/32)	
$19	ELC	R/W	[2:0]	000	ELP duty = 1/8	7/8
				001	ELP duty = 2/8	
				010	ELP duty = 3/8	
				011	ELP duty = 4/8	
				100	ELP duty = 5/8	
				101	ELP duty = 6/8	
				110	ELP duty = 7/8	
				111	ELP always high	

Addr.	Name	R/W	Bit	Data	Description	Default
			[5:3]	000	ELC duty = 1/8	1/8
				001	ELC duty = 2/8	
				010	ELC duty = 3/8	
				011	ELC duty = 4/8	
				100	ELC duty = 5/8	
				101	ELC duty = 6/8	
				110	ELC duty = 7/8	
				111	ELC always high	
			[7]	1/0	EL Enable/Disable	Disable
$1A	LCDPC	R/W	[2:0]	000	Charge pump clock = RT[6] (256Hz, F_{SLOW}/128)	32KHz
				001	Charge pump clock = RT[5] (512Hz, F_{SLOW}/64)	
				010	Charge pump clock = RT[4] (1KHz, F_{SLOW}/32)	
				011	Charge pump clock = RT[3] (2KHz, F_{SLOW}/16)	
				100	Charge pump clock = RT[2] (4KHz, F_{SLOW}/8)	
				101	Charge pump clock = RT[1] (8KHz, F_{SLOW}/4)	
				110	Charge pump clock = RT[0] (16KHz, F_{SLOW}/2)	
				111	Charge pump clock = 32KHz (F_{SLOW})	
			[3]	1/0	LCD Power (Charge Pump) Enable/Disable	Disable
			[4]	1/0	Internal Voltage Regulator Enable/Disable	Disable
			[6:5][1]	00	Voltage Regulator (Vreg) = 1V	1.50V
				01	Voltage Regulator (Vreg) = 1.17V	
				10	Voltage Regulator (Vreg) = 1.33V	
				11	Voltage Regulator (Vreg) = 1.50V	
$1B	LCDC[2]	R/W	[2:0]	0x0	LCD clock = RT[8] (64Hz, F_{SLOW}/512)	RT[6]
				0x1	LCD clock = RT[7] (128Hz, F_{SLOW}/256)	
				100	LCD clock = RT[6] (256Hz, F_{SLOW}/128)	
				101	LCD clock = RT[5] (512Hz, F_{SLOW}/64)	
				110	LCD clock = RT[4] (1KHz, F_{SLOW}/32)	
				111	LCD clock = RT[3] (2KHz, F_{SLOW}/16)	
			[4:3]	00	LCD OFF	OFF
				01	LCD ON	
				10	LCD all '0'	
				11	LCD all '1'	
$1C	LVD	R/W	[3:0]	0100	LVD level = 2.2V	2.2V
				0101	LVD level = 2.4V	
				0110	LVD level = 2.6V	
				0111	LVD level = 2.8V	
				1x00	LVD level = 3.0V	
				1x01	LVD level = 3.2V	
				1x10	LVD level = 3.4V	
				1x11	LVD level = 3.6V	
		R/W	[4]	1/0	LVD Enable/Disable	Disable
		R	[7]	1/0	LVD status: VDD<LVD level/VDD>LVD level	x
$1E	SPIMD	R/W	[0]	1/0	SPI shift at Mode3/Mode0	Mode3
		R	[7]	1/0	SPI shift Processing/Done	Done

Addr.	Name	R/W	Bit	Data	Description	Default
$1F	SPID	R	[7:0]		Read the shifted-in data from SDI	xx
		W	[7:0]		Latch the data in shift register and starts to shift	xx
$20	VPRL	R/W	[7:0]		Voice pointer low byte	00
$21	VPRH	R/W	[7:0]		Voice pointer high byte	C0
$22	BANK	R/W	[2:0]		Bank register for accessing voice ROM, total 8 Banks (16K bytes/Bank)	0
$23	RBF2	W	[7:0]		CH2 ROM data buffer	FF
$24	OBF2	R/W	[7:0]		CH2 voice output data buffer	00/FF (by option)
$25	VDH0	R	[3:0]		Always be {0000}	0
		R	[7:4]		Voice ROM data high nibble	F
$26	VDL0	R	[3:0]		Always be {0000}	0
		R	[7:4]		Voice ROM data low nibble	F
$27	VD0H	R	[3:0]		Voice ROM data high nibble	F
		R	[7:4]		Always be {0000}	0
$28	VD0L	R	[3:0]		Voice ROM data low nibble	F
		R	[7:4]		Always be {0000}	0
$29	AUD	R/W	[0]	1/0	Audio Output Enable/Disable	Disable
		R/W	[1]	1/0	Interpolation Enable/Disable	Disable
$2A	VOL[3]	R/W	[3:0]		8-level/16-level of PWM/DAC volume	1111
$2B	MIX	R	[3:0]		MSB 4 bits of the data after Mixer	1000/0000 (by option)
		R/W	[4]	0	CH01 = CH0 + CH1	0
				1	CH01 = CH0 + CH0	
			[6:5]	0x	CHDT = CH01 + (CH2 * 2)	00
				10	CHDT = CH01 + CH01	
				11	CHDT = (CH2 * 4)	
			[7]	1/0	PWM/DAC	1/0 (by option)
$2C	VSIGN	R/W	[4]		CH2 sign bit	1/0 (by option)
$2F	IRC	R/W	[0]	1/0	IR data output register	H/L (by option)
			[1]	1/0	CMOS/Open-Drain of IR output	CMOS
			[2]	1/0	IR Enable/Disable	Disable
		W	[3]	0	Write 0 to initial IR counter (read as high)	x
		R/W	[5:4]	00	RFC Disable	Disable
				01	RFC output the reverse signal of CX from RR	
				10	RFC output the reverse signal of CX from RT0	
				11	RFC output the reverse signal of CX from RT1	
$30	PAIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$31	PBIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$32	PCIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$33	PDIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$34	PA	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$35	PB	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00

Addr.	Name	R/W	Bit	Data	Description	Default
$36	PC	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$37	PD	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$38	PAC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$39	PBC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$3A	PCC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$3B	PDC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00

Note: 1. Voltage regulator are NOT available in NY8L005A~NY8L010A.

2. Voltage regulator level: 1.67V and 1.84V are NOT available in NY8L020A.

3. LCDC[2] ($1B) keep high, and 64Hz & 128Hz are NOT available in NY8L030A~NY8L050A.

4. VOL[3] ($2A) keep low, and ONLY 8-level of PWM/DAC volume are provided in NY8L030A.

4.2 RAM

The static RAM (SRAM) is organized with 64~320 bytes for zero page register (ZP) and stack register (STK). The ZP and STK are created to store data and save the program counter (PC) for returning. Their addressing region are $0040~$00FF and $0180~$01FF, respectively. If the address is not existed, the data will be read as unknown.

For LCD application, DPRAM and ranges from $200 to $25F (refer to chapter 5.2), and the size of DPRAM is 8B~96B. The data for display can be updated at any time, and depicted on the LCD panel immediately.

4.3 ROM

In the NY8L IC, a large program/data/voice single ROM is provided, and the size of ROM is 4KB~128KB. It supports up to 8 banks to be configured as ROM and each bank size is 16KB. Because of system information comprised in the reserved region, users are not allowed to access this part. Otherwise, it would cause a reset procedure or other unpredictable impact.

For accessing voice ROM efficiently, the NY8L IC provide auto-increase function, including the voice pointer register (VPRH/VPRL) and the bank register (BANK). The VPRH includes a set reload value latch and a set 8-bit upward counter. Firstly, user must turn on bank auto-increase option. Then, when the VPRL is increased to 0xFF, the VPRH counter will count up automatically. At last, when the VPRL and VPRH are both increased to 0xFF, the BANK will count up automatically. At the same time, the VPRH will be reloaded as 0xC0.

These control registers shown as below can be used for indirect addressing.

Addr.	Name	R/W	Bit	Data	Description	Default
$20	VPRL	R/W	[7:0]		Voice pointer low byte	00
$21	VPRH	R/W	[7:0]		Voice pointer high byte	C0
$22	BANK	R/W	[2:0]		Bank register for accessing voice ROM, total 8 Banks (16K bytes/Bank)	0

Chapter 5. LCD & LED Control

The NY8L IC provides users to drive LCD panel to match the most common-used panels. With the help of Bipolar Junction Transistor (BJT), the NY8L IC also provides users to drive Light Emitting Diode (LED) through driver mode or sink mode. This chapter is mainly to state the power supply for LCD & LED and how it displays on the panel with the corresponding setting.

5.1 LCD Power Supply

According to LCD display theory, it is requested to use the multiple voltage levels to support LCD panel display, otherwise the display will turn to white or black permanently with single voltage level. The NY8L IC contains 1/2, 1/3, 1/4 bias (by body) settings for the most LCD panels. For instance, 1/3 bias needs 4 voltage levels, VSS, V1 (1/3*VLCD), V2 (2/3*VLCD), VLCD. The following descriptions, charge pump mode is configured for this application. And charge pump mode can be based on the power source (VDD), or internal voltage regulator (Vreg) by option.

5.1.1 Power Pumping Mode

For power pumping mode, it also generates the necessary voltage levels for driving the LCD panel with a different bias such as 1/2, 1/3 and 1/4. Its power sources involve VDD and internal Vreg. Basically, the power source (VDD), provides the voltage to VLCD, V3, V2, or V1, which is chosen by option.

On the contrary, internal voltage regulator only acts as the power source to V1. For internal Vreg (= 1V + 0.167V* LCDPC[7:5]), users can select 4~6 levels from 1V to 1.84V.

In charge pump mode, if the display of LCD panel is correct but weak, the method to develop this situation is to increase the clock frequency for charge pump. The relative setting for charge pump clock is defined as below. Please refer to chapter 9.3 for the LCD bias connection diagram.

Addr.	Name	R/W	Bit	Data	Description	Default
$1A	LCDPC	R/W	[2:0]	000	Charge pump clock = RT[6] (256Hz, F_{SLOW}/128)	32KHz
				001	Charge pump clock = RT[5] (512Hz, F_{SLOW}/64)	
				010	Charge pump clock = RT[4] (1KHz, F_{SLOW}/32)	
				011	Charge pump clock = RT[3] (2KHz, F_{SLOW}/16)	
				100	Charge pump clock = RT[2] (4KHz, F_{SLOW}/8)	
				101	Charge pump clock = RT[1] (8KHz, F_{SLOW}/4)	
				110	Charge pump clock = RT[0] (16KHz, F_{SLOW}/2)	
				111	Charge pump clock = 32KHz (F_{SLOW})	
			[3]	1/0	LCD Power (Charge Pump) Enable/Disable	Enable
			[4]	1/0	Internal Voltage Regulator Enable/Disable	Enable
			[6:5][1]	00	Voltage Regulator (Vreg) = 1V	1.50V
				01	Voltage Regulator (Vreg) = 1.17V	
				10	Voltage Regulator (Vreg) = 1.33V	
				11	Voltage Regulator (Vreg) = 1.50V	

Note: 1. Voltage regulator are NOT available in NY8L005A~NY8L010A.

For the LCD display power being stable, the procedure of LCD turning on and turning off is suggested below.

```
M_LCD_Regulator_ON      ;turn on regulator
M_LCD_Charge_ON         ;turn on charge pump
M_LCD_ON                ;turn on LCD
…
M_LCD_OFF               ;turn off LCD
M_LCD_Charge_OFF        ;turn off charge pump
M_LCD_Regulator_OFF     ;turn off regulator
```

5.2 LCD & LED RAM Alignment

The following table is used to store the data for the LCD or LED display, n = 0~5 for 12 common pins, and the maximum address is $025F for DPRAM. It is strongly recommended to initialize the data of DPRAM. Because of uncertain data stored in undecided address, it may cause the defect of display.

Addr.	Data	Common	Segment
$2n0	D[7:0]	COM[2*n]	SEG[7:0]
$2n1			SEG[15:8]
$2n2			SEG[23:16]
$2n3			SEG[31:24]
$2n4			SEG[39:32]
$2n5			SEG[47:40]
$2n6			SEG[55:48]
$2n7			SEG[63:56]
$2n8	D[7:0]	COM[2*n+1]	SEG[7:0]
$2n9			SEG[15:8]
$2nA			SEG[23:16]
$2nB			SEG[31:24]
$2nC			SEG[39:32]
$2nD			SEG[47:40]
$2nE			SEG[55:48]
$2nF			SEG[63:56]

5.3 LCD Display System

As for a plenty of LCD display usages, the NY8L IC provides three mask options to match the most common-used LCD panels of the market, they are:

LCD Duty: 1/(X) duty. (X = 2 ~12)
LCD Bias: 1/(Y) bias. (Y = 2 ~ 4)
LCD SEG: (Z) segments. (Z = 0 ~ 63)

Because of the variety of panels, each display system should set its own combination from those three options. Here is an example works under 1/4 duty, 1/3 bias and 8 segments, the mask options should set X as 4, Y as 3 and Z as 8 to match the specific LCD panel, shown in Figure 5-1. According to Table 1, if users want to display digits such as “1.2.3.4.”, the relative data has to be written in DPRAM to show out. Those data would be 0x7D, 0xA6, 0xE8 and 0xBE and LCD Panel will display “1.2.3.4.” when LCD is turned on by instruction.

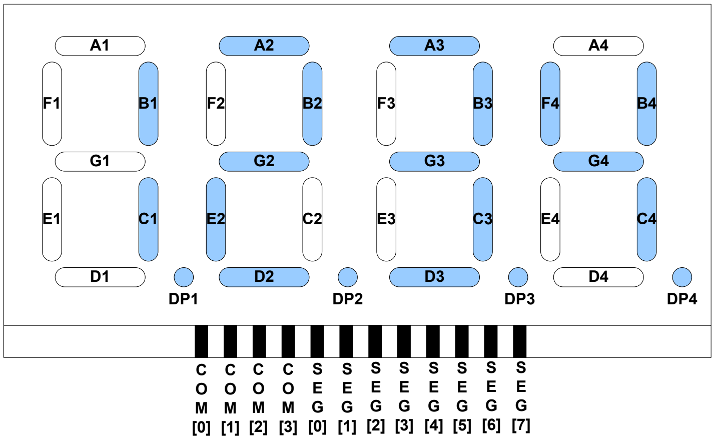


Figure 5-1: 4 digits LCD Panel

Addr.	Item	SEG[7]	SEG[6]	SEG[5]	SEG[4]	SEG[3]	SEG[2]	SEG[1]	SEG[0]
$200	COM[0]	D4	DP4	D3	DP3	D2	DP2	D1	DP1
$208	COM[1]	C4	E4	C3	E3	C2	E2	C1	E1
$210	COM[2]	G4	F4	G3	F3	G2	F2	G1	F1
$218	COM[3]	B4	A4	B3	A3	B2	A2	B1	A1

Table 1: LCD panel mapping

The frame rate for each display system is based on the setting of duty and LCD clock, and these can be selected by setting special function register (SFR). Users can select through setting of SFR, 64Hz, 128Hz, 256Hz, 512Hz, 1KHz or 2KHz for the LCD display (All of the LCD frame rates are sourced from slow clock F_{SLOW}, 32KHz). The following is the formula to calculate frame rate:

Frame Rate = LCD Clock/COM Number

If the pattern on the LCD panel starts to flash, it is suggested to tune to higher rate to be correspondent with the desired display.

Addr.	Name	R/W	Bit	Data	Description	Default
$1B	LCDC[1]	R/W	[2:0]	0x0	LCD clock = RT[8] (64Hz, F_{SLOW}/512)	RT[6]
				0x1	LCD clock = RT[7] (128Hz, F_{SLOW}/256)	
				100	LCD clock = RT[6] (256Hz, F_{SLOW}/128)	
				101	LCD clock = RT[5] (512Hz, F_{SLOW}/64)	
				110	LCD clock = RT[4] (1KHz, F_{SLOW}/32)	
				111	LCD clock = RT[3] (2KHz, F_{SLOW}/16)	
			[4:3]	00	LCD OFF	OFF
				01	LCD ON	
				10	LCD all '0'	
				11	LCD all '1'	

Note: 1. LCDC[2] ($1B) Keep high, and 64Hz & 128Hz are NOT available in NY8L030A~NY8L050A.

For the most common-used LCD panels, the relationship between LCD duty and LCD bias is shown as below table.

LCD bias ≈ $1/(1+\sqrt{duty})$	
LCD duty	**LCD bias select**
2	1/2
3	1/2, 1/3
4	1/2, 1/3
5	1/2, 1/3
6	1/3, 1/4
7	1/3, 1/4
8	1/3, 1/4
9	1/3, 1/4
10	1/3, 1/4
11	1/4
12	1/4

Either common pins or segment pins are operated under alternative voltage level according to the mode it acts. The following lists voltage level of corresponding bias settings and users have to connect with the identical power system.

Bias	Voltage Level
1/2	VSS, V1 (1/2*VLCD), VLCD
1/3	VSS, V1 (1/3*VLCD), V2 (2/3*VLCD), VLCD
1/4	VSS, V1 (1/4*VLCD), V2 (2/4*VLCD), V3 (3/4*VLCD), VLCD

The LCD timing waveforms are shown as the following Figure 5-2 ~ Figure 5-4.

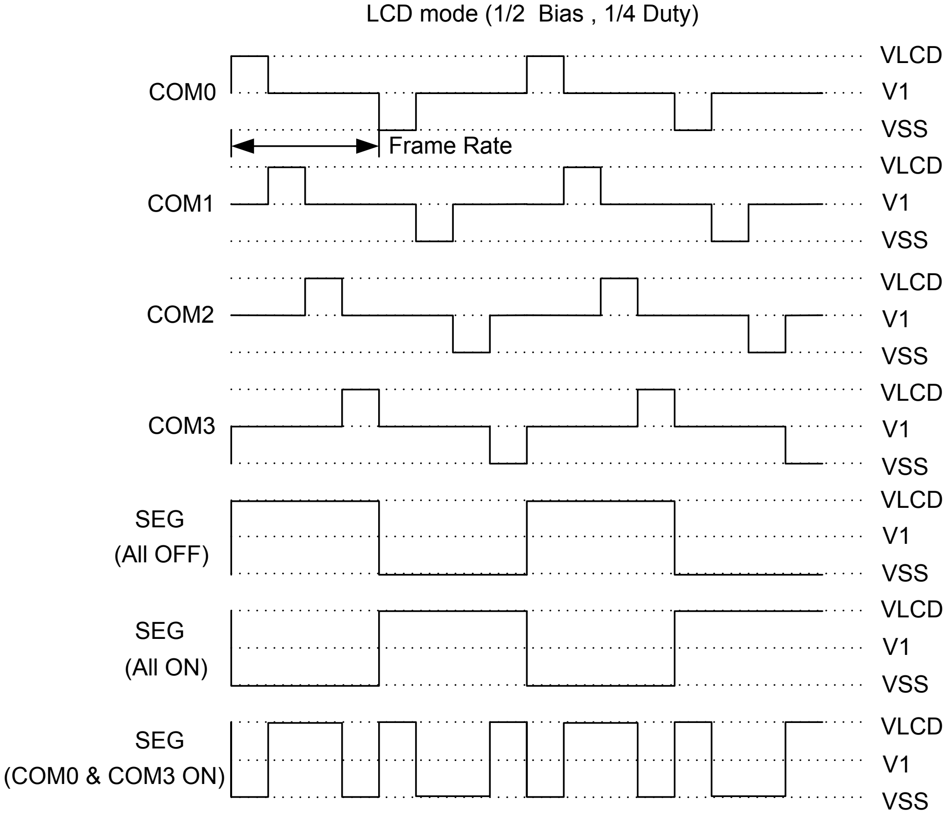


Figure 5-2: LCD timing waveform of 1/2 Bias, 1/4 Duty

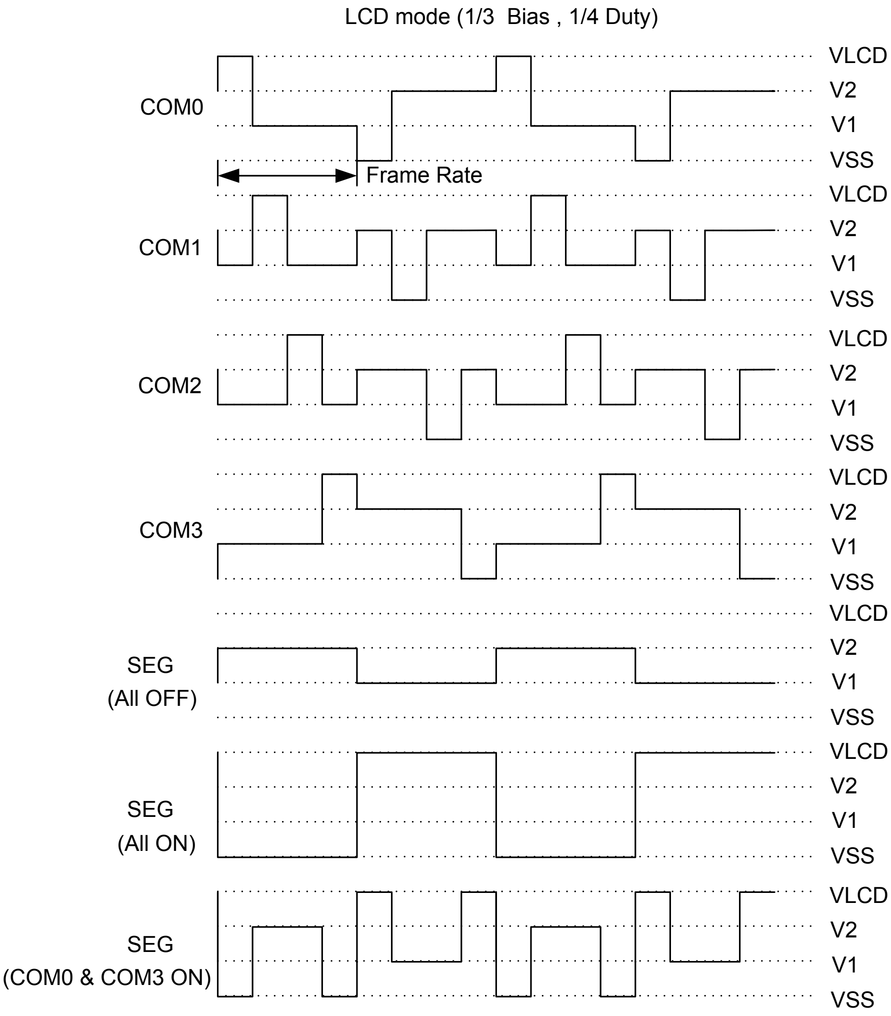


Figure 5-3: LCD timing waveform of 1/3 Bias, 1/4 Duty

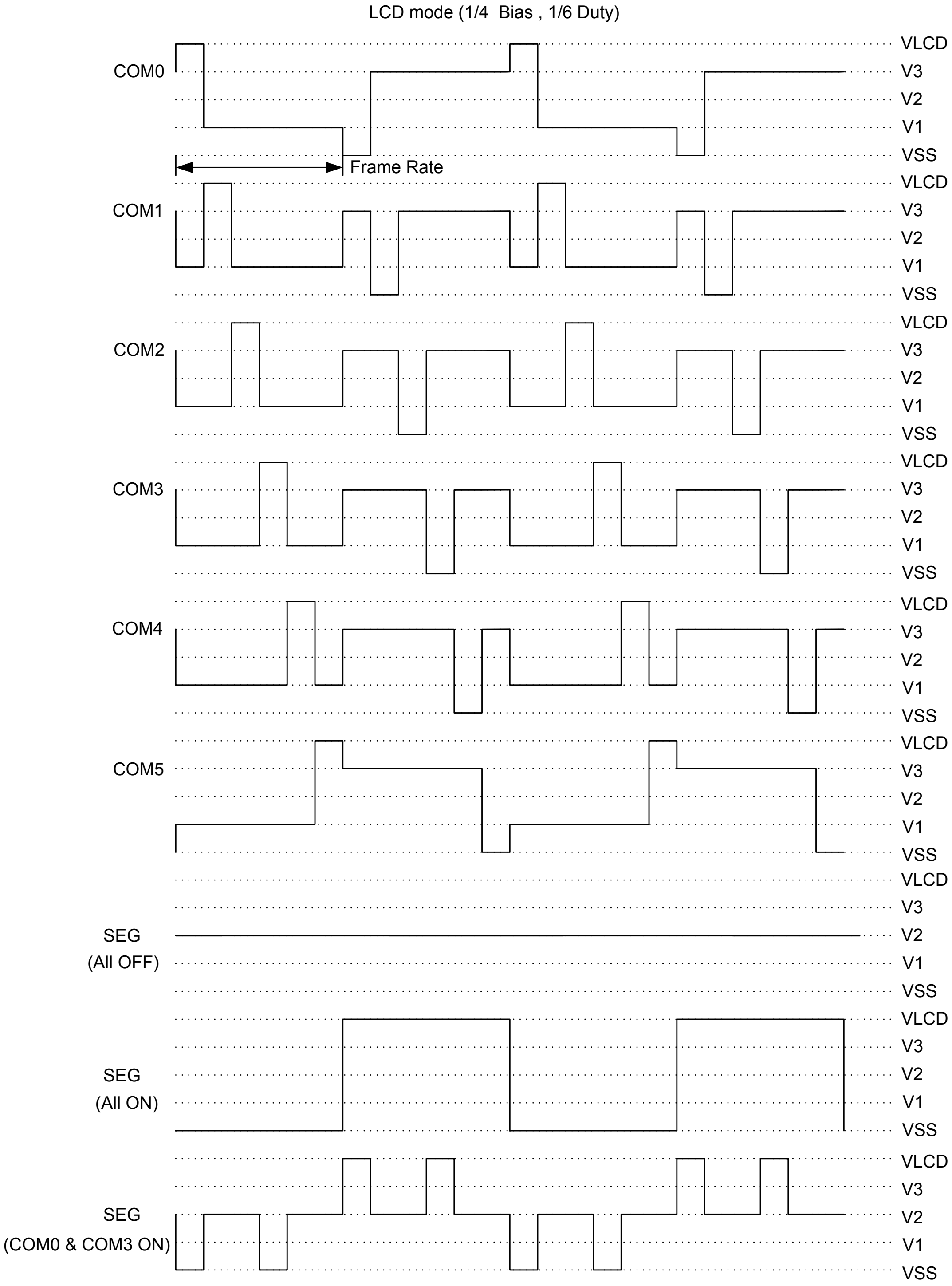


Figure 5-4: LCD timing waveform of 1/4 Bias, 1/6 Duty

5.4 LED Display System

With the help of Bipolar Junction Transistor (BJT), the NY8L IC also provides users to drive Light Emitting Diode (LED) through driver mode or sink mode. LCD and LED functions must all be enabled with the corresponding setting. LED power source is based on VDD without bias circuitry. And users should initialize the data of DPRAM, and choose the frame rate for each display system, which is the same as LCD display by setting control register (LCD).

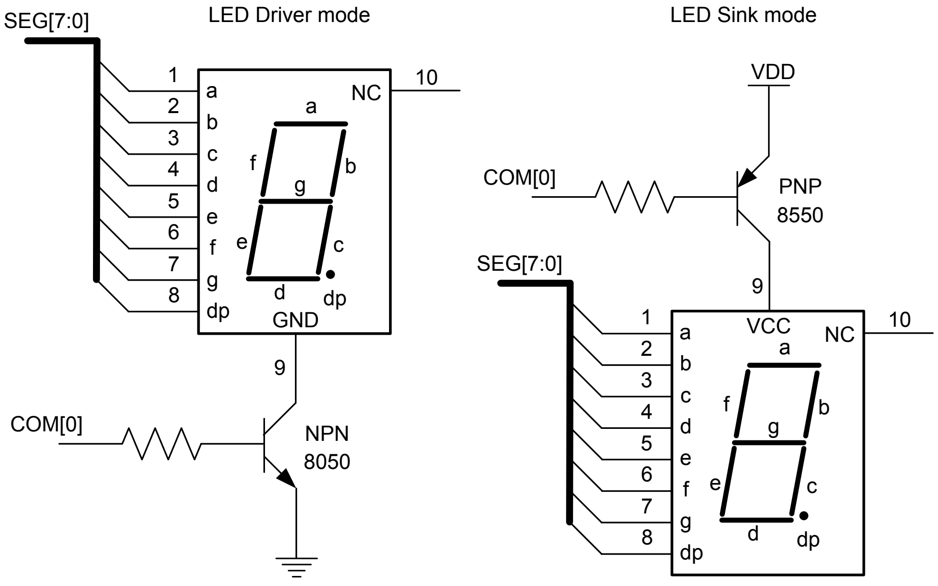


Figure 5-5: LED driving circuitry of drive mode & sink mode

The LED timing waveforms are shown as the following Figure 5-6 & Figure 5-7.

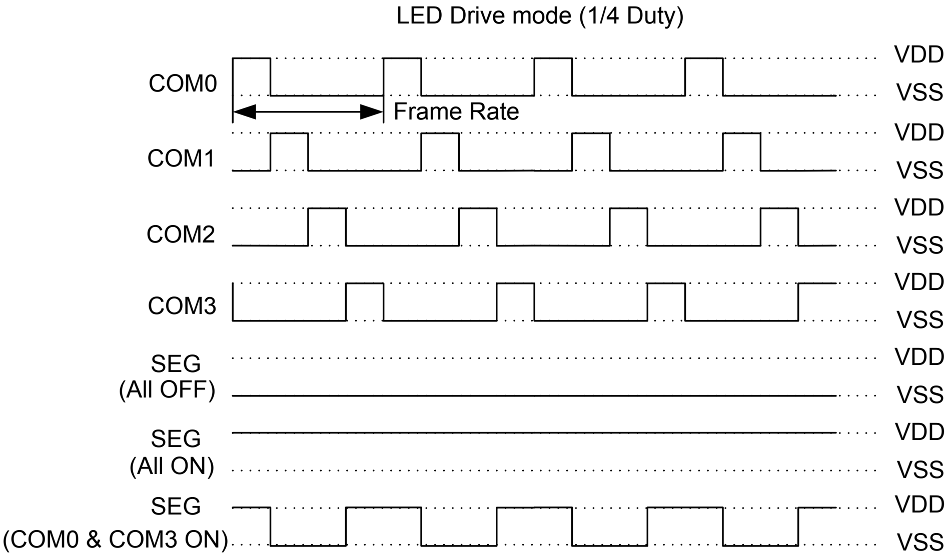


Figure 5-6: LED timing waveform of driver mode

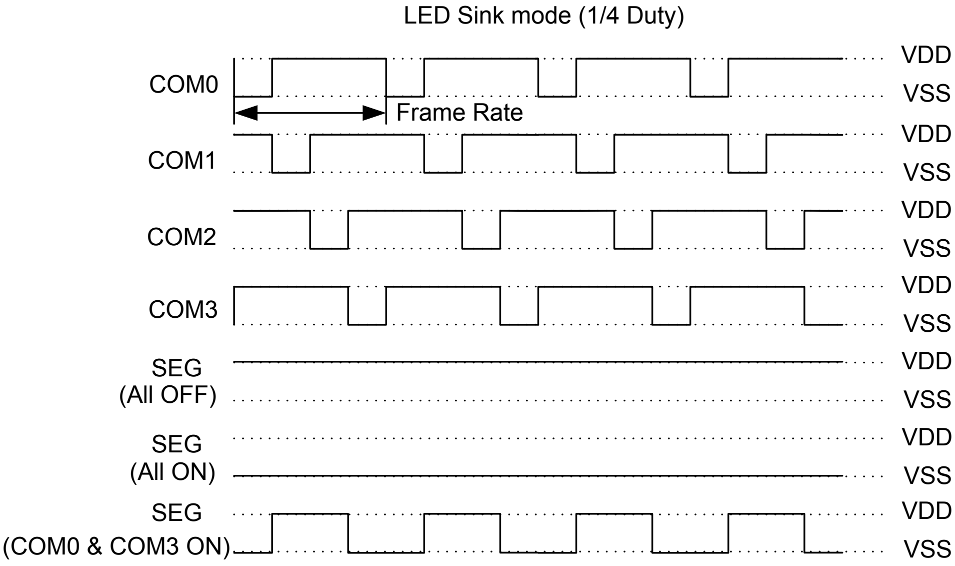


Figure 5-7: LED timing waveform of sink mode

Chapter 6. Audio Control

There are 2-ch tone and 1-ch voice in NY8L030A~050A, channel 0 and channel 1 work as tone mode, channel 2 works as voice mode. And the synthesizer mixes the multi-channel and output the result audio data through 9-bit PWM or 10-bit DAC (by option). And NY8L005A~020A also provides buzzer output, users can select single or dual channel buzzer, which is controlled by MIX[4].

6.1 Audio Control Register

6.1.1 VOL

The control register VOL dominates the hardware volume control of PWM and DAC. The VOL has 8 or 16 steps. 0x0 means the smallest volume and 0xF is the loudest level. Remember the VOL has to be 0xF before ramping up or down of DAC. The setting of VOL is shown as below.

Addr.	Name	R/W	Bit	Data	Description	Default
$2A	VOL[1]	R/W	[3:0]		8-level/16-level of PWM/DAC volume	1111

Note 1. VOL[3]($2A) Keep Low, and 8-level of PWM/DAC volume are provided in NY8L030A.

6.1.2 TONEx (x=0,1)

Channel 0 and channel 1 work as tone mode, tone0 is based on Timer0, and tone1 is based on Timer1 respectively. Each channel can assign one set of programmable patch and envelope. The 8-bit ENVx registers control the envelope of the `x' channel (x=0, 1). Different envelope variations of a tone or midi timbre can display varied expressions.

Each patch can be one of 4-step, 8-step or 16-step programmable square waveform, and meanwhile the clock source of timer must multiply by 4, 8 or 16. When the fast oscillator (F_{FAOS}) is 4MHz, the 4-step patch can support the pitch range up to the 7th octave, 8-step patch can support the pitch range up to the 6th octave, and other 16-step patch can support up to the 5th octave.

Addr.	Name	R/W	Bit	Data	Description	Default
$02	TM0EN	R/W	[0]	1/0	Timer0 Enable/Disable	Disable
		R/W	[2:1]	0x	Tone0 4 points patch	00
				10	Tone0 8 points patch	
				11	Tone0 16 points patch	
$03	ENV0	R/W	[7:0]		Envelope of channel 0	00
$06	TM1EN	R/W	[0]	1/0	Timer1 Enable/Disable	Disable
		R/W	[2:1]	0x	Tone1 4 points patch	00
				10	Tone1 8 points patch	
				11	Tone1 16 points patch	
$07	ENV1	R/W	[7:0]		Envelope of channel 1	00

Tone0 and tone1 patch controlled by options, and there are 4-level altitudes of 100%, 50%, 25% and 0% for each square step. Through level change of different altitude, the timbre will be much of variety, which won't change the 8-bit data of SFR (ENV0, ENV1).

Section Pattern Option	Tone Envelope(x=0, 1)
00	100% * ENVx[7:0]
01	50%* ENVx[7:0]
10	25%* ENVx[7:0]
11	0%* ENVx[7:0]

6.1.3 AUD

The AUD register turns on/off the audio playing procedure. For playing audio, the register has to be set as high. Otherwise, the music won't show out.

Channel 2 works as voice mode and provides interpolation function. When Interpolation function is enabled, the clock source of Timer2 will be speed up two times automatically. In the first Timer2 cycle, voice data equal (previous data + current data)/2, and in the second cycle, voice data equal current data. If Interpolation function is disabled, voice data always equal current data.

Addr.	Name	R/W	Bit	Data	Description	Default
$29	AUD	R/W	[0]	1/0	Audio output Enable/Disable	Disable
		R/W	[1]	1/0	Interpolation Enable/Disable	Disable

As there are different coding types, the NY8L IC provides SWAP function. The coding data of channel 2 will be stored in RBF2 ($23), users can read back voice data (high or low nibble of RBF2) through VDH0, VDL0, VD0H, or VD0L. After voice data being decoded by software, the OBF2 ($24) is used for voice output data buffer, and the VSIGN ($2C) is sign bit. Programmer can write data to OBF2 and VSIGN for generating audio data directly when finish decoding a voice data.

For muting at the beginning of audio output, it is recommended that the initial value of OBF2 should be 0x00 and VSIGN should be 1 if audio option selects PWM; and on the contrary, the initial value of OBF2 should be 0xFF, and VSIGN should be 0 if audio option selects DAC.

Addr.	Name	R/W	Bit	Data	Description	Default
$23	RBF2	W	[7:0]		CH2 ROM data buffer	FF
$24	OBF2	R/W	[7:0]		CH2 voice output data buffer	00/FF (by option)
$25	VDH0	R	[3:0]		Always be {0000}	0
		R	[7:4]		Voice ROM data high nibble	F
$26	VDL0	R	[3:0]		Always be {0000}	0
		R	[7:4]		Voice ROM data low nibble	F
$27	VD0H	R	[3:0]		Voice ROM data high nibble	F
		R	[7:4]		Always be {0000}	0
$28	VD0L	R	[3:0]		Voice ROM data low nibble	F
		R	[7:4]		Always be {0000}	0
$2C	VSIGN	R/W	[4]		CH2 sign bit	1/0 (by option)

6.1.4 TMxD (x=0, 1, 2)

The TMxD registers include a set 8-bit timer reload value latch and a set 8-bit downward counter of the `x' channel. With the data loaded in the latch, the counter counts down until to 0. If the register is set to the reload mode, the counter will be automatically reloaded from the latch, and the reload period is TMxD+1 (TMxD ≠0). When the counter counts to 0, the audio engine plays the audio data. So the TMxD value affects the sample rate of a speech or the pitch of a tone.

TMxD = (F_{TCS} / F_{SR}) – 1

TMxD: Timer value in decimal

F_{TCS}: Frequency of the timer clock source

F_{SR}: Frequency of the sample rate

In theory, there are TMxD+1 (TMxD ≠0) cycles of timer used to control an accurate period of time. ***But actually the maximum deviation is perhaps 1 cycle of timer, because timer enable signal is asynchronous to timer clock source. And it is strongly recommended to speed up timer clock source to decrease the deviation.*** Users can read back the content of counter through register TMxD by instruction.

Addr.	Name	R/W	Bit	Data	Description	Default
$00	TM0D	R	[7:0]		Read the Timer0 counting data[7:0]	xx
		W	[7:0]		Preload Timer0 data[7:0]	xx
$04	TM1D	R	[7:0]		Read the Timer1 counting data[7:0]	xx
		W	[7:0]		Preload Timer1 data[7:0]	xx
$08	TM2D	R	[7:0]		Read the Timer2 counting data[7:0]	xx
		W	[7:0]		Preload Timer2 data[7:0]	xx

6.1.5 TMxC (x=0, 1, 2)

The TMxC registers indicate the TMx clock source of the `x' channel. Different channel mode has different frequency of the TMxC. The value of the TMxC affects the tone. Besides, real timer (RT[n]) is based on F_{SLOW} (32.768KHz), and base timer (BT[n]) is divided from F_{FAOS} (8MHz/4MHz/2MHz/500KHz), which is selected by option. So BT[n] will be slow down as well as the fast system clock. The settings of TMxC are shown as below.

Addr.	Name	R/W	Bit	Data	Description	Default
$01	TM0C	R/W	[3:0]	000x	Timer0 clock = CX	BT[0]
				001x	Timer0 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer0 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer0 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer0 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer0 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer0 clock = BT[6] (F_{FAOS}/128)	
				1001	Timer0 clock = BT[5] (F_{FAOS}/64)	
				1010	Timer0 clock = BT[4] (F_{FAOS}/32)	
				1011	Timer0 clock = BT[3] (F_{FAOS}/16)	
				1100	Timer0 clock = BT[2] (F_{FAOS}/8)	
				1101	Timer0 clock = BT[1] (F_{FAOS}/4)	

Addr.	Name	R/W	Bit	Data	Description	Default
				1110	Timer0 clock = BT[0] (F_{FAOS}/2)	
				1111	Timer0 clock = F_{FAOS}	
			[4]	1/0	Timer0 Reload/One shot	1
			[5]	1/0	Tone0 Enable/Disable	Enable
			[7:6]	00	Timer0 clock stop mode OFF	OFF
				01	Timer0 clock stopped by Timer2 overflow	
				10	Timer0 clock stopped by a full cycle of CX	
				11	Timer0 clock stopped by a full cycle of Timer2 clock	
$05	TM1C	R/W	[3:0]	000x	Timer1 clock = TM0D[7]	BT[0]
				001x	Timer1 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer1 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer1 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer1 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer1 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer1 clock = BT[6] (F_{FAOS}/128)	
				1001	Timer1 clock = BT[5] (F_{FAOS}/64)	
				1010	Timer1 clock = BT[4] (F_{FAOS}/32)	
				1011	Timer1 clock = BT[3] (F_{FAOS}/16)	
				1100	Timer1 clock = BT[2] (F_{FAOS}/8)	
				1101	Timer1 clock = BT[1] (F_{FAOS}/4)	
				1110	Timer1 clock = BT[0] (F_{FAOS}/2)	
				1111	Timer1 clock = F_{FAOS}	
			[4]	1/0	Timer1 Reload/One shot	1
$09	TM2C	R/W	[3:0]	000x	Timer2 clock = TM1D[7]	BT[2]
				001x	Timer2 clock = RT[13] (2Hz, F_{SLOW}/16384)	
				0100	Timer2 clock = RT[11] (8Hz, F_{SLOW}/4096)	
				0101	Timer2 clock = RT[9] (32Hz, F_{SLOW}/1024)	
				0110	Timer2 clock = RT[7] (128Hz, F_{SLOW}/256)	
				0111	Timer2 clock = RT[5] (512Hz, F_{SLOW}/64)	
				1000	Timer2 clock = BT[8] (F_{FAOS}/512)	
				1001	Timer2 clock = BT[7] (F_{FAOS}/256)	
				1010	Timer2 clock = BT[6] (F_{FAOS}/128)	
				1011	Timer2 clock = BT[5] (F_{FAOS}/64)	
				1100	Timer2 clock = BT[4] (F_{FAOS}/32)	
				1101	Timer2 clock = BT[3] (F_{FAOS}/16)	
				1110	Timer2 clock = BT[2] (F_{FAOS}/8)	
				1111	Timer2 clock = BT[1] (F_{FAOS}/4)	
			[4]	1/0	Timer2 Reload/One shot	1

6.1.6 TMxEN(x=0, 1, 2)

The bit0 of TM0EN, TM1EN and TM2EN are mainly to control these timers to turn on/off respectively. As the value is 1, the timer is turned on and the counter starts counting downward.

Addr.	Name	R/W	Bit	Data	Description	Default
$02	TM0EN	R/W	[0]	1/0	Timer0 Enable/Disable	Disable
$06	TM1EN	R/W	[0]	1/0	Timer1 Enable/Disable	Disable
$0A	TM2EN	R/W	[0]	1/0	Timer2 Enable/Disable	Disable

6.1.7 MIX

Users can define the configuration of mixer through control register MIX to output desired audio data type. It supports 3-channel audio sources, and the mixer can mix 2-ch tones & 1-ch speech data to audio output. The setting of control register MIX are shown as below.

Addr.	Name	R/W	Bit	Data	Description	Default
$2B	MIX	R	[3:0]		MSB 4 bits of the data after Mixer	1000/0000 (by option)
		R/W	[4]	0	CH01 = CH0 + CH1	0
				1	CH01 = CH0 + CH0	
			[6:5]	0x	CHDT = CH01 + (CH2 * 2)	00
				10	CHDT = CH01 + CH01	
				11	CHDT = (CH2 * 4)	
			[7]	1/0	PWM/DAC	1/0 (by option)

Reading the 4-bit data of MIX acquires the value of the MSB 4-bit audio data (16 levels). This information helps users to get the amplitude of the playing sound. 0x0 means the smallest and 0xF means the largest level of the output audio data. If 1-ch speech (channel 2) is enabled merely (MIX[6:5]=11). the initial value of MIX[3:0] is 1000 at PWM mode; otherwise the initial value of MIX[3:0] is 0000 at DAC mode. And on the other hand, if tone (channel 0 or/and channel 1) is enabled, the initial value of MIX[3:0] is unstable with the changing tone periodically. The diagram shown as below is the structure of mixer.

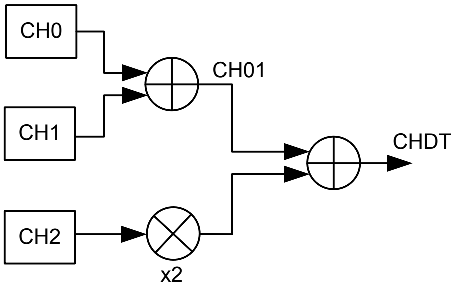
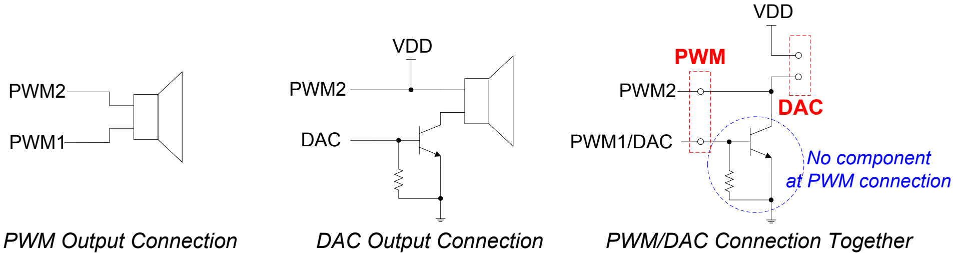


Figure 6-1: The structure of Mixer.

The option of audio output can be set as (a) PWM only, (b) both PWM & DAC, or (c) buzzer only. If PWM only mode is selected, the value of MIX[7] is always high. Besides, if both PWM and DAC mode are selected, the NY8L IC provides a pad detecting mechanism, please refer to Figure 6-2. The pad detecting mechanism detects the PWM2 pad during the reset initialization period, and sets the initial value of the audio output register(MIX[7]) as high if the PWM2 connection is floating (PWM mode), or sets the initial value of the audio output register(MIX[7]) as low if the PWM2 connection is high (DAC mode).



PWM Output Connection *DAC Output Connection* *PWM/DAC Connection Together*

Figure 6-2: The diagram of Audio Module

In conclusion, connect the speaker to PWM1 and PWM2 only if using PWM, and otherwise connect PWM2 to VDD if using DAC. Since the mechanism sets only the initial value of MIX[7], don't change the value of the MIX[7] register if the pad detecting mechanism is adopted.

PWM2 Pad	Audio Output Initialization
Speaker (Floating)	PWM
VDD	DAC

6.2 Audio Control Procedure (Only for NY8L030A~050A)

6.2.1 Tone Control

6.2.1.1 Tone Playing Procedure

The TMxC, TMxD and ENVx (x=0, 1) dominate the adjustment of the tone channel. Setting the TMxC and TMxD to generate a tone with its octave and pitch, and alter the ENVx to make it colorful. Setting the ENVx to 0 will turn off the tone channel. The following table lists the start-up procedure.

Step	Process
#1	Set TMxC to alter the octave
#2	Set TMxD to alter the pitch
#3	Set ENVx to alter the intensity
#4	Set TMxEN to turn on its channel
#5	If ENVx=0, the channel is off.

6.2.1.2 Tone Octave

The octave of a tone is controlled by the TMxC with eight frequency bases. The continued table lists the frequency of note C with 4-step patch for different octave.

Octave	TMxC	Freq. of note C
1	62.5KHz	32.7Hz
2	125KHz	65.4Hz
3	250KHz	130.8Hz
4	500KHz	261.6Hz
5	1MHz	523.3Hz
6	2MHz	1046.5Hz
7	4MHz	2093Hz
8	8MHz	4186Hz

6.2.1.3 Tone Pitch

The note pitch is controlled by the TMxD, and the following table lists the TMxD for different pitches with 4-step patch at TMxC=4MHz. Although it comes with the different octave, the setting for pitch is same for each octave setting as well.

Pitches	Freq. of Tone	TMxD
C7	2093.0Hz	EE
C7#	2217.5Hz	E0
D7	2349.3Hz	D4
D7#	2489.0Hz	C8
E7	2637.0Hz	BD
F7	2793.8Hz	B2
F7#	2960.0Hz	A8
G7	3136.0Hz	9E
G7#	3322.4Hz	95
A7	3520.7Hz	8D
A7#	3729.3Hz	85
B7	3951.1Hz	7E

6.3 Buzzer Control (Only for NY8L005A~020A)

The NY8L005A~020A also provides buzzer output, and users can select single or dual channel buzzer, which is controlled by MIX[4]. When MIX[4]=1, single buzzer toggled by Timer0 overflow, otherwise when MIX[4]=0, dual buzzer toggled by Timer0 or Timer1 overflow, and the buzzer switching rate is equal to F_{SLOW}. The diagram of buzzer module is shown as Figure 6-3.

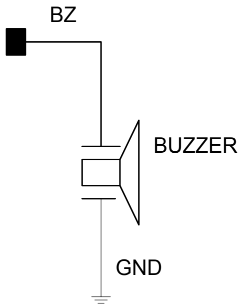


Figure 6-3: The diagram of buzzer module

Chapter 7. I/O Control

7.1 I/O Ports (by body and option)

There are 13~32 I/O ports (by body and option), designated as PAy through PDy, and y=0~7, the structure of I/O ports is shown as Figure 7-1, and users can also enable them by setting options. The bi-direction I/O port can be an input or output by the value of control register PXIO (X = A/B/C/D). If the register value is 1, the port will be set as an input; therefore, the value 0 means output setting. Users can set the control register PXC (X = A/B/C/D) to define the I/O ports to be with/without a pull-low resister if input or configured as COMS/Open-Drain type if output. As for the internal pull-low resistor of input, it can be set as strong or weak pull-low through option. The weak one is about 1MΩ@3V for normal application and the strong one is about 100KΩ@3V.

If Port X (X = A/B/C/D) is key change wake-up source, users must read input pads data and write the value to input port registers before enter Standby/Halt mode. The system will be waked up as long as one of input pads status change.

If Port A is key change wake-up source, the code of entering Standby mode is shown below.

```
LDA  OPMD       ;
ORA  #$08       ;
STA  OPMD       ; Set OPMD[3] as high
LDA  #$FF
STA  PAIO       ; Set PortA as Input
LDA  PA         ; Read PortA data
STA  PA         ; Write to PortA register

LDA  #$5A
STA  SLP        ; Enter Standby mode
```

And the code of entering Halt mode is shown below.

```
LDA  OPMD       ;
AND  #$F7       ;
STA  OPMD       ; Clear OPMD[3] to low
LDA  #$FF
STA  PAIO       ; Set PortA as Input
LDA  PA         ; Read PortA data
STA  PA         ; Write to PortA register

LDA  #$5A
STA  SLP        ; Enter Halt mode
```

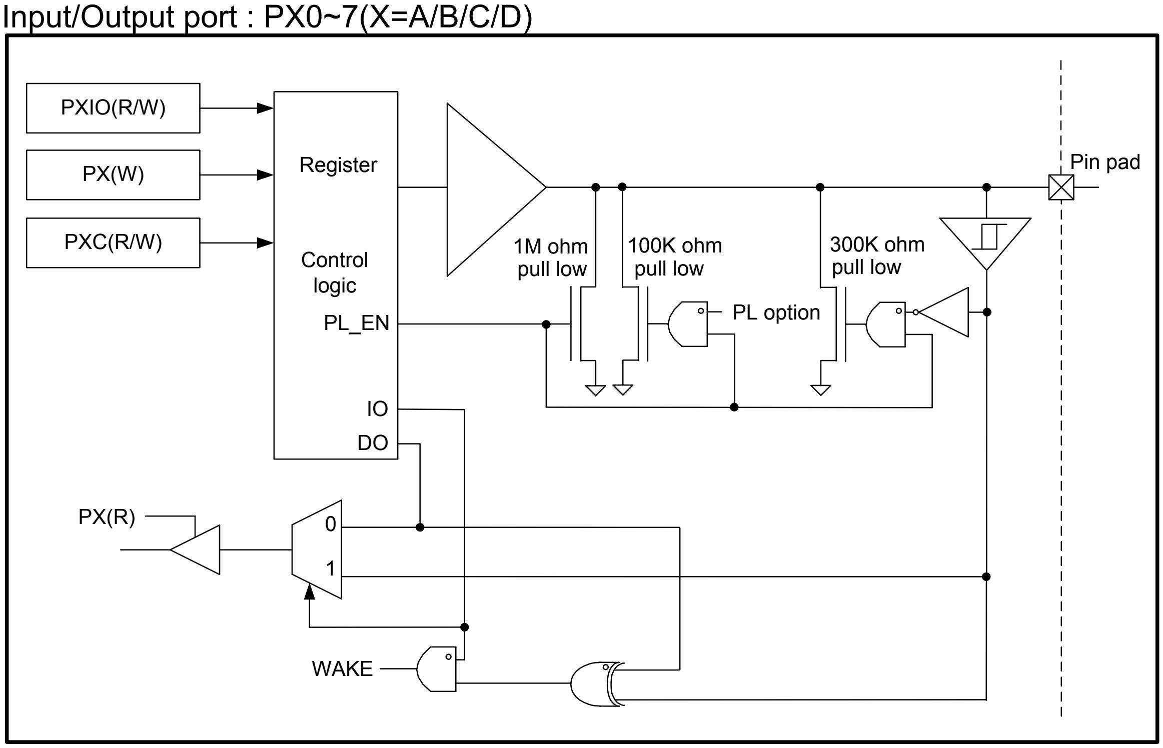


Figure 7-1: I/O port Configuration

The following tables describe the functionalities of each register:

Addr.	Name	R/W	Bit	Data	Description	Default
$30	PAIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$31	PBIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$32	PCIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$33	PDIO	R/W	[7:0]	1/0	1 = Input/0 = Output	FF
$34	PA	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$35	PB	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$36	PC	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$37	PD	R	[7:0]		Read input pad data/output register data	xx
		W	[7:0]	1/0	Write to input or output port register	00
$38	PAC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$39	PBC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$3A	PCC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00
$3B	PDC	R/W	[7:0]	1/0	Pull-Low Resistor Enable/Disable of input; CMOS/Open-Drain of output	00

7.2 Matrix Key Strobe

The NY8L IC utilizes the partial timing of the LCD display to scan the key strobe circuitry. These scanning output pins are SEGi~SEG(i+15) (K0~15). The input part of key strobe circuitry (KI0~7) comprises I/O ports. With the control register KSB, users can define the scan rate, the interrupt mode and the key strobe output, and see as below table.

Addr.	Name	R/W	Bit	Data	Description	Default
$14	KSB	R/W	[3:0]		Key strobe output segment select (SEGi~SEG(i+15)) (i = 0~48) (by body)	0000
			[4]	1/0	All/One segments selected as key strobe output	One
			[5]	0	Key strobe interrupt = key strobe occurs	0
				1	Key strobe interrupt = each key strobe scanning cycle	
			[7:6]	00	Key strobe scan rate = RT[6] (256Hz, F_{SLOW}/128)	RT[6]
				01	Key strobe scan rate = RT[5] (512Hz, F_{SLOW}/64)	
				10	Key strobe scan rate = RT[4] (1KHz, F_{SLOW}/32)	
				11	Key strobe scan rate = RT[3] (2KHz, F_{SLOW}/16)	

In the Figure 7-2, the key strobe scanning inputs (KI0~7) are connected with PX0~7 (X = A/B/C/D) (by body), and key strobe outputs (K0~15) are connected with LCD panel SEGi~(i+15) (i = 0~48) (by body). When any of the buttons is pressed, a HIGH signal will deliver from the segment pins to port side and be detected by program to execute the additional instructions. In this case, if the interrupt enable (IEF[7]) is provided, the interrupt is accepted.

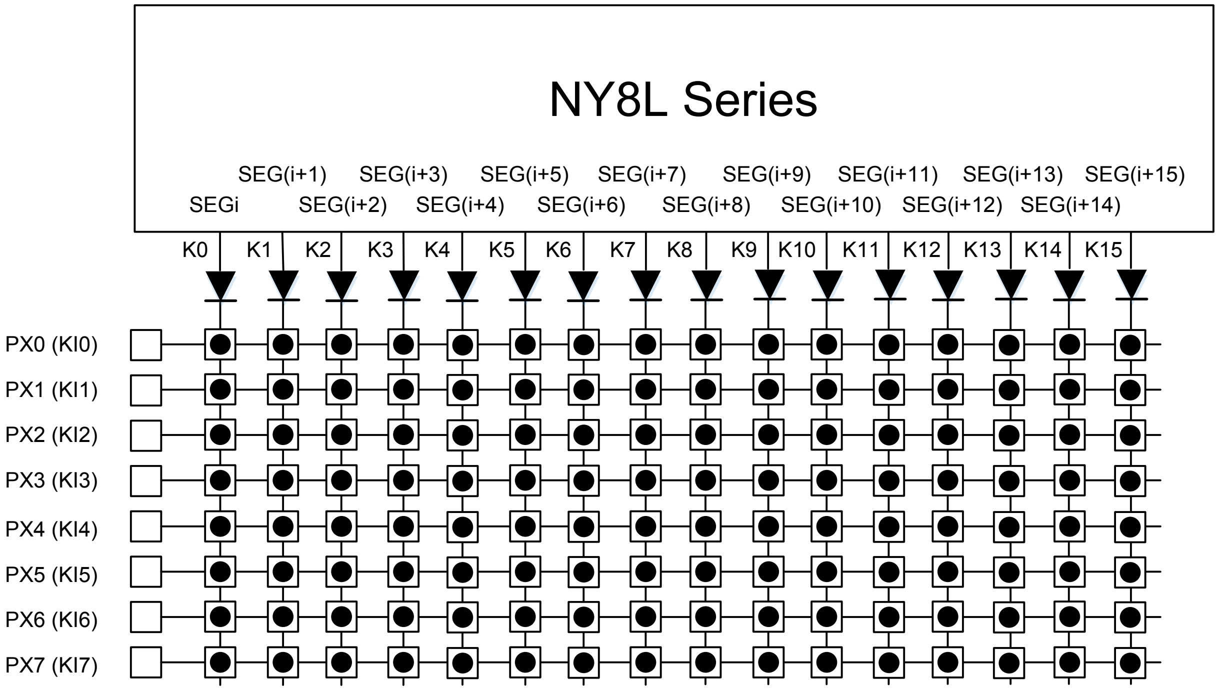


Figure 7-2: The 8x16 key strobe

The procedure of key strobe application is shown below.

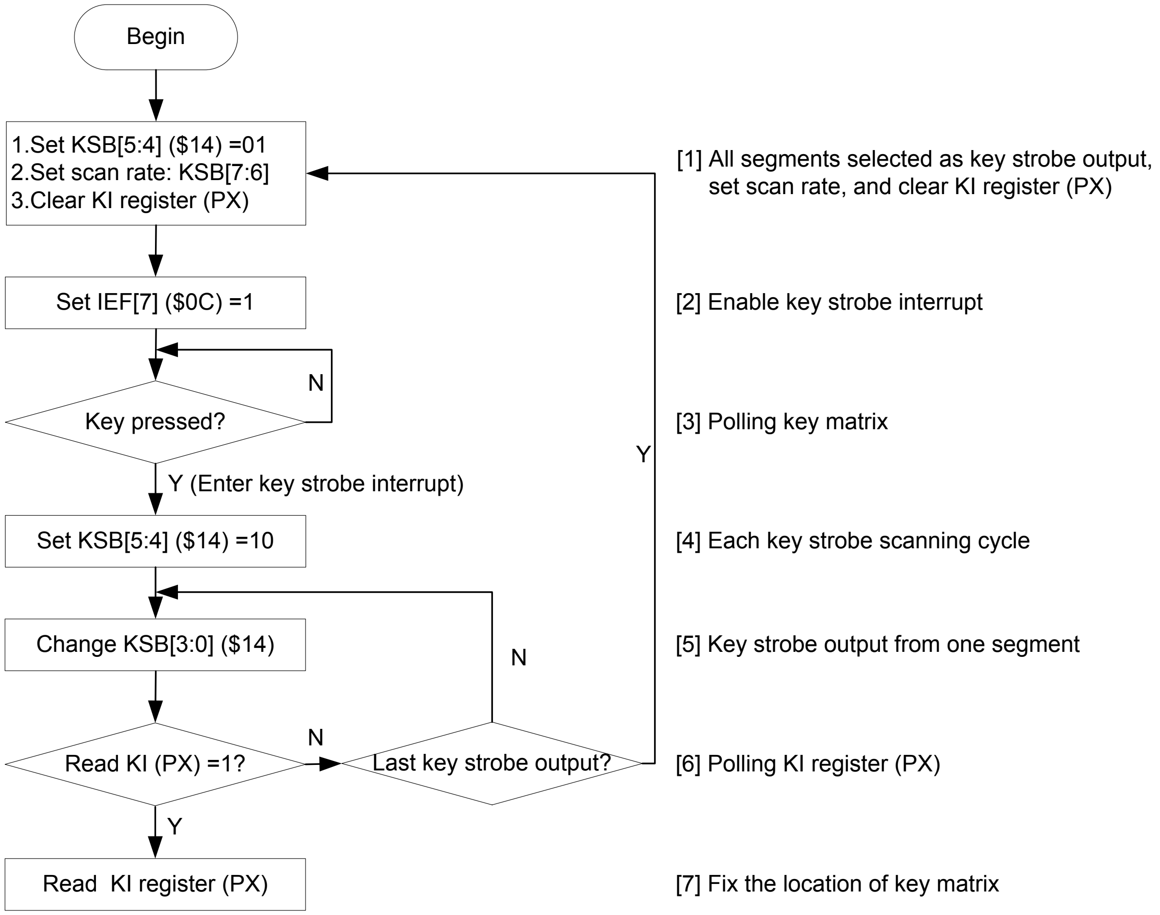

Begin
1.Set KSB[5:4] ($14) =01
2.Set scan rate: KSB[7:6]
3.Clear KI register (PX)
Set IEF[7] ($0C) =1
Key pressed?
N
Y (Enter key strobe interrupt)
Set KSB[5:4] ($14) =10
Change KSB[3:0] ($14)
Read KI (PX) =1?
N
Y
Last key strobe output?
N
Y
Read KI register (PX)
[1] All segments selected as key strobe output, set scan rate, and clear KI register (PX)
[2] Enable key strobe interrupt
[3] Polling key matrix
[4] Each key strobe scanning cycle
[5] Key strobe output from one segment
[6] Polling KI register (PX)
[7] Fix the location of key matrix

Figure 7-3: Flow chart of the key strobe scanning

Besides, the system can also be waked up from Standby mode by key strobe wake-up source, which is similar as key change. Before enter Standby mode, users must set the KI port as input, and clear KI register through writing 0 to PX. Normally key strobe outputs are controlled by KSB[3:0] ($14), when any of the buttons is pressed, a HIGH signal will deliver from the segment pins to port side, which will wake up the system.

Addr.	Name	R/W	Bit	Data	Description	Default
$34	PA	R	[7:0]		KSB: Read register data	xx
		W	[7:0]	1/0	KSB: Write 0 to Clear register data	00
$35	PB	R	[7:0]		KSB: Read register data	xx
		W	[7:0]	1/0	KSB: Write 0 to Clear register data	00
$36	PC	R	[7:0]		KSB: Read register data	xx
		W	[7:0]	1/0	KSB: Write 0 to Clear register data	00
$37	PD	R	[7:0]		KSB: Read register data	xx
		W	[7:0]	1/0	KSB: Write 0 to Clear register data	00

For example, if PC is KSB wake-up source, the code of entering Standby mode is shown below.

```
LDA  OPMD      ;
ORA  #$08      ;
STA  OPMD      ; Set OPMD[3] as high
LDA  #$FF
STA  PCIO      ; Set PC as input
LDA  #$00
STA  PC        ; Clear PC data register

LDA  #$5A
STA  SLP       ; Enter Standby mode
```

On the contrary, for avoiding awaking Halt mode wrongly, the code of entering Halt mode is shown below.

```
LDA  OPMD      ;
AND  #$F7      ;
STA  OPMD      ; Clear OPMD[3] to low
LDA  #$00
STA  PCIO      ; Set PC as output
LDA  #$00
STA  PC        ; Clear PC data register

LDA  #$5A
STA  SLP       ; Enter Halt mode
```

Chapter 8. Other Applications (by option)

The NY8L IC supports many applications with external components, such as EL, SPI, RFC, IR and ADC. The continued sections will describe their functionalities and operations.

8.1 Electroluminescent (EL) Back Light Driver

NY8L provides an EL panel driver for the back light of the LCD panel. The typical EL driver circuitry consists of built-in EL driver and external diode, transistor, resistor and inductance. The circuitry and waveform are shown as Figure 8-1 and Figure 8-2. Users can choose different voltage pump frequency, duty cycle and Enable/Disable frequency through register ELFQ and ELC to operate. To enable the EL block, users still have to enable the relative option.

EL pump (ELP) frequency is divided from fast oscillator clock (8MHz/4MHz/2MHz/500KHz), so it will be slow down as well as F_{FAOS}. EL clear (ELC) frequency is based on slow oscillator clock (32.768KHz). With different frequency and duty for ELP and ELC, the voltage of EL panel can be generated up to 100V or above.

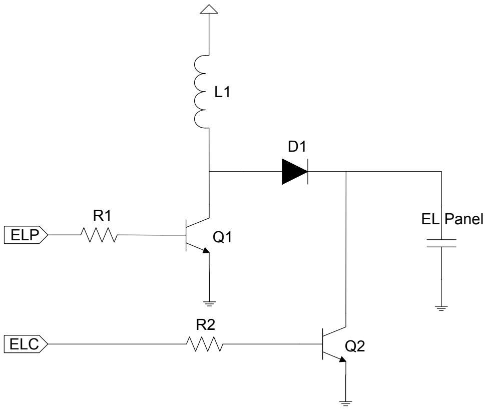


Figure 8-1: The circuitry of EL Driver.

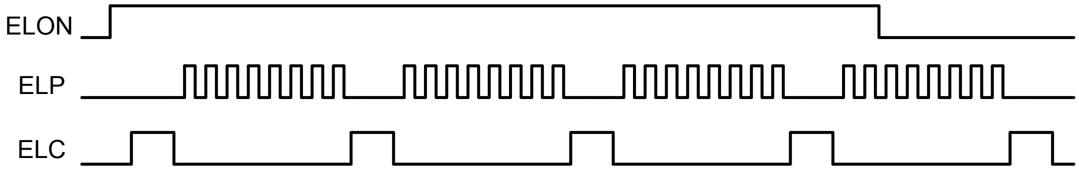


Figure 8-2: The timing waveform of ELP & ELC.

The procedure of EL application goes to three major parts. First, once ELON is enabled, the first pulse of ELC is to discharge uncertain charge stored in EL panel before pumping. Then, the ELP pin will output clocks to pump voltage to the EL panel and the ELC pin will output the pulse right after the last ELP pulse to discharge the EL panel. Eventually, when ELON signal is disabled, the ELC pin will output a pulse to discharge the EL panel after the last pump clock. It insures that there is no residual voltage to cause damage. The relative settings for frequencies and duties are set by the control registers and shown as the following tables.

Addr.	Name	R/W	Bit	Data	Description	Default
$18	ELFQ	R/W	[2:0]	000	ELP frequency = BT[9] (F_{FAOS}/1024)	BT[7]
				001	ELP frequency = BT[8] (F_{FAOS}/512)	
				010	ELP frequency = BT[7] (F_{FAOS}/256)	
				011	ELP frequency = BT[6] (F_{FAOS}/128)	
				100	ELP frequency = BT[5] (F_{FAOS}/64)	
				101	ELP frequency = BT[4] (F_{FAOS}/32)	
				110	ELP frequency = BT[3] (F_{FAOS}/16)	
				111	ELP frequency = BT[2] (F_{FAOS}/8)	
			[4:3]	00	ELC frequency = RT[7] (128Hz, F_{SLOW}/256)	RT[5]
				01	ELC frequency = RT[6] (256Hz, F_{SLOW}/128)	
				10	ELC frequency = RT[5] (512Hz, F_{SLOW}/64)	
				11	ELC frequency = RT[4] (1KHz, F_{SLOW}/32)	
$19	ELC	R/W	[2:0]	000	ELP duty = 1/8	7/8
				001	ELP duty = 2/8	
				010	ELP duty = 3/8	
				011	ELP duty = 4/8	
				100	ELP duty = 5/8	
				101	ELP duty = 6/8	
				110	ELP duty = 7/8	
				111	ELP always high	
			[5:3]	000	ELC duty = 1/8	1/8
				001	ELC duty = 2/8	
				010	ELC duty = 3/8	
				011	ELC duty = 4/8	
				100	ELC duty = 5/8	
				101	ELC duty = 6/8	
				110	ELC duty = 7/8	
				111	ELC always high	
			[7]	1/0	EL Enable/Disable	Disable

8.2 Serial Peripheral Interface (SPI)

NY8L supports only master mode to access serial Flash/SRAM memory. The relative control register SPIMD and SPID are shown as the continued tables. To enable the SPI block, users still have to enable the relative option.

Addr.	Name	R/W	Bit	Data	Description	Default
$1E	SPIMD	R/W	[0]	1/0	SPI shift at Mode3/Mode0	Mode3
		R	[7]	1/0	SPI shift Processing/Done	Done
$1F	SPID	R	[7:0]		Read the shifted-in data from SDI	xx
		W	[7:0]		Latch the data in shift register and starts to shift	xx

Users can set control register SPIMD to select mode3 or mode0 for shifting data out and also read back bit7 of SPIMD to recognize the state of process for further control. SPI always shifts data at rising edge of SCK, and when SPI enters idle mode, SCK keeps high at mode3, otherwise keep low at mode0. For WRITE procedure, it starts with data loaded in register SPID, then shifts 8-bit data out from MSB to LSB at upcoming eight rising edges, and confirms the end of process by reading bit7 of SPIMD. If the bit7 of SPIMD is high, that means system is busy and is not allowed to execute an additional byte of data until it turns to low.

For the connection with external Flash/SRAM, the applied pins are SDI, SDO and SCK. The SDI is the input pin to receive data from the external device, and the SDO is the output pin to deliver data. The SCK is the output pin to offer the clock signal, and configured as F_{CPU}. However, the enable pin for the external device can share with one of I/O ports and define it as output.

For example, the following waveform shows that SPID data is preloaded data “0xAA” and shift out “10101010” in order and shift data into the control register SPID with data “0x55”. Because the programmer knows the process is to send out the data to the external slave device, the data stored in register SPID will be ignored.

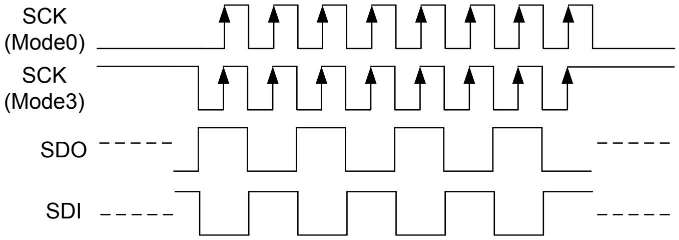


Figure 8-3: The timing waveform of SPI operating mode

8.3 Resistor to Frequency Converter (RFC)

The resistor to frequency converter (RFC) is used to compare two different sensors with the reference resistor individually. The operating principle is based on a RC oscillator network. This Figure 8-4 shows the block diagram of RFC:

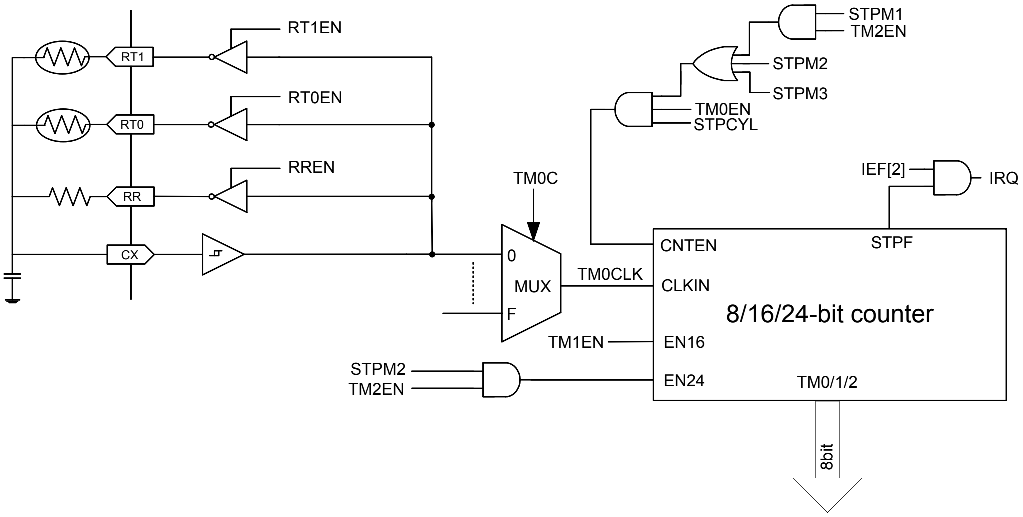


Figure 8-4: The circuitry of RFC.

The architecture of RFC contains four external pins:

CX: the oscillation Schmitt trigger input
RR: the reference resistor output pin
RT0: the temperature sensor output pin
RT1: the humidity sensor output pin (this can also be used as another temperature sensor, or can even be left floating), which can be set as I/O by option

The following definitions of RFC circuitry signals are described below.

RREN/RT0EN/RT1EN: To enable the tri-state buffer to output the reverse signal of CX.
STPCYL: To activate 8/16/24-bit counter.
STPM1/2/3: The control signal defined Timer0 stop mode.
TM0EN/TM1EN/TM2EN: The signals defined the Timer0/1/2 are turned on.
TM0CLK: The clock for Timer0, defined by register TM0C.
IEF2: The bit2 of register IEF.
STPF: The stopped signal as counter is ended to count.
EN16: The signal to extend counter from 8-bit to 16-bit.
EN24: The signal to extend counter from 16-bit to 24-bit.

8.3.1 RC Oscillation Network

The RFC circuitry may build up 3 RC oscillation networks through RR, RT0, or RT1 and CX pins with external resistors and can be disabled by mask option. The oscillation network can be built up by setting register IRC to enable RR, RT0 and RT1 respectively, but only one RC oscillation network is active simultaneously.

Addr.	Name	R/W	Bit	Data	Description	Default
$2F	IRC	R/W	[5:4]	00	RFC Disable	Disable
				01	RFC output the reverse signal of CX from RR	
				10	RFC output the reverse signal of CX from RT0	
				11	RFC output the reverse signal of CX from RT1	

As relative settings are ready, the clock will be generated by the oscillation network and feedback to the counter through CX pin. If counter is enabled, the clock will be to down-count the preloaded value of counter.

The RC oscillation network needs to set up with three simple steps:

1. Connect RR, RT0 and RT1 with a resistor respectively and a capacitor between CX and VSS. The above RFC circuitry shows the connection of these networks.
2. Switch on RREN, RT0EN or RT1EN to output the reverse signal of CX, and then it forms the clock source feedback from RC oscillation network. Those pins will be tri-state as the corresponding enable signal is off.
3. Choose Stop Mode1, Stop Mode2, or Stop Mode3 (TM0C[7:6]=01/10/11) and turn on Timer0, Timer0&1, or Timer0&1&2 to enable 8/16/24-bit counter.

It strongly recommends users to switch on output pin for each RC network before the counter is activated to get better clock signal from CX pin.

The NY8L IC provides 3 counting modes for the operation of the counter. Each mode can trigger Timer0 interrupt if IEF2 is enabled, and will be described in the following sections. Users can program the control register TM0C to choose the mode it will act, and the table is listed as below. For the other settings of timers, refer to chapter 6.1.4~6.1.6.

Addr.	Name	R/W	Bit	Data	Description	Default
$01	TM0C	R/W	[7:6]	00	Timer0 clock stop mode OFF	OFF
				01	Timer0 clock stopped by Timer2 overflow	
				10	Timer0 clock stopped by a full cycle of CX	
				11	Timer0 clock stopped by a full cycle of Timer2 clock	

8.3.2 Timer0 Counting within Timer2 Overflow Cycle (STOP Mode1)

In this mode, Timer2 will dominate the operation of the Timer0(8-bit), or Timer0&1(16-bit) counter. Firstly, the value of Timer2 for counting is loaded by instruction, and the counter won't start to operate until Timer0, or Timer0&1 and Timer2 are enabled. **It strongly recommends users to switch on Timer2 at last.** And then STPCYL goes high, the Timer2 will count down (Y-2) cycles once its falling edge occurs, supposed Y is the initial value. By counting to 0x00, the Timer2 will be stopped to end the process. In this case, if the interrupt enable (IEF[2]) is provided, the interrupt is accepted.

In theory, there are Y full cycles of Timer2 used to control an accurate period of time for CX pin clocking into 8/16-bit counter. **But actually the maximum deviation is perhaps 2 cycle of Timer2**, because Timer2 enable signal is asynchronous to Timer2 clock source. And the deviation will be decreased when Timer2 clock source speed up. Users can read back the content of counter through register TMxD (x=0, 1) by instruction. The procedure of RFC counter controlled by Timer2 is shown as Figure 8-5.

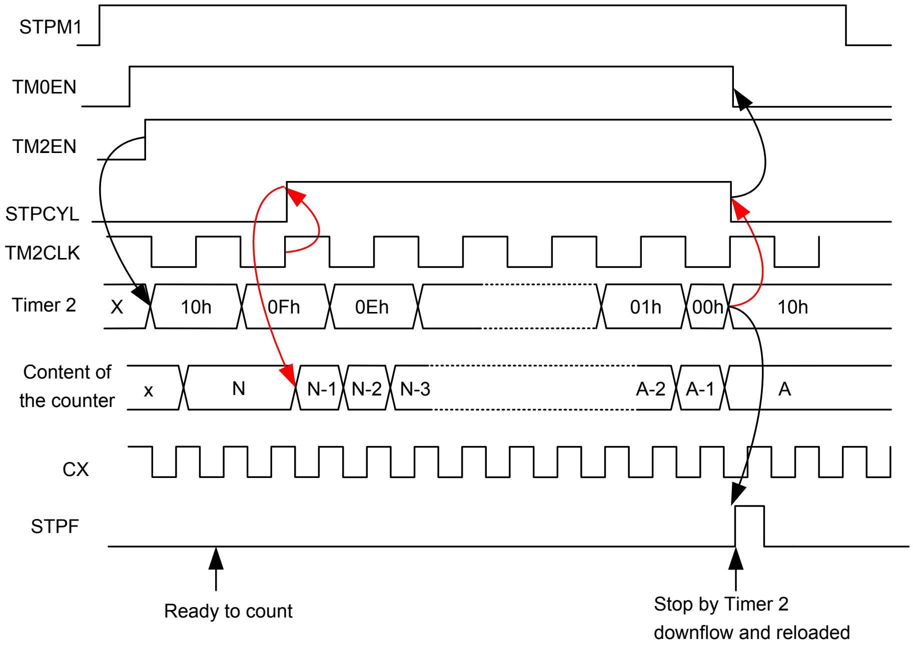


Figure 8-5: Timing of the RFC counter controlled by Timer2 downflow

8.3.3 Timer0 Counting within a Full CX Cycle (STOP Mode 2)

This is the other way to utilize the Timer0 (8-bit), Timer0&1 (16-bit), or Timer0&1&2 (24-bit) counter. Here, CX pin is used to control the enabling of the counter and the clock of Timer0 (TM0CLK), which defined by register TM0C, becomes the clock source of 8/16/24-bit counter. The procedure is quiet similar with previous mode, choose mode it will act, turn on Timer0, Timer0&1, or Timer0&1&2 and load a specific value into the counter at beginning. **It strongly recommends users to switch on Timer0 at last.**

As the first falling edge of CX clock comes out, the STPCYL will goes high to enable the counter. Then the counter will start to count downward until the second falling edge of CX clock occurs. At the time, the stop flag (STPF) will be high to disable TM0CLK and end of procedure. In this case, if the interrupt enable (IEF[2]) is provided, the interrupt is accepted.

Users can read back the content of counter through register TMxD (x=0, 1, 2) by instruction. The timing procedure of this mode is shown as Figure 8-6.

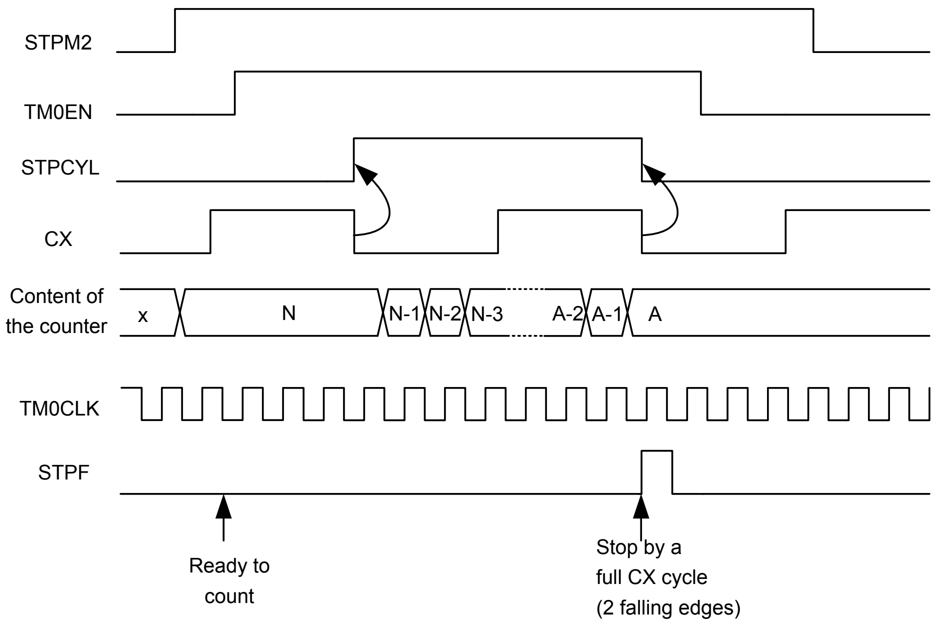


Figure 8-6: Timing of the RFC counter controlled by CX full cycle

8.3.4 Timer0 Counting within a Full Timer2 Clock Cycle (STOP Mode 3)

In this mode, Timer2 clock cycle will dominate the operation of the Timer0(8-bit), or Timer0&1(16-bit) counter. At first, users have to swap to mode 3 through register TM0C, load counting data and turn on Timer0, or Timer0&1 and Timer2 for the initial setting. **It strongly recommends users to switch on Timer0 at last.** Instead of falling edge of CX pin, this mode utilizes two falling edges of Timer2 clock to control the enabling of the 8, or 16-bit counter. The CX pin is applied to be the clock input of the counter. Provided the second falling edge of Timer2 occurs, a pulse of signal STPF would be generated to stop the counting procedure. In this case, if the interrupt enable (IEF[2]) is provided, the interrupt is accepted.

Users can read the content of the counter through register TMxD (x=0, 1) by instruction. The timing procedure of this mode is shown as Figure 8-7.

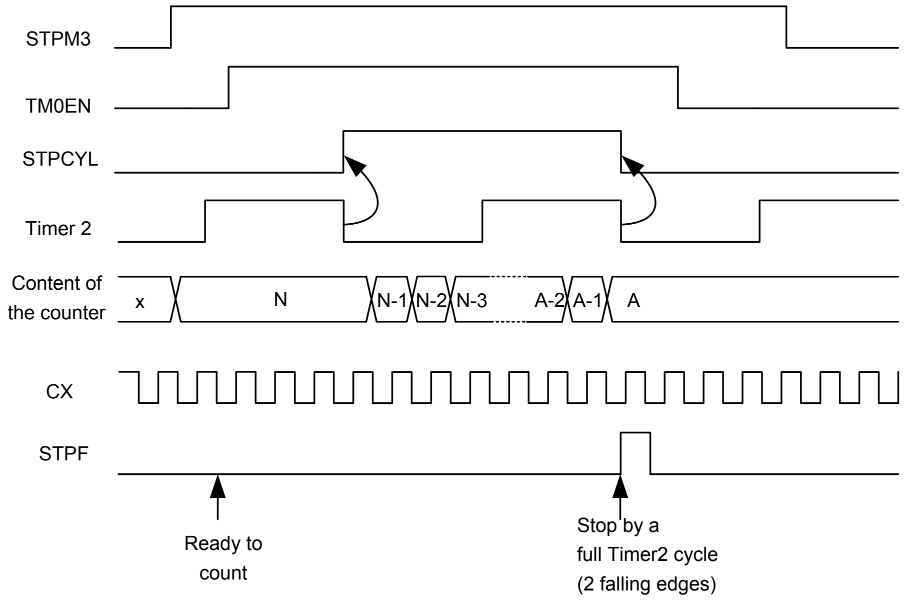


Figure 8-7: Timing of the RFC counter controlled by Timer2 full cycle

8.4 Infrared (IR) Transmitter

The NY8L IC provides an infrared transmit block which is used to send infrared signal. Users can select the frequency of IR carrier, low/high carrier and enable/disable the IR output by option. The IR low/high carrier means that if users option the IR low carrier, the IR output port sends infrared signal when the IR data output register value is low, and vice versa.

The control register IRC is to program its output register and port configuration. If IR option select high(drive) carrier, the initial value of IR data output register bit0 (DO) is 0, and vice versa. Besides, the IR counter can be cleared by writing 0 to IRC[3], and which is always read as high.

Addr.	Name	R/W	Bit	Data	Description	Default
$2F	IRC	R/W	[0]	1/0	IR data output register	H/L (by option)
			[1]	1/0	CMOS/Open-Drain of IR output	CMOS
			[2]	1/0	IR Enable/Disable	Disable
		W	[3]	0	Write 0 to initial IR counter (read as high)	x

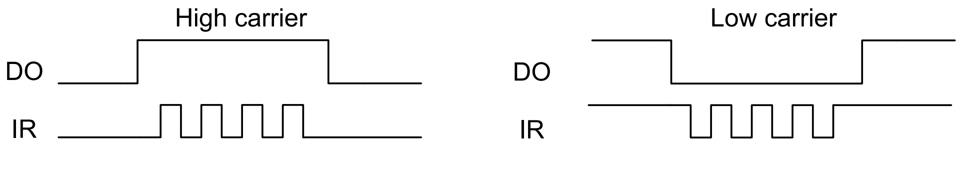


Figure 8-8: Timing of the IR with CMOS output

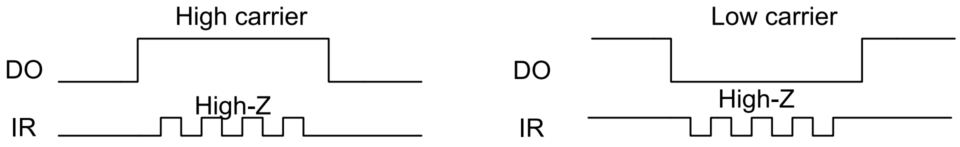


Figure 8-9: Timing of the IR with open-drain output

8.5 Analog to Digital Converter (ADC) (Only for NY8L040A~050A)

The NY8L040A~050A provide an 8-bit analog to digital converter (ADC) with 4 channels, only one of them is active simultaneously. The Analog input is selected from analog signal input pin AIN0~3, and the input signal voltage must be from VSS to VDD.

The ADC clock ADCLK can be selected to be F_{FAOS}/1, F_{FAOS}/2, F_{FAOS}/4 or F_{FAOS}/8. The sampling pulse width can be selected to be ADCLK*1, ADCLK*2, ADCLK*4 or ADCLK*8. Set ADEN=1 before ADC take into operation. Then set START=1, the ADC will start to convert analog signal to digital. EOC=0 means ADC is in processing. EOC=1 indicates ADC is at end of conversion. In this case, if the interrupt enable (IEF[6]) is provided, the ADC interrupt will issue after EOC low go high. The block diagram is as Figure 8-10.

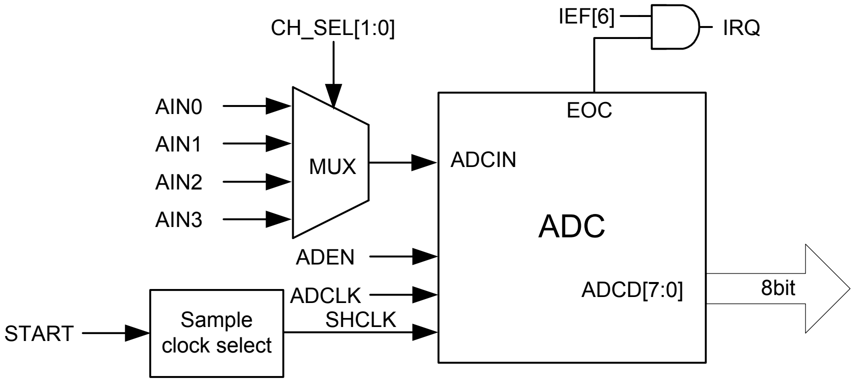


Figure 8-10: ADC block diagram

8.5.1 ADC Clock (ADCLK) and Sampling Clock (SHCLK)

Conversion speed is affected by the selection of the ADC clock (ADCLK) and sampling pulse width (SHCLK). ADCLK is the base clock of ADC, which should be selected from 250KHz (500KHz/2) to 1MHz (8MHz/8). During the operation of ADC, bit operation is synchronized with ADCLK. SHCLK is the duration of analog signal sampling time, larger SHCLK will restore original analog signal level more closely but it will slow down the ADC conversion speed, and vice versa.

The ADC converting time is from START (Start to ADC convert) to EOC=1 (End of ADC convert). The duration is depending on ADC clock rate and sampling clock width.

ADC conversion time ≈ sampling clock width + 12 * ADCLK width.

ADC clock	SHCLK	Conversion Time (ADCLK No.)	F_{FAOS} = 2MHz		F_{FAOS} = 500KHz	
			Time	Rate	Time	Rate
F_{FAOS}/8	ADCLK*8	20	80us	12.5KHz	323us	3.1KHz
F_{FAOS}/4	ADCLK*4	16	32us	31.3KHz	128us	7.8KHz
F_{FAOS}/2	ADCLK*2	14	14us	71.4KHz	56us	17.8KHz
F_{FAOS}/1	ADCLK*1	13	6.5us	153.8KHz	26us	38.5KHz

8.5.2 ADC Operation

Set ADC clock (ADCLK), sampling clock width (SHCLK), and select input channel (AIN0~3). Then set ADEN=1. After setting ADEN=1, it must wait at least 5ms (ADC internal bias stable time) before ADC can operate. Write START to 1 to start an ADC conversion session. During ADC is in processing EOC=0. Polling EOC=1 or wait for ADC interrupt at the end of ADC conversion. The procedure of ADC conversion are shown as Figure 8-11.

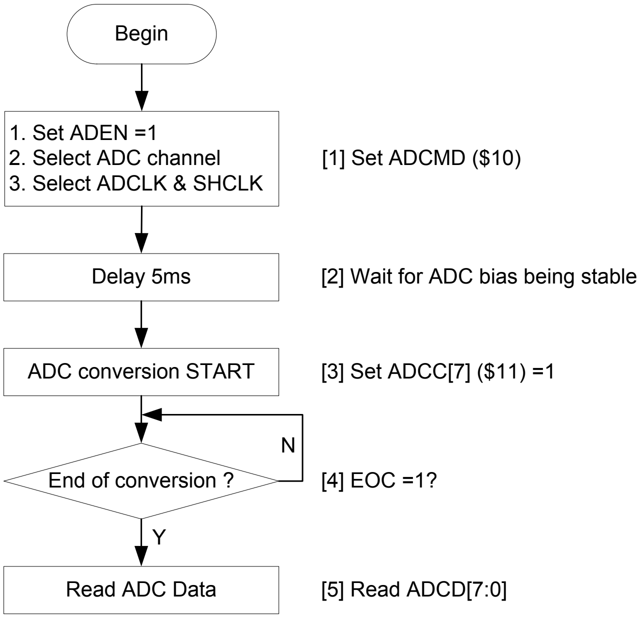


Figure 8-11: Flow chart of the ADC conversion

The timing of ADC conversion is shown as Figure 8-12.

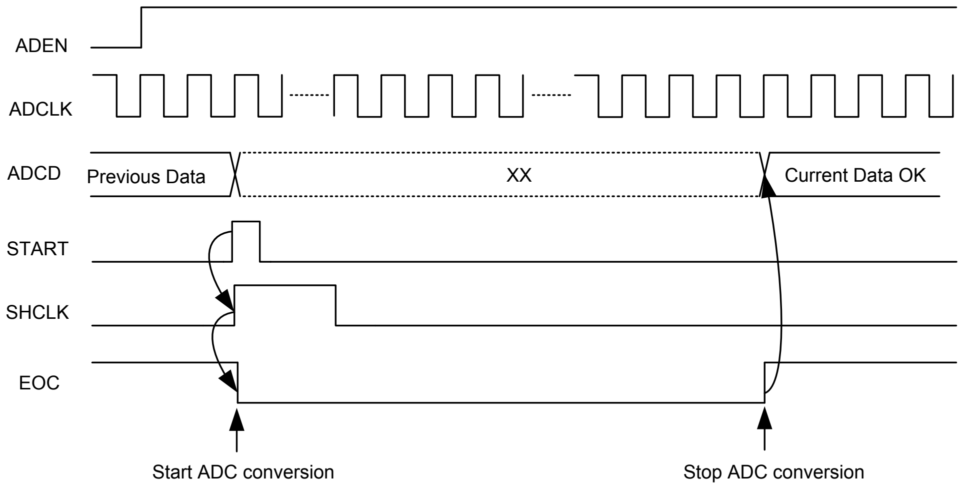


Figure 8-12: Timing of the ADC conversion

The relative settings for ADC are shown as the following tables.

Addr.	Name	R/W	Bit	Data	Description	Default
$10	ADCMD	R/W	[0]	1/0	ADC Enable/Disable	Disable
			[2:1]	00	ADC input from channel 0 (AIN0)	00
				01	ADC input from channel 1 (AIN1)	
				10	ADC input from channel 2 (AIN2)	
				11	ADC input from channel 3 (AIN3)	
			[4:3]	00	ADCLK = F_{FAOS}/1	00
				01	ADCLK = F_{FAOS}/2	
				10	ADCLK = F_{FAOS}/4	
				11	ADCLK = F_{FAOS}/8	
			[6:5]	00	SHCLK = ADCLK*1	00
				01	SHCLK = ADCLK*2	
				10	SHCLK = ADCLK*4	
				11	SHCLK = ADCLK*8	
$11	ADCC	R	[0]	1/0	ADC conversion status: ADC is end-of-convert/ADC is in procession	1
		R/W	[7]	1/0	ADC conversion start/reset	0
$12	ADCD	R	[7:0]		Voltage ADC data output buffer	xx

Chapter 9. Application Circuits

9.1 Application Circuits with Low Loading

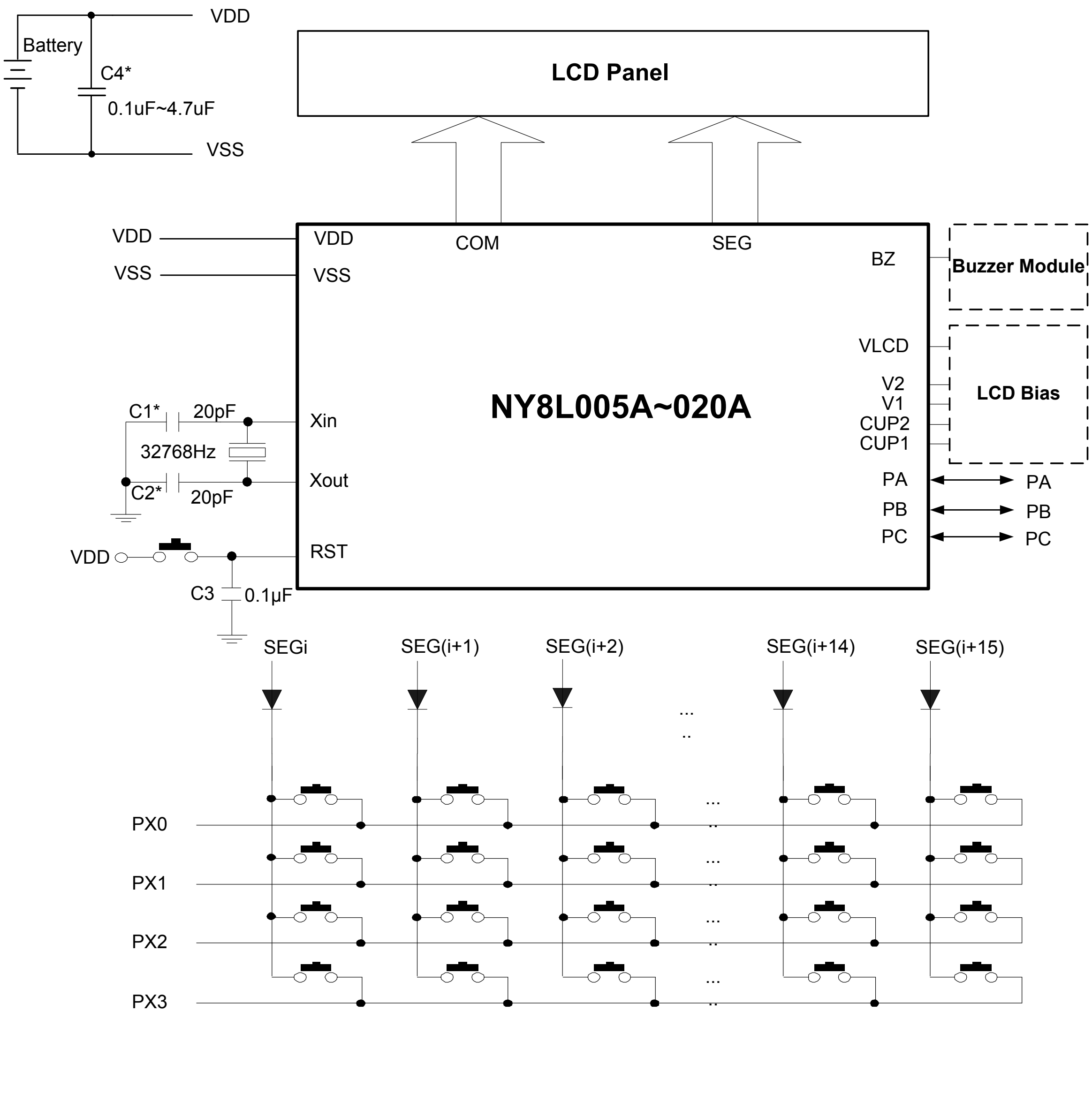


Figure 9-1: The Application Circuits for NY8L005A~020A

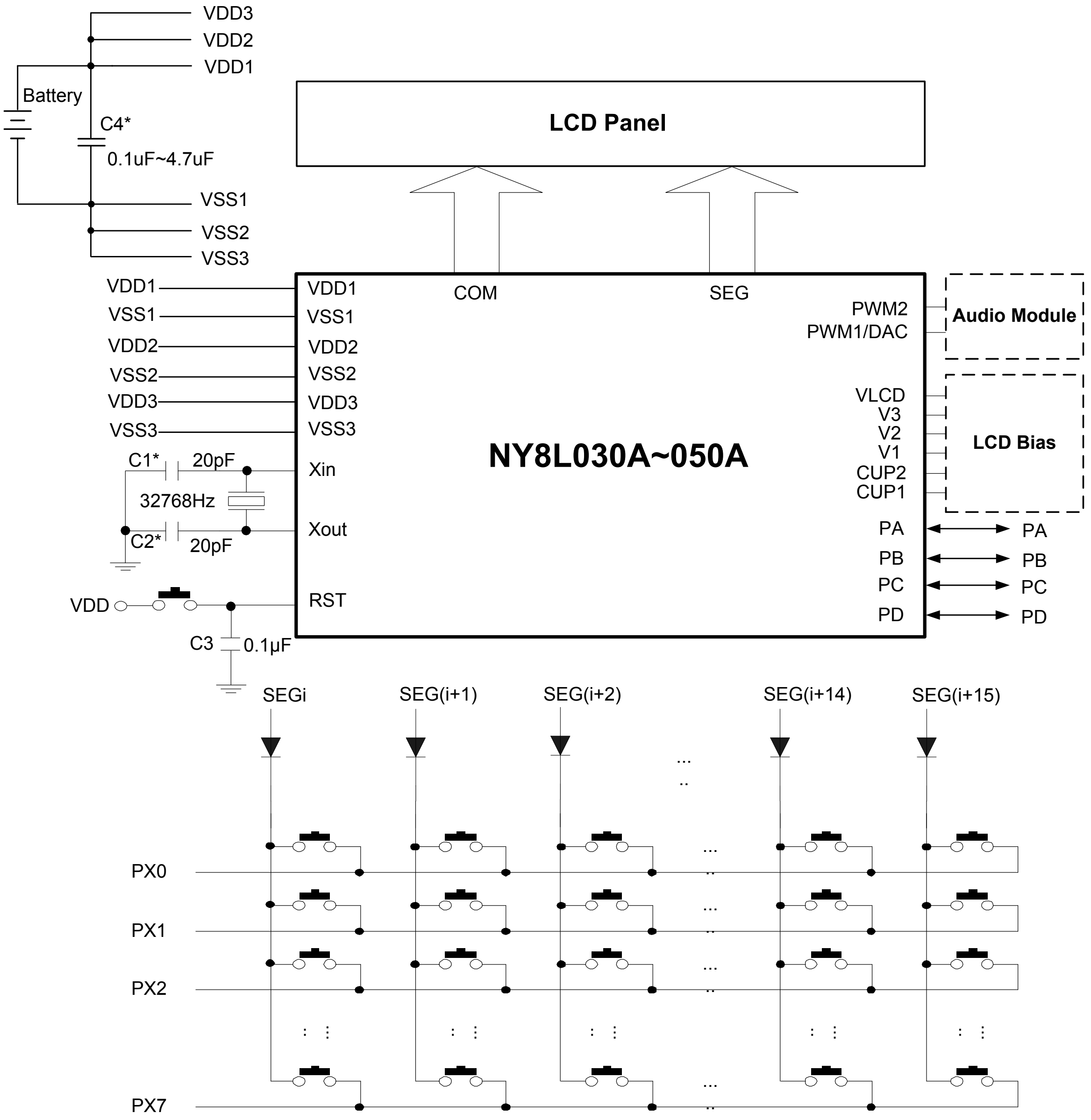


Figure 9-2: The Application Circuits for NY8L030A~050A with Low Loading

PCB Layout Guidelines:

1. VDD must be connected to power input port directly, not the branch of each other.

2. VLCD should be higher than or equal to VDD, otherwise will cause large current.

3. VSS must be connected to ground input directly, not the branch of each other.

4. C1 and C2 (used for XTAL32K) are proposed to be 12~20 pF.

5. C4 is suggested 0.1uF~4.7uF, and should be increased in high volume application.

9.2 Application Circuits with Heavy Loading (Such as Motor, High Brightness LED)

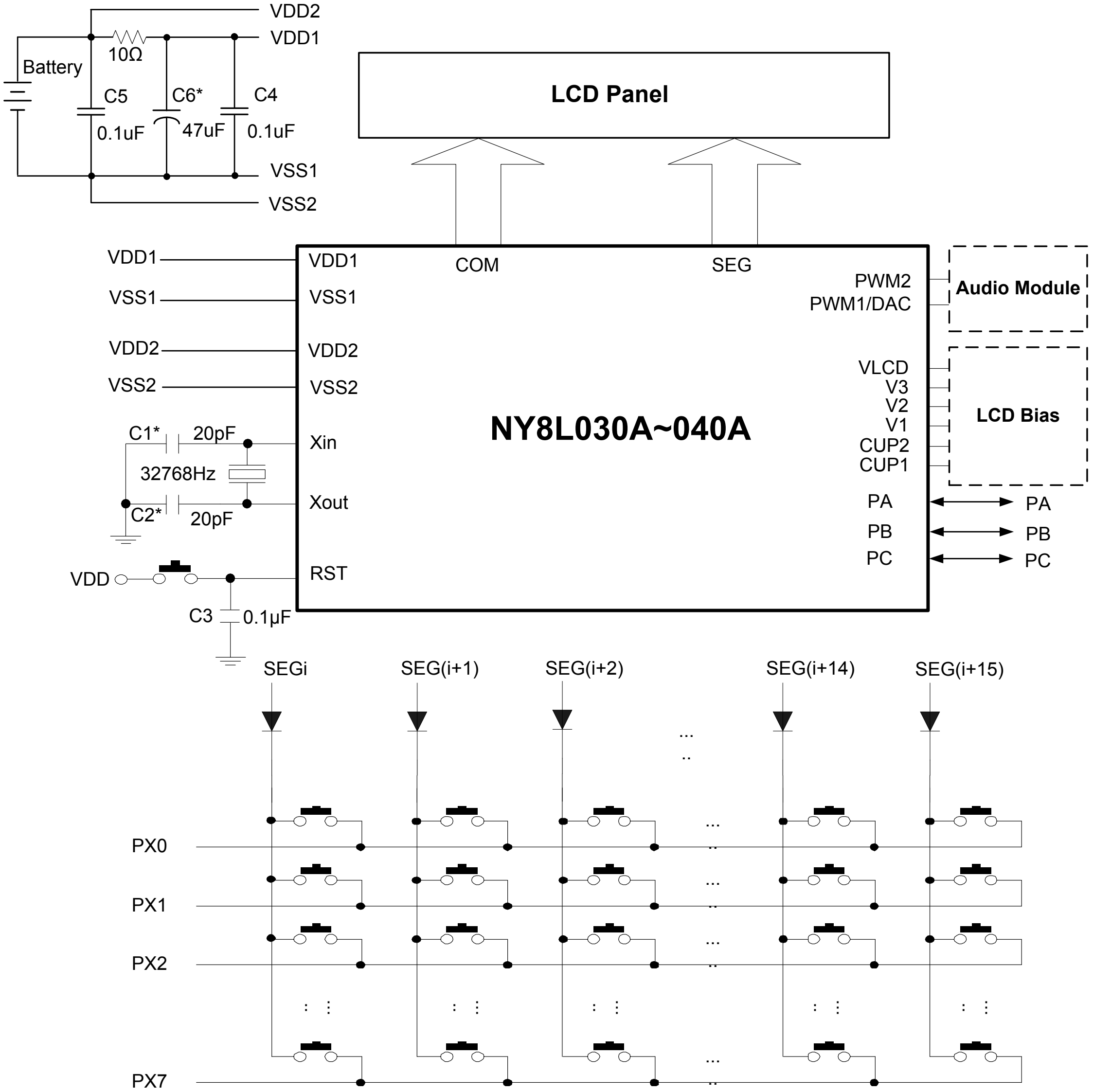


Figure 9-3: The Application Circuits for NY8L030A~040A with Heavy Loading

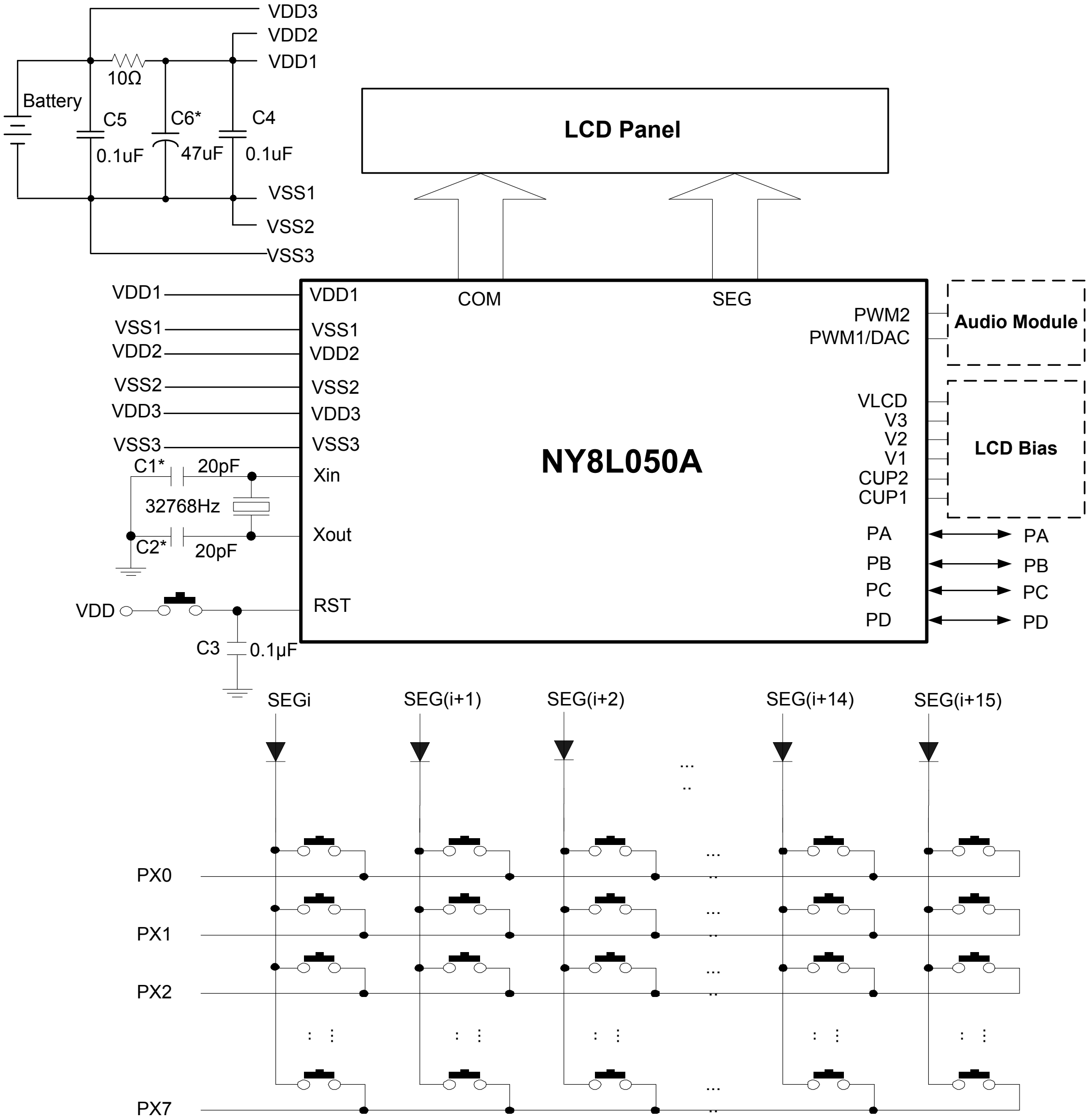


Figure 9-4: The Application Circuits for NY8L050A with Heavy Loading

PCB Layout Guidelines:

1. VDD must be connected to power input port directly, not the branch of each other.

2. VLCD should be higher than or equal to VDD, otherwise will cause large current.

3. VSS must be connected to ground input directly, not the branch of each other.

4. C1 and C2 (used for XTAL32K) are proposed to be 12~20 pF.

5. C6 is suggested 47uF, and should be modified in different loading.

9.3 LCD Bias (VDD for VLCD/V3/V2/V1, or internal Vreg for V1)

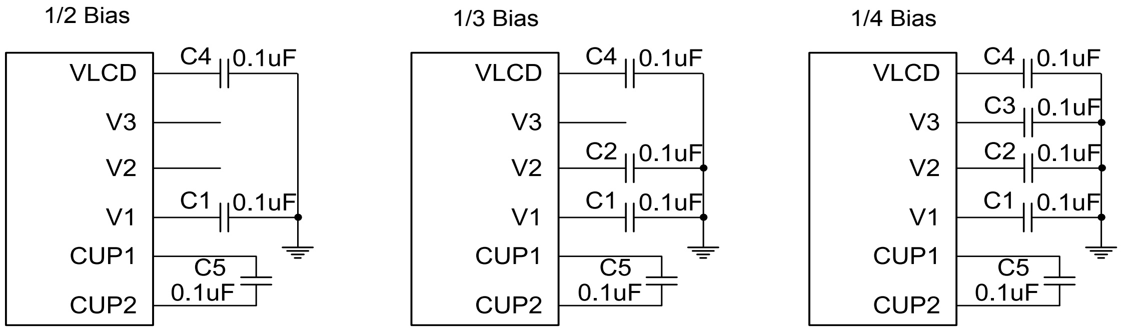


Figure 9-3: The diagram of LCD Bias based on VDD or internal Vreg