SinoMCU 8 MCU

MC30P6280

User Manual

V1.1

Table of contents

1 Product Overview

1.1 Product Features

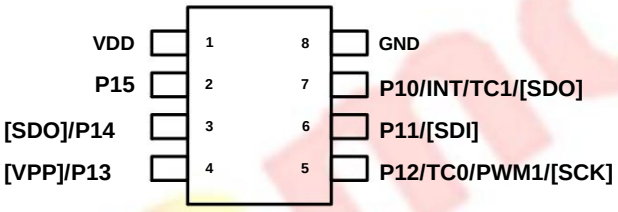
ÿ 8-bit CPU coreÿ Reduced
instruction set, 5-level deep hardware stackÿ CPU
is single clock, only runs under the system main clockÿ Under
the system main clock, FCPU is fixed to 2 divisionÿ
Program memoryÿ
1K×14-bit OTP type program memory (burned once)ÿ 0.5K×14-bit
OTP type program memory (burned twice)ÿ Data memoryÿ 48-byte
SRAM type general
data memory, supporting multiple addressing modes such as direct addressing and indirect addressingÿ 1 group of 6
I/Osÿ P1 (P10~P15)ÿ P13
is an input/open drain
output port, which is a high-voltage VPP input when programmedÿ All ports
have built-in pull-up and pull-down resistors, which can be enabled
individuallyÿ Except for P13, the remaining ports can select open
drain or push-pull outputÿ All ports support keyboard interrupt wake-up function
and can be enabled
individuallyÿ System clock sourceÿ Built-in high-frequency RC oscillator (16MHz), its 1/2/4/8/16/32 divided clock can be
used as the system main clock sourceÿ Built-in low-frequency RC oscillator (40KHz), can be used as the system
main clock source, or the
system low-frequency clock sourceÿ System working modeÿ
Running mode: CPU runs under the system main clockÿ Sleep mode: CPU
stops running, the system main clock source stops
workingÿ Internal self-oscillating watchdog
counter (WDT)ÿ Shared prescaler with timer T0ÿ Overflow time is configurable: 3.6ms/
14ms/58ms/230ms (no prescaler)ÿ Working mode is configurable: turn on WDT, turn off WDT, and can
also be turned on or off by softwareÿ 2 timers
ÿ 8-bit timer T0, supports system low-frequency clock, can realize external counting function, and shares prescaler with WDT
ÿ 8-bit timer T1, can realize external counting and PWM functions ÿ
Interrupt ÿ
External interrupt (INT), keyboard interrupt (P10~P15), timer interrupt (T0~T1)
ÿ Low voltage reset LVR: Off/1.8V/2.0V/2.7V/3.6V ÿ Low voltage
detection LVD
ÿ 1.8V/2.0V/2.1V/2.2V/2.4V/2.5V/2.6V/2.7V/2.8V/2.9V/3.0V/3.2V/3.3V/3.6V/4.0V/4.2V
ÿ Operating voltage
ÿ VLVR27 ~ 5.5V @ Fcpu = 0~8MHz ÿ VLVR20 ~
5.5V @ Fcpu = 0~4MHz ÿ VLVR18 ~ 5.5V @ Fcpu
= 0~1MHz
ÿ Package type: SOP8/DIP8/SOT23-6

1.2 Ordering Information

Product Name	Package	Remark
MC30P6280A0H	SOP8	
MC30P6280A0A	DIP8	
MC30P6280A0T	SOT23-6	
MC30P6280A1T	SOT23-6	

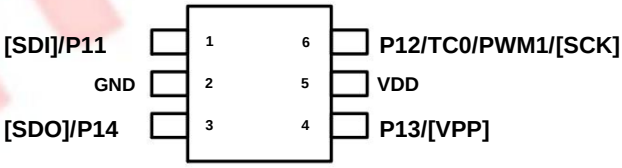
1.3 Pinout

MC30P6280A0H/A0A



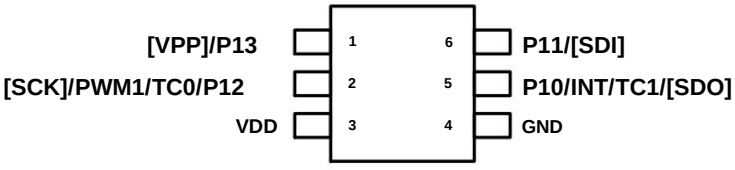
SOP8/DIP8

MC30P6280A0T



SOT23-6

MC30P6280A1T



SOT23-6

1.4 Port Description

Port Name	type	Functional Description
VDD	P	Power
GND	P	
P10~P12ÿP14~P15	D	GPIO (push-pull/open-drain output selectable), internal pull-up/pull-down
P13	D	GPIO (open drain output), internal pull-up/down
INT	DI	External interrupt input
TC0~TC1	DI	External counting input of timer T0~T1
PWM1	DO	PWM output of timer T1
SCK, SDI, SDO	D	Programming clock/data input/data output interface
VPP	P	Programming high voltage input

Note: P-power supply; D-digital input/output, DI-digital input, DO-digital output; A-analog input/output, AI-analog input, AO-analog output.

2 Electrical characteristics

2.1 Limit parameters

Parameters	symbol	value	unit
Power supply voltage	VDD	-0.3~6.0	V
Input voltage	come	-0.3~VDD+0.3	V
Operating temperature	Facing	-40~85	ÿ
Storage	Tstg	-65~150	ÿ
Temperature Into VDD Maximum	IVDDmax	50	mA
Current Out of GND Maximum Current	IGNDmax	50	mA

Note: If the chip working conditions exceed the limit value, it will cause permanent damage; if the chip works under extreme conditions for a long time, its reliability will be affected.

2.2 DC electrical characteristics

VDD=5VÿT=25ÿ

Property Symbol Port		condition	Min.	Typ.	Max.	Unit
Working voltage VDD	VDD	Fcpu=8MHz@HIRC(16M)/2	VLVR27		5.5	V
		Fcpu=4MHz@HIRC(8M)/2	VLVR20		5.5	
		Fcpu=2MHz@HIRC(4M)/2	VLVR20		5.5	
		Fcpu=1MHz@HIRC(2M)/2	VLVR18		5.5	
		Fcpu=500KHz@HIRC(1M)/2	VLVR18		5.5	
		Fcpu=250KHz@HIRC(500K)/2	VLVR18		5.5	
		Fcpu=20KHz@LIRC(40K)/2	VLVR18		5.5	
Input leakage current Ileak All input pins VDD=5V			-1		1	uA
Input high level Vih All input pins		SMT enabled, SMTVS configured	0.8VDD			V
		SMT enabled, SMTVS configured	2.0			V
		SMT Close	2.0			V
Input low level Vil All input pins		SMT enabled, SMTVS configured			0.2VDD	V
		SMT enabled, SMTVS configured			0.8	V
		SMT Close			1.0	V
Output source current Ioh Push-pull output pin		Voh=VDD-0.6V, IOHS configuration		14		mA
		Voh=VDD-0.6V, IOHS configuration		4		mA
Output sink current Iol All output pins Vol=0.6V				20		mA
Pull-up resistor Rpu Pull-	P1	VDD=5V, Vin=0		20		Kÿ
down resistor Rpd	P1	Vin=VDD=5V		20		Kÿ
Running mode power consumption Irun	VDD	Fcpu=8MHz@HIRC(16M)/2		2.9		mA
		Fcpu=4MHz@HIRC(8M)/2		1.6		mA

			Fcpu=2MHz@HIRC(4M)/2		900		uA
			Fcpu=1MHz@HIRC(2M)/2		580		uA
			Fcpu=500KHz@HIRC(1M)/2		420		uA
			Fcpu=250KHz@HIRC(500K)/2		330		uA
			Fcpu=20KHz@LIRC(40K)/2		8		uA
Sleep mode power consumption Istop		VDD	LIRC Off		0.1	1	uA
			LIRC On		0.9	3	uA
Low voltage detection voltage VLVD		VDD		-10%		+10% V	
LVD response time TLVD				1	50	200	us
Low voltage reset voltage VLVR		VDD	LVRVS Configuration	-15%		+15% V	
Power-on reset voltage VPOR		VDD	LVR Off	-30% 1.5	+30% V		
LVD/LVR Hysteresis Voltage		VDD			6% 12%		

Note: In the conditional items, if no indication is given, the module is turned off by default, and the port voltage is VDD/GND if the irrelevant port status is output without load, input or open-drain output is high.

2.3 AC electrical characteristics

characteristic	symbol	condition	Min.	Typ.	Max.	Unit
HIRC oscillation frequency FHIRC		VDD=2.0V~5.5V, T=25ÿ, VDD external 0.1uF capacitor to ground	-3%	16	+3%	MHz
		VDD=2.0V~5.5V, T=-40ÿ~85ÿ, VDD external 0.1uF capacitor to ground	-5%		+5%	
LIRC Oscillator Frequency	FLIRC	VDD=5V, T=25ÿ	-50%	40	+50%	KHz

3 CPU and Memory

3.1 Instruction Set

The instruction set of the chip is a reduced instruction set. Except for program jump instructions, the rest of the instructions are single-cycle instructions, that is, the execution time is 1 instruction. instruction cycle; all instructions are single-word instructions, that is, the instruction code only occupies 1 program memory address space.

Instruction Summary Table

Mnemonics	illustrate	operate	Cycle length flag		
ADDAR R	Add R and ACC and store the result in ACC	R+ACCÿACC	1	1 C, DC, Z	
ADDRA R	Add R and ACC and store the result in R	R+ACCÿR	1	1 C, DC, Z	
ADCAR R	Add R and ACC (with C flag), and store the result in ACC	R+ACC+CÿACC	1	1 C, DC, Z	
ADCRA R	Add R and ACC (with C flag), and store the result in R	R+ACC+CÿR	1	1 C, DC, Z	
RSUBAR R	Subtract R from ACC and store the result in ACC	R-ACCÿACC	1	1 C, DC, Z	
RSUBRA R	Subtract R from ACC and store the result in R	R-ACCÿR	1	1 C, DC, Z	
RSBCAR R	Subtract R from ACC (with C flag), and store the result in ACC	R-ACC-/CÿACC	1	1 C, DC, Z	
RSBCRA R	Subtract R from ACC (with C flag), and store the result in R	R-ACC-/CÿR	1	1 C, DC, Z	
FLOOR R	R and ACC are ANDed, and the result is stored in ACC	R and ACCÿACC	1	1	WITH
SECOND R	R and ACC are ANDed, and the result is stored in R	R and ACCÿR	1	1	WITH
ORAR R	R and ACC are ORed, and the result is stored in ACC	R or ACCÿACC	1	1	WITH
ORRA R	R and ACC are ORed and the result is stored in R	R or ACCÿR	1	1	WITH
XORAR R	XOR operation between R and ACC, and the result is stored in ACC	R xor ACCÿACC	1	1	WITH
XORRA R	XOR operation between R and ACC, and the result is stored in R	R xor ACCÿR	1	1	WITH
COMAR R negates R and stores the result in ACC		R Negation ÿ ACC	1	1	WITH
COMR R negates R and stores the result in R		R Negation ÿ R	1	1	WITH
RLAR R	R Circular left shift (with C flag), the result is stored in ACC	R[7]ÿC R[6:0]ÿACC[7:1] CÿACC[0]	1	1	C
RLR R	R Circular left shift (with C flag), the result is stored in R	R[7]ÿC R[6:0]ÿR[7:1] CÿR[0]	1	1	C
RRAR R	R Circular right shift (with C flag), the result is stored in ACC	R[0]ÿC R[7:1]ÿACC[6:0] CÿACC[7]	1	1	C
RRR R	R Circular right shift (with C flag), the result is stored in R	R[0]ÿC R[7:1]ÿR[6:0] CÿR[7]	1	1	C
SWAPAR R swaps the high and low nibbles of R and stores the result in ACC		R[7:4]ÿACC[3:0] R[3:0]ÿACC[7:4]	1	1	-
SWAPR R swaps the high and low nibbles of R and stores the result in R		R[7:4]ÿR[3:0] R[3:0]ÿR[7:4]	1	1	-

MOVRA R stores ACC in R		ACCýR	1	1	·
MOVAR R stores R in ACC		RýACC	1	1	WITH
MOVR R stores R into R		RýR	1	1	WITH
CLRA	Clear ACC	0ýACC	1	1	WITH
CLRR R clear R		0ýR	1	1	WITH
INCR R	R increments by 1	R+1ýR	1	1	WITH
AIM R	R plus 1, the result is stored in ACC	R+1ýACC	1	1	WITH
DECR R	R Decrement by 1	R-1ýR	1	1	WITH
DECAR R	R minus 1, the result is stored in ACC	R-1ýACC	1	1	WITH
JZR R	R increments by 1; if the result is 0, skip the next instruction	R+1ýR; if the result is 0, then PC+2ýPC	1/2	1	·
JZAR R	R plus 1, the result is stored in ACC; if the result is 0, skip the next instruction R+1ýACC; if the result is 0, PC+2ýPC 1/2			1	·
DJZR R	R is decremented by 1; if the result is 0, the next instruction is skipped	R-1ýR; if the result is 0, then PC+2ýPC	1/2	1	·
DJZAR R	R minus 1, the result is stored in ACC; if the result is 0, skip the next instruction R-1ýACC; if the result is 0, PC+2ýPC		1/2	1	·
BCLR R,b clears the bth bit of R to 0		0ýR[b]	1	1	·
BSET R,b Set the bth bit of R to 1		1ýR[b]	1	1	·
JBCLR R,b If the bth bit of R is 0, skip the next instruction		If R[b]=0, then PC+2ýPC	1/2	1	·
JBSET R,b If the bth bit of R is 1, skip the next instruction		If R[b]=1, then PC+2ýPC	1/2	1	·
ADDAI K	Add K and ACC and store the result in ACC	K+ACCýACC	1	1 C, DC, Z	
ISUBAI K	Subtract K from ACC and store the result in ACC	K-ACCýACC	1	1 C, DC, Z	
IF K	K and ACC are ANDed, and the result is stored in ACC	K and ACCýACC	1	1	WITH
WEATHER K	K and ACC are ORed, and the result is stored in ACC	K or ACCýACC	1	1	WITH
CHORUS K	XOR operation between K and ACC, and the result is stored in ACC	K xor ACCýACC	1	1	WITH
MOVAI K deposits K into ACC		KýACC	1	1	·
CALL K Subroutine call		PC+1ýTOS KýPC[10:0]	2	1	·
GOTO K unconditional jump		KýPC[10:0]	2	1	·
RETURN Return from a subroutine		TOSýPC	2	1	·
RETAI K returns from the subroutine and stores K in ACC		TOSýPC KýACC	2	1	·
RETIE returns from interrupt		TOSýPC 1ýGIE	2	1	·
NOP	No operation	No operation	1	1	·
DAA	After adding the BCD code, adjust the value of ACC to BCD code ACC (HEX code) ÿ ACC (BCD code)		1	1	C
DSA	After BCD code subtraction, adjust the value of ACC to BCD code ACC (HEX code) ÿ ACC (BCD code)		1	1	·
CLRWDT Clears the watchdog timer		0ýWDT	1	1 TO,PD	
STOP	Entering low power mode	0ýWDT; CPU halt	1	1 TO,PD	

Note:

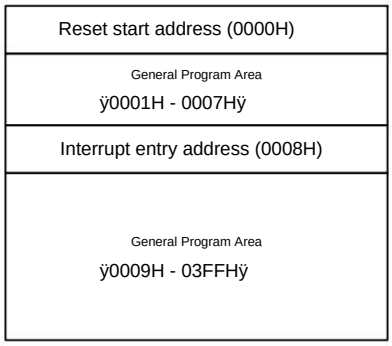
1ÿ ACC-arithmetic logic unit accumulator, R-data memory, K-immediate data;

2ÿ For conditional jump instructions, if the jump condition is met, the instruction takes cycles, otherwise it only takes cycles;

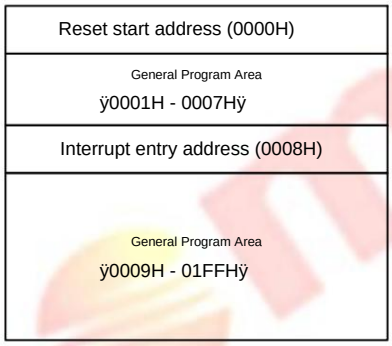
3.2 Program Memory

The program memory of the chip is an OTP type memory, and the address space range of the memory can be selected through the configuration word PAGE.

The address space range of 1K×14 bits is 0000H~03FFH, and can only be programmed once. The address allocation is shown in the figure below:



The address space range of 0.5K×14 bits is 0000H~01FFH, which can be programmed twice. The address allocation is shown in the figure below:



3.3 Data Storage

The data memory includes the general data memory GPR and the special function register SFR. For specific address allocation, refer to the table below. GPR/SFR Can be addressed directly or indirectly via INDF.

Data memory area address mapping table

Address Type 0/8			1/9	2/A	3/B	4/C	5/D	6/E	7/F
00H-07H	SFR	INDF T0CNT		PCL	STATUS	FSR		P1	
08H-0FH		MCR P1KBCR PCLATH PDCON ODCON PUCON NOT							INTF
10H-3FH GPR		General data memory area							
40H-47H	SFR		T0CR					DDR1	
48H-4FH					TMRCR T1CR T1CNT T1LOAD T1DATA				
50H-7FH Reserved									

Note: The gray addresses in the above table are reserved for the system. The read data is uncertain and the write operation may affect the normal operation of the chip.

Data memory addressing mode address composition

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																Addressing Mode
/	/	/	/	/	/	/	/	/	7-bit address from instruction							Direct addressing mode
/	/	/	/	/	/	/	/	/	FSR							Indirect Addressing Mode

In direct addressing mode, the lower 7 bits of the instruction are used as the data memory address. The addressing range is 00H~7FH through instruction access. Example:

Write data 55H to data memory address 10H via direct addressing mode

```
MOVAI   55H
MOVRA   10H     ; Write data 55H to data memory address 10H
```

Indirect addressing mode uses FSR as the data memory address pointer and accesses it through INDF. The addressing range is 00H~7FH. Example:

Write data 55H to data memory address 10H through indirect addressing mode

```
MOVAI   10H
MOVRA   FSR
MOVAI   55H
MOVRA   INC     ; Write data 55H into the data memory pointed to by FSR
```

3.4 Stack

5-level stack depth. When the program responds to an interrupt or executes a subroutine call instruction, the CPU will automatically push the PC to the stack for storage.

When a return instruction or subroutine return instruction is issued, the top data of the stack is automatically popped out and loaded into the PC.

3.5 Control Registers

Data Pointer Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR	-	FSR6	FSR5	FSR4	FSR3	FSR2	FSR1	FSR0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	1	0	0	0	0	0	0	0

BIT[6:0] **FSR[6:0]** – Data Pointer Register

FSR: pointer for indirect addressing mode.

Indirect addressing register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INC	INDF7	INDF6	INDF5	INDF4	INDF3	INDF2	INDF1	INDF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	X	X	X	X	X	X	X	X

BIT[7:0] **INDF[7:0]** – Indirect addressing register

INDF: INDF is not a physical register. Addressing INDF is actually accessing the data memory address pointed to by FSR.

Question, thereby realizing the indirect addressing function.

Program pointer counter low byte

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCL	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

BIT[7:0] **PC[7:0]** – Program Pointer Counter lower 8 bits

Program Pointer Counter High Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCLATH							PCH1	PCH0
R/W							R/W	R/W
Initial Value							0	0

BIT[1:0] **PCH[1:0]** – Program Pointer Counter Upper 2 Bits Register

The program pointer counter (PC) has the following operating modes:

ÿ Sequential running instructions: PC = PC + 1;

ÿ Branch instruction GOTO/CALL: PC = lower 10 bits of instruction code;

ÿ Return instruction RETIE/RETURN/RETAI: PC = top of stack (TOS);

For PCL operation instructions (assembly mode):

ÿ Addition instruction for PCL operation: PC = (PC[9:0]+ALU[7:0]);

ÿ For other instructions of PCL operation: PC = {PC[9:8]:ALU[7:0](ALU operation result)};

For PCL operation instructions (C compilation mode):

ÿ For PCL operation instruction: PC = {PCLATH[1:0]:ALU[7:0](ALU operation result)};

CPU Status Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS WKUP				TO	PD	WITH	DC	C
R/W	R/W			R	R	R/W	R/W	R/W
Initial Value	0			1	1	X	X	X

BIT[7] **WKUP** – wakeup source flag

0: The chip is awakened by other wake-up sources;

1: The chip is awakened by external interrupt or keyboard interrupt;

BIT[4] **TO** – Watchdog overflow flag

0: WDT overflow occurs;

1: Power-on reset, or execute CLRWDT/STOP instruction;

BIT[3] **PD** – Enter low power mode flag 0:
Execute STOP instruction;
1: Power on reset, or execute CLRWDT instruction;

BIT[2] **Z** – Zero flag 0: The
result of an arithmetic or logic operation is not zero; 1:
The result of an arithmetic or logic operation is zero;

BIT[1] **DC** – Half-byte carry/borrow flag 0:
no carry in addition; borrow in subtraction; 1: carry in addition; no borrow
in subtraction;

BIT[0] **C** – Carry/borrow flag 0:
No carry during addition; borrow during subtraction; shift out logic 0; 1: Carry during
addition; no borrow during subtraction; shift out logic 1;

3.6 User Configuration Word

To ensure the normal operation of the system, the chip will pre-store the configuration information of key modules in a separate memory area, load the configuration information into the register after power-on or other reset occurs, and control the working status of key modules through the register. The user-selectable content in this part of the memory is the user configuration word, which can be configured and burned when burning the user program code.

The user configuration word of the chip is defined as follows:

symbol	Functional Description
PAGE	ROM burning mode settings: 0.5K capacity MTP mode, 1st programming; 0.5K capacity MTP mode, 2nd programming; 1K capacity OTP mode;
OSCM	System main clock source selection: FOSC = post-divided clock of internal high-frequency RC oscillator clock FHIRC; FOSC = internal low frequency RC oscillator clock FLIRC;
HIRCDS	HIRC post-divided output selection: FOSC=FHIRC/1ÿFHIRC/2ÿFHIRC/4ÿFHIRC/8ÿFHIRC/16ÿFHIRC/32ÿ
LVRVS	LVR reset voltage selection: (LVR voltage should meet the operating voltage characteristics determined by FCPU) LVR off (VPOR=1.5V); 1.8V; 2.0V; 2.7V; 3.6V;
WDTM	WDT mode setting: WDT is always off; WDT is always on;
WDTT	WDT overflow time selection (without prescaler): 3.6msÿ14msÿ58msÿ230msÿ
SMTEN	Port input Schmitt setting: Input SMT function is off; Input SMT function is turned on;

Smtvs	Port Schmitt threshold selection: 2.0V/0.8Vÿ 0.8VDD/0.2VDDÿ
IOHS	Port drive source current selection: Current limiting Normal drive;
MCUSEL	drive; Chip mode selection: Assembly mode; C compilation mode;
ENCR	Program code encryption settings: Program code encryption; The program code is not encrypted;

4 System clock

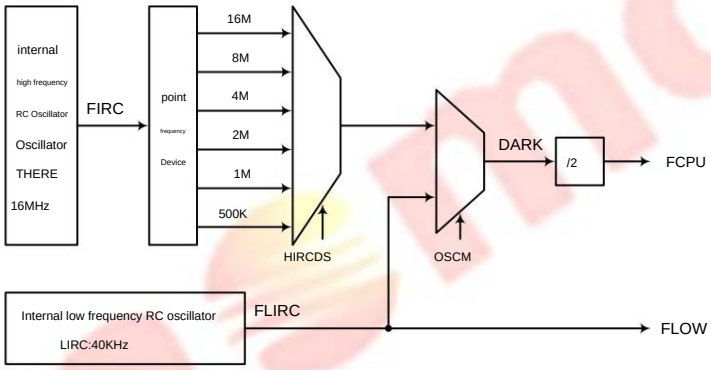
The chip is a dual-clock system. The internal circuits all work under the system main clock FOSC or the system low-frequency clock FLOSC . The clocks of some modules It is also possible to switch between FOSC and FLOSC .

The system main clock FOSC can select the internal high frequency RC oscillator HIRC (16MHz) clock FHIRC through the configuration word OSCM. 1/2/4/8/16/32 divided clock, or internal low frequency RC oscillator LIRC (40KHz) clock FLIRC; system low frequency clock FLOSC is fixed at Internal low frequency RC oscillator LIRC (40KHz) clock FLIRC.

The CPU uses a single clock, and the clock source is fixed to the system main clock FOSC. The CPU clock frequency FCPU is fixed to 2 of FOSC .

The clock source of the WDT (watchdog timer) circuit is fixed to the internal low frequency RC oscillator.

System clock diagram



4.1 Internal High Frequency RC Oscillator

The chip has a built-in high-precision HIRC oscillator with an oscillation frequency of 16MHz, which can then be divided into two output clocks (16MHz/8MHz/4MHz). /2MHz/1MHz/500KHz) can be used as the system main clock source.

4.2 Internal Low Frequency RC Oscillator

The chip has a built-in LIRC oscillator with a typical oscillation frequency of 40KHz, which can be used as the system main clock source or system low-frequency clock source. It is also used in system power-on delay control, WDT timer and other circuits.

Note: If the system main clock source is configured as HIRC, LIRC will work only when WDT is turned on, or when WDT is enabled and its clock is selected as FLIRC.

T0

4.3 System working mode

The chip supports two system working modes: run mode and sleep mode.

Working Mode	Entry conditions	System Status
Operation Mode	System Reset	CPU running, main clock source working
	CPU wakes up in sleep mode	
In sleep mode, execute the STOP instruction.		CPU is halted, main clock source is stopped

Note: If the system main clock is LIRC, then WDT is turned on, or after it is enabled the TO is always working.

4.4 Low Power Mode

The low power consumption mode of the chip is sleep mode.

Executing the STOP instruction can put the system into low power consumption mode, which will have the following effects on the system:

- ÿ The CPU stops running;
- ÿ Stop the oscillation of the corresponding clock source according to different modes;
- ÿ RAM contents remain unchanged;
- ÿ All input and output ports remain unchanged;
- ÿ If the timer's clock source is not stopped, it can continue to work;

The following situations can cause the system to exit low power mode:

- ÿ Power-on reset;
- ÿ External reset (if there is external reset function);
- ÿ There is a WDT overflow (if the WDT continues to work in low power mode);
- ÿ An external interrupt request occurs (if there is an external interrupt function);
- ÿ A timer overflow interrupt occurs (if the timer continues to work in low power mode);
- ÿ A keyboard interrupt request occurs (if there is a keyboard interrupt function);

Note:

1ÿ

When an interrupt request is triggered in low power mode, if the corresponding interrupt enable bit is turned off, the low power mode will not be exited; if the corresponding interrupt enable bit is turned on and the interrupt enable bit is turned off, only the CPU will be woken up to execute the next instruction; if the corresponding interrupt enable bit and the interrupt enable bit are both turned on, the CPU will be woken up.

The CPU then executes the interrupt service routine;

2ÿ Unused or unsealed pins should be connected to their corresponding I/O

The port is set to a stable state such as output, input pull-up or input pull-down to avoid leakage current or unexpected interrupt wake-up due to pin floating;

5 Reset

5.1 Reset Conditions

The chip has the following reset modes: ÿ Power-on

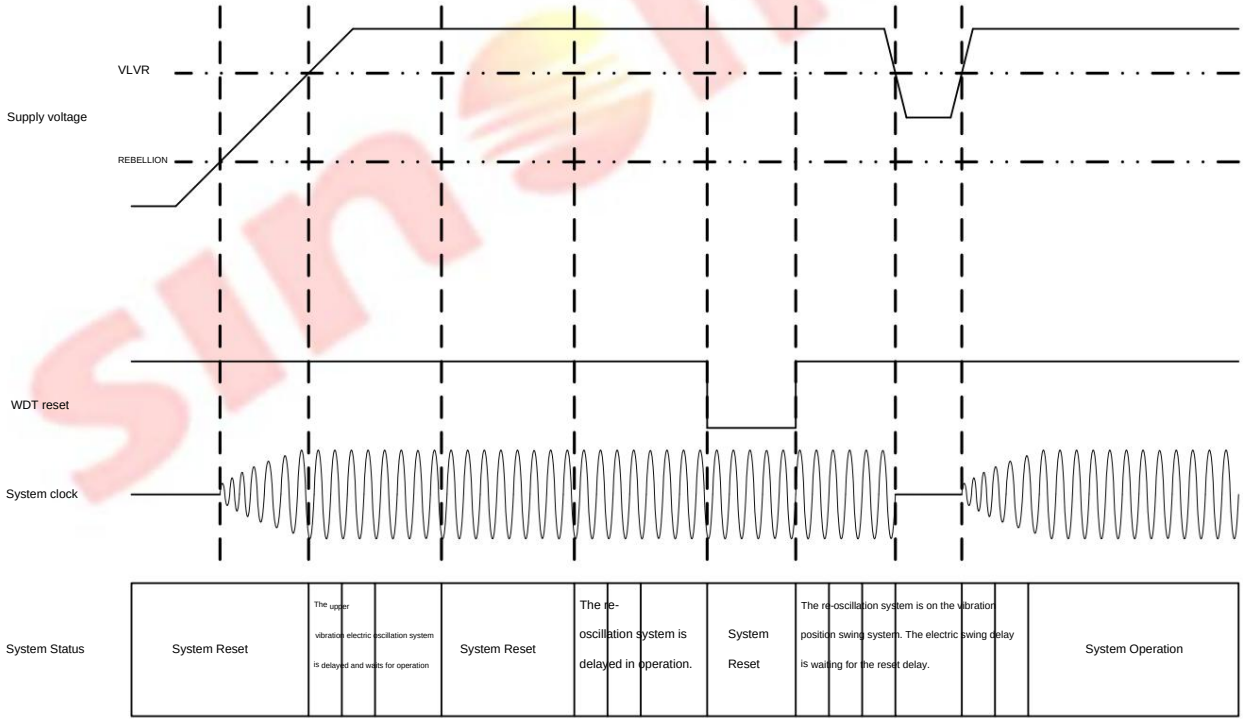
reset POR; ÿ Low voltage reset

LVR; ÿ WDT watchdog reset;

After any reset occurs, the system enters the reset state, performs initialization operations and resets SFR to the reset initial value; reset condition solution After the removal, the system exits the reset state and the CPU starts to execute instructions again from the address 0000H of the program memory.

The power-on reset POR and low voltage reset LVR will turn off the system main clock oscillator, and the oscillator will be turned on again after the reset is released. Because it takes a certain amount of time for the oscillator to start and stabilize, the system will maintain a certain power-on delay (typical value is about 13ms) and oscillation wait before starting to work; while the WDT reset will not turn off the main clock oscillator. When the reset is released, the system will start working after a shorter reset delay and oscillation wait.

The following figure is a schematic diagram of the timing relationship between reset generation and system operating status:



Note: If the VDD voltage of the chip rises slowly when the application system is powered on or recovers from power failure, a software delay should be performed after reset to ensure that the VDD is stable within the operating voltage range corresponding to the FCPU when the chip starts working.

5.2 Power-On Reset

The chip's power-on reset circuit can adapt to fast and slow power-on situations, and can also reset the chip when power supply voltage jitter occurs during the chip power-on process. Can ensure reliable reset of the system.

The power-on reset process can be summarized as follows: (1) Detect the system operating voltage and wait for the voltage to be higher than the power-on reset voltage VPOR and remain stable. (2) If there is an LVR function, wait for the voltage to be higher than VLVR and remain stable. (3) If there is an external reset function, wait for the reset pin voltage to be higher than Vih. (4) Initialize all registers. (5) Turn on the main clock oscillator and wait for a while for the oscillator to stabilize. (6) After power-on, the system starts to execute instructions.

5.3 Low Voltage Reset

The low voltage reset voltage of the chip can be selected through the configuration word LVRVS. The voltage detection circuit has a certain hysteresis characteristic, and the hysteresis voltage is about 6% (typical value). When the power supply voltage drops to the LVR voltage, the LVR reset is effective. Otherwise, the power supply voltage needs to rise to the LVR voltage + 6% before the LVR reset is released.

If the system main clock source is HIRC, LVR will automatically turn off when the system enters sleep mode and automatically turn on when entering run mode;
If the system main clock source is LIRC, LVR will always be disabled.

5.4 Watchdog reset

The watchdog (WDT) reset is a protection mechanism for the normal operation of the program. Under normal circumstances, the user program needs to regularly clear the WDT timer to ensure that the WDT does not overflow. If an abnormal situation occurs and the program does not clear the WDT timer on time, the chip will generate a watchdog reset due to WDT overflow, and the system will be reinitialized and return to the controlled state.

Note: In low power mode, the CPU is suspended. If a WDT overflow occurs at this time, it will only wake up the CPU without generating a reset.

6 I/O ports

6.1 General I/O Function

The chip's input/output ports are a set of 6-bit ports P1. All I/O ports support Schmitt input and push-pull output.

In addition to being used as general-purpose digital I/O ports, some ports also have multiplexing functions such as external interrupt input or PWM output.

Port Data Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1	-	-	P15D	P14D	P13D	P12D	P11D	P10D
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	-	-	X	X	X	X	X	X

BIT[5:0] **P1nD** – P1n port data bit (n=5-0)

Port Direction Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DDR1	-	-	DDR15	DDR14 DDR13 DDR12			DDR11	DDR10
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	-	-	1	1	1	1	1	1

BIT[5:0] **DDR1n** – P1n port direction control bit (n=5-0)

0: The port is used as an output port, and the port read operation will read the port's data register value;

1: The port is used as an input port, and the port read operation will read the input level status of the port;

6.2 Internal Pull-up/Pull-down Resistors

All ports have internal pull-up and pull-down resistors, and each port has a separate register bit to control its pull-up/pull-down resistor when the port is in the input state.

When the port is in output state, the pull-up/pull-down resistor and its control bit are invalid.

Pull-up resistor control register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PUCON	-	-	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	-	-	1	1	1	1	1	1

BIT[5:0] **P1nPU** – P1n port pull-up resistor control bit (n=5-0)

0: The internal pull-up resistor of the port is valid;

1: The internal pull-up resistor of the port is invalid;

Pull-down resistor control register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDCON			P15PD	P14PD	P13PD	P12PD	P11PD	P10PD
R/W			R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			1	1	1	1	1	1

BIT[5:0] **P1nPD** – P1n port pull-down resistor control bit (n=5-0)

0: The internal pull-down resistor of the port is valid;

1: The internal pull-down resistor of the port is invalid;

6.3 Port Mode Control

When used as digital output ports, except for P13 which is fixed as open-drain output, the other ports can select push-pull output or open-drain output.

When the port is in pull output or open drain output low, the input path remains connected; when the port is in open drain output high, the input path will be automatically shut down.

Port Output Mode Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ODCON			P15OD	P14OD	Keep	P12OD	P11OD	P10OD
R/W			R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	0	0	0	0

BIT[5:4,2:0] **P1nOD** – P1n port output mode selection bit (n=5-4,2-0)

0: When outputting, the port is push-pull output;

1: The port is open drain output when output;

BIT[3] Reserved bit, must be set to "1"

7 Timer

7.1 Watchdog Timer WDT

The clock source of the watchdog timer WDT is an internal low frequency RC oscillator, and different counting clock frequencies can be selected through the prescaler.
WDT counter overflow will reset the chip or wake up the CPU.

Whether to turn on WDT can be determined by configuring word WDTM and register bit WDTEN.
When WDTEN is 0, the WDT timer is disabled; when WDTM is selected to be always on and WDTEN is 1, the WDT timer is enabled.
If the WDT timer is enabled, the WDT will still work in sleep mode and wake up the CPU when it overflows.
If it is output, the chip will be reset.

WDT and timer T0 share a prescaler, and the prescaler allocation is determined by the register bit.
When the prescaler is assigned to WDT, the T0 clock is not divided.

Executing the CLRWDT instruction or the STOP instruction will clear the WDT counter and, if the prescaler is assigned to WDT, will also clear the prescaler.
Frequency counter (prescaler ratio remains unchanged).

The basic overflow time of WDT (i.e. the time without pre-scaling) can be configured as 3.6ms/14ms/58ms/230ms.

Note: The WDT overflow time is a typical value. The actual value may have large deviations. It is necessary to ensure that the WDT clearing time is less than 1/4 of the typical value.

Miscellaneous Control Registers

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MCR WDTEN		OUR	LVDF	LVDVS3	LVDVS2	LVDVS1	LVDVS0	LVDEN
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Initial Value	1	0	0	0	0	0	0	0

BIT[7] **WDTEN** – Watchdog Enable Bit
0: turn off WDT;
1: Turn on WDT;

BIT[6] **EIS** – INT interrupt external input enable bit
0: INT interrupt external input is invalid, the port is used for other functions;
1: INT interrupt external input is valid, the port needs to be set as input;

BIT[5] **LVDF** – LVD detection status flag

BIT[4:1] **LVDVS[3:0]** – LVD voltage detection value selection bit

BIT[0] **LVDEN** – Low Voltage Detector LVD Enable Bit

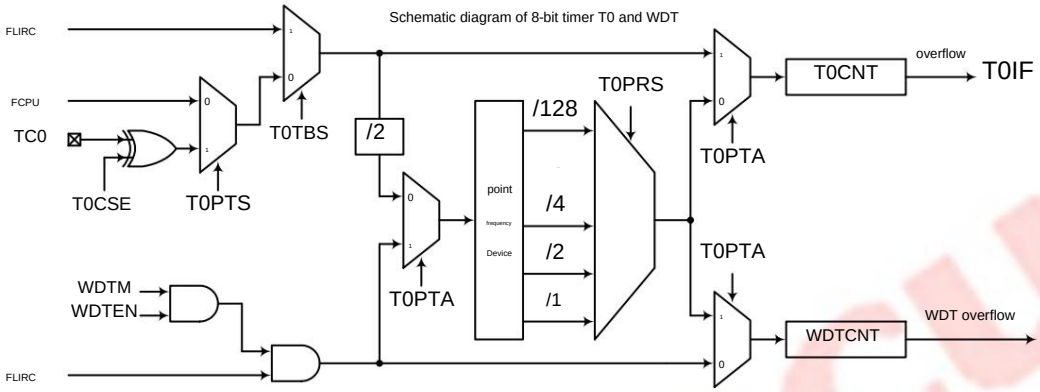
7.2 Timer T0

Timer T0 is an 8-bit timer/counter, which includes an 8-bit up-counter, a programmable prescaler, and a control register.

ÿ The counting frequency can be set by the pre-scaling ratio;

ÿ Optional clock source: FLIRC, FCPU, external clock (TC0 input);

ÿ Support overflow interrupt and overflow wake-up functions;



T0CNT is an 8-bit readable and writable up counter. When the count overflows to 0, an overflow signal is generated and an interrupt is triggered. The interrupt flag T0IF is set to Set to 1.

The prescaler is shared by T0 and WDT. The prescaler allocation is controlled by register bit T0PTA.

When T0PTA=0, the prescaler is assigned to T0. T0 counting cycle = prescaler ratio / T0 counting clock frequency. Write T0CNT The prescaler counter will be cleared, while executing CLRWDT or STOP instruction will not affect the prescaler count.

When T0PTA=1, the prescaler is assigned to WDT. Executing CLRWDT or STOP instruction will clear the prescaler counter. Writing T0CNT also does not affect the prescaler count.

Changing the prescaler assignment via T0PTA will also clear the prescaler counter.

The action of clearing the prescaler counter does not change the prescaler ratio, and changing the prescaler ratio does not clear the prescaler counter.

When FLIRC is selected as T0 clock via T0TBS , T0 will continue to work in low power mode and can be woken up by overflow.

Timer T0 Control Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0CR	-	INTM	T0PTS	T0CSE	T0PTA	T0PRS2	T0PRS1	T0PRS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	-	0	1	1	1	1	1	1

BIT[6] **INTM** – External interrupt INT trigger mode selection bit

0: falling edge trigger;

1: rising edge trigger;

BIT[5] **T0PTS** – T0 clock source selection bit

0: T0 clock source is FCPU;

1: T0 clock source is the external clock input by TC0 (the port needs to be set to input state);

BIT[4] **T0CSE** – T0 external clock counting edge selection bit

0: External clock rising edge counting;

1: External clock falling edge counting;

BIT[3] **T0PTA** – Prescaler allocation control bit

0: prescaler is assigned to T0;

1: prescaler is assigned to WDT;

BIT[2:0] **T0PRS[2:0]** – Prescaler selection bits

T0PRS[2:0]	T0 clock prescaler ratio ÿT0PTA=0ÿ	WDT clock prescaler ratio ÿT0PTA=1ÿ
000	1ÿ2	1ÿ1
001	1ÿ4	1ÿ2
010	1ÿ8	1ÿ4
011	1ÿ16	1ÿ8
100	1ÿ32	1ÿ16
101	1ÿ64	1ÿ32
110	1ÿ128	1ÿ64
111	1ÿ256	1ÿ128

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMRCR		T0TBS	·	·	·	·	T1IE	T1IF
R/W		R/W	·	·	·	·	R/W	R/W
Initial Value		0	·	·	·	·	0	0

BIT[6] **T0TBS** – T0 clock source selection bit

0: T0 clock source is determined by T0PTS;

1: T0 clock source is the system low-frequency clock FLIRC;

BIT[1] **T1IE** – Timer T1 interrupt enable bit

0: Mask timer T1 interrupt;

1: Enable timer T1 interrupt;

BIT[0] **T1IF** – Timer T1 interrupt flag

0: No timer T1 interrupt occurs;

1: Timer T1 interrupt occurs, and needs to be cleared to 0 by software;

Timer T0 Counter

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0CNT	T0CNT7	T0CNT6	T0CNT5	T0CNT4	T0CNT3	T0CNT2	T0CNT1	T0CNT0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	X	X	X	X	X	X	X	X

BIT[7:0] **T0CNT[7:0]** – T0 counter, a readable and writable up counter

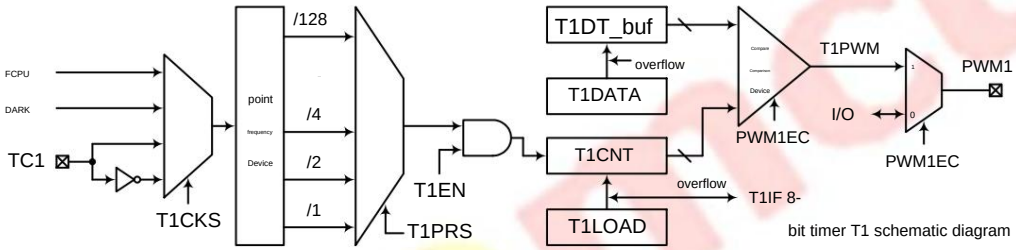
7.3 Timer T1

Timer T1 is an 8-bit timer/counter, which includes an 8-bit down counter, a programmable prescaler, a control register, an 8-bit rescaler, and a Load register and compare register.

ÿ The counting frequency can be set by the pre-scaling ratio, and the counting cycle can be controlled by the reload register;

ÿ Support 8-bit PWM output, and the PWM duty cycle can be set through the comparison register;

ÿ Support overflow interrupt and overflow wake-up functions;



bit timer T1 schematic diagram

Timer T1 can select the clock source through register bit T1CKS and the prescaler ratio through T1PRS. The pre-scaling ratio can be selected from 1 to 128, which is used to divide the T1CNT.

A write operation will clear the prescaler counter, while the prescaler ratio remains unchanged.

When T1EN=0, T1CNT remains unchanged, and writing the reload register T1LOAD will immediately load into T1CNT; when T1EN=1, T1CNT counts down, and after the clock count reaches 0, an overflow signal is generated and an interrupt is triggered. The interrupt flag T1IF will be set to 1, and T1 Automatically load the current T1LOAD value into T1CNT and restart counting.

As shown in the figure, timer T1 can realize PWM function (PWM1), and can enable/disable PWM function or control Whether the port outputs PWM waveform. When PWM1 is turned off, the T1PWM signal is low level. After PWM1 is enabled, T1CNT starts from the reload value. The count starts to decrease until the count overflows to become a PWM cycle: When the count is equal to the comparison register T1DATA, T1PWM becomes high level; when the count overflows, T1PWM becomes a low level.

T1DATA is equipped with an 8-bit comparison buffer (T1DT_buf) for comparison with T1CNT. When PWM1 is turned off, write T1DATA If T1DATA is written after PWM1 is enabled, it will be loaded into the compare buffer immediately.

To ensure that the PWM period and duty cycle are accurate, you need to first write the reload register and compare register, then enable PWM, and finally start the timer.

The duty cycle of the T1PWM signal is calculated as follows:

ÿ High level time = (T1DATA) × T1CNT count clock cycle

ÿ Cycle (T1 overflow cycle) = (T1LOAD+1) × T1CNT count clock cycle

ÿ Duty cycle = high level time / period = (T1DATA) / (T1LOAD+1)

Timer T1 Control Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CR	T1EN	PWM1EC		T1CKS1	T1CKS0	T1PRS2	T1PRS1	T1PRS0
R/W	R/W	R/W		R/W	R/W	R/W	R/W	R/W
Initial Value	0	0		0	0	0	0	0

BIT[7] **T1EN** – Timer T1 enable bit

0: turn off timer T1;

1: Start timer T1;

BIT[6] **PWM1EC** – PWM1 enable bit and port output control bit

0: Disable the PWM1 function and prohibit the port from outputting PWM waveform;

1: Enable PWM1 function and allow the port to output PWM waveform;

BIT[4:3] **T1CKS[1:0]** – T1 clock source selection bits

T1CKS[1:0]	T1 clock source
00	FCPU
01	DARK
10	TC1 rising edge
11	TC1 falling edge

Note: When the system main clock source is HIRC, and the T1 clock source is FOSC, the 16MHz **FHIRC will be used directly** instead of **the HIRC post-divided clock.**

BIT[2:0] **T1PRS[2:0]** – T1 prescaler selection bits

T1PRS[2:0]	T1 clock prescaler ratio
000	1ÿ1
001	1ÿ2
010	1ÿ4
011	1ÿ8
100	1ÿ16
101	1ÿ32
110	1ÿ64
111	1ÿ128

Timer T1 Counter

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CNT	T1CNT7	T1CNT6	T1CNT5	T1CNT4	T1CNT3	T1CNT2	T1CNT1	T1CNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	1	1	1	1	1	1	1	1

BIT[7:0] **T1CNT[7:0]** – T1 counter, a readable and writable down counter

Timer T1 reload register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1LOAD	T1LOAD7	T1LOAD6	T1LOAD5	T1LOAD4	T1LOAD3	T1LOAD2	T1LOAD1	T1LOAD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	1	1	1	1	1	1	1	1

BIT[7:0] **T1LOAD[7:0]** – T1 reload register, used to set the T1 counting period

Note: The value of the timer reload register cannot be 0, otherwise the timer will not work properly.

Timer T1 Compare Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1DATA	T1DATA7	T1DATA6	T1DATA5	T1DATA4	T1DATA3	T1DATA2	T1DATA1	T1DATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

BIT[7:0] **T1DATA[7:0]** – T1 compare register, used to set the duty cycle of PWM1

8 Low voltage detection LVD

The chip has a built-in low voltage detection module LVD, which can be turned on through the register bit LVDEN and the voltage detection value can be selected through LVDVS. When the VDD voltage is lower than the voltage detection value, the detection status flag LVDF will be set to 1; because the LVD circuit has a hysteresis characteristic (hysteresis voltage The typical value is 6%). LVDF is cleared only after the VDD voltage rises to the voltage detection value + 6%.

Note: When turning on LVD or switching voltage detection values, the LVD output will be valid only after the circuit is stable (time>200us).

Miscellaneous Control Registers

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MCR WDTEN		OUR	LVDF	LVDVS3 LVDVS2		LVDVS1	LVDVS0 LVDEN	
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Initial Value	1	0	0	0	0	0	0	0

BIT[7] **WDTEN** – Watchdog Enable Bit

BIT[6] **EIS** – INT interrupt external input enable bit

BIT[5] **LVDF** – LVD detection status flag

0: VDD voltage is higher than the voltage detection value, or LVD is turned off;

1: VDD voltage is lower than the voltage detection value;

BIT[4:1] **LVDVS[3:0]** – LVD voltage detection value selection bit

LVDVS[3:0]	LVD voltage detection value
0000	1.8V
0001	2.0V
0010	2.1V
0011	2.2V
0100	2.4V
0101	2.5V
0110	2.6V
0111	2.7V
1000	2.8V
1001	2.9V
1010	3.0V
1011	3.2V
1100	3.3V
1101	3.6V
1110	4.0V
1111	4.2V

BIT[0] **LVDEN** – Low Voltage Detector LVD Enable Bit

0: turn off LVD;

1: Turn on LVD;

9 Interruption

The interrupt sources of the chip include external interrupt (INT), timer interrupt (T0~T1) and keyboard interrupt.

GIE masks all interrupts.

The process of CPU responding to interrupt is as follows:

ÿ When the CPU responds to an interrupt request triggered by an interrupt source, it automatically pushes the address of the next instruction to be executed after the current instruction onto the stack.

Automatically clears the interrupt enable bit GIE to 0 to stop responding to subsequent interrupts. Unlike reset, hardware interrupts do not stop the current instruction.

Instead of executing the current instruction, the interrupt is temporarily suspended until the current instruction is completed.

ÿ After the CPU responds to the interrupt, the program jumps to the interrupt entry address (0008H) and starts executing the interrupt service routine.

The sequence should first save the accumulator A and the status register STATUS, and then handle the triggered interrupt.

ÿ After the interrupt service routine handles the interrupt, it should first restore the accumulator A and the status register STATUS, and then execute RETIE

Return to the main program. At this time, the chip will automatically restore GIE to 1, and then take out the PC value from the stack, from the current value when the interrupt occurs.

Execution continues with the next instruction.

Note: To use the external interrupt function or keyboard interrupt function, the corresponding port needs to be set to input status.

9.1 External Interrupts

The chip has one external interrupt source INT. Through INTM, you can select the rising edge or falling edge trigger mode.

The interrupt flag INTIF will be set to 1. If the general interrupt enable bit GIE is 1 and the external interrupt enable bit INTIE is 1, an external interrupt will be generated.

Note: When the P10 port is multiplexed as INT through register EIS control bits, the keyboard interrupt wake-up function of the port is invalid.

9.2 Timer Interrupt

When the timer Tn (n=0-1) overflows, the timer interrupt is triggered and the interrupt flag TnIF (n=0-1) will be set to 1.

If the enable bit GIE is 1 and the timer interrupt enable bit TnIE (n=0-1) is 1, a timer interrupt is generated.

9.3 Keyboard Interrupt

The chip has 6 keyboard interrupt sources, which can be enabled or masked individually through register bits. The input level of any enabled interrupt source

When a change occurs, a keyboard interrupt will be triggered and the interrupt flag KBIF will be set to 1. If the interrupt enable bit GIE is 1 and the keyboard interrupt enable bit

If KBIE is 1, a keyboard interrupt is generated.

Keyboard interrupt control register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1KBCR			P15KE	P14KE	P13K	P12K	P11KE	P10KE
R/W			R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	0	0	0	0

BIT[5:0] **P1Nke** – P1n port keyboard interrupt enable bit (n=5-0)

0: Shield the port keyboard interrupt function;

1: Enable port keyboard interrupt function;

9.4 Interrupt related registers

Interrupt Enable Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
NOT	GIE					INTIE	KBIE	T0IE
R/W	R/W					R/W	R/W	R/W
Initial Value	0					0	0	0

BIT[7] **GIE** – Global Interrupt Enable Bit

0: Mask all interrupts;

1: The corresponding interrupt enable bit determines whether the CPU responds to the interrupt triggered by the interrupt source;

BIT[2] **INTIE** – INT interrupt enable bit

0: Mask INT interrupt;

1: Enable INT interrupt;

BIT[1] **KBIE** – Keyboard Interrupt Enable Bit

0: shield keyboard interrupt;

1: Enable keyboard interrupt;

BIT[0] **T0IE** – Timer T0 interrupt enable bit

0: Mask timer T0 interrupt;

1: Enable timer T0 interrupt;

Interrupt Flag Register

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTF						INTIF	KBIF	T0IF
R/W						R/W	R/W	R/W
Initial Value						0	0	0

BIT[2] **INTIF** – INT interrupt flag

0: INT interrupt is not triggered;

1: INT interrupt has been triggered and needs to be cleared by software;

BIT[1] **KBIF** – Keyboard interrupt flag

0: keyboard interrupt is not triggered;

1: The keyboard interrupt has been triggered and needs to be cleared by software;

BIT[0] **T0IF** – Timer T0 interrupt flag

0: Timer T0 interrupt is not triggered;

1: Timer T0 interrupt has been triggered and needs to be cleared by software;

10 Characteristic curves

Note:

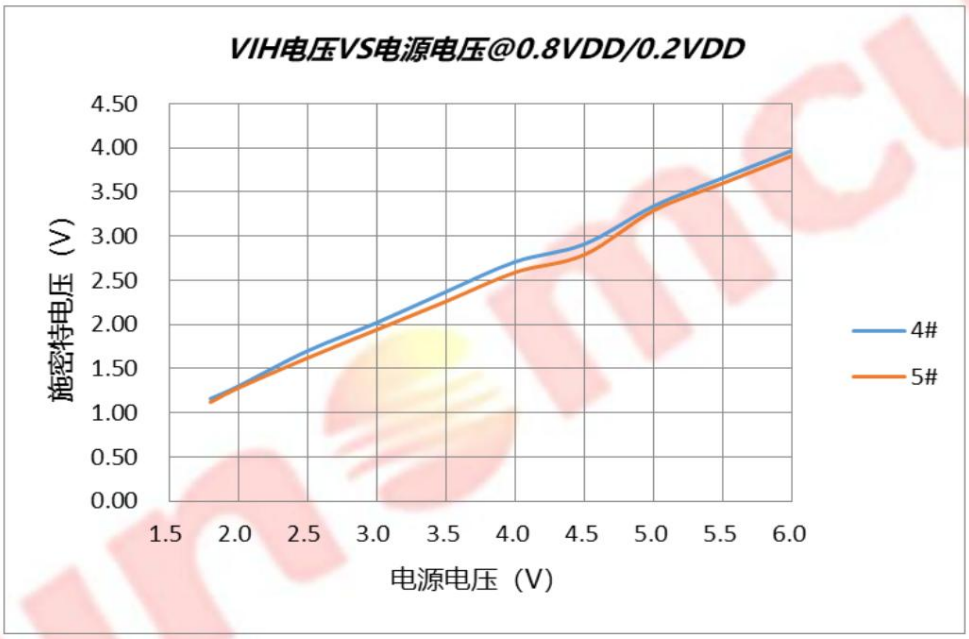
1ÿ

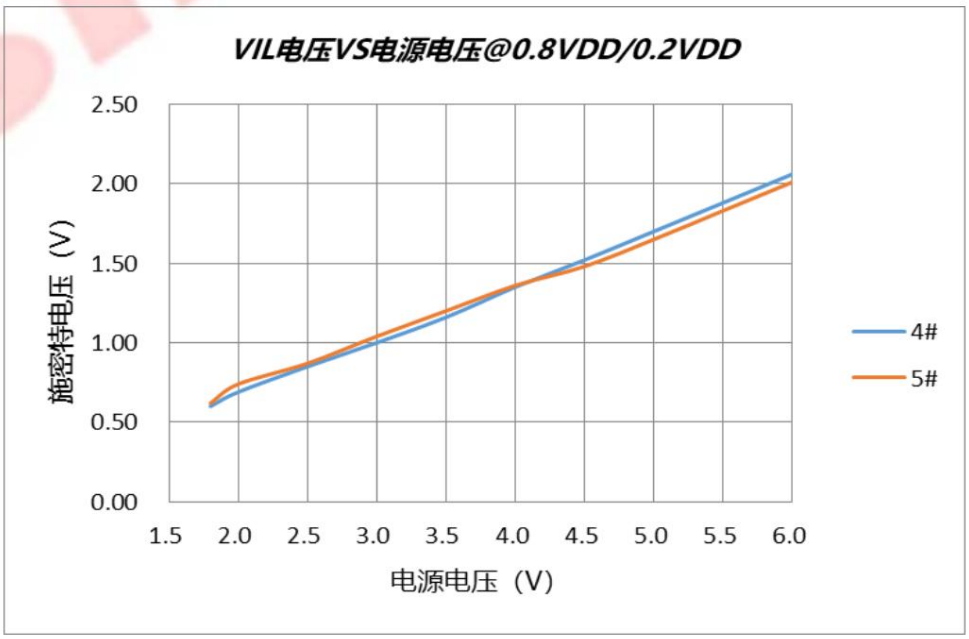
The data in the characteristic curves are all from sampling measurements and are only used as a reference for application. Some data may not match the actual chip due to production process deviations. To ensure that the chip can work properly, please ensure that its working conditions meet the electrical characteristic parameter description;

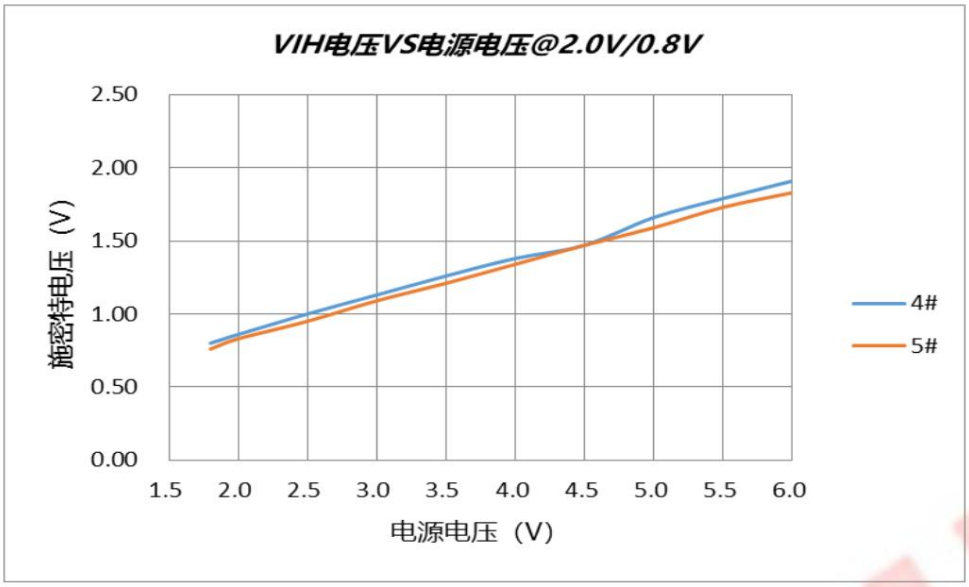
2ÿ **If there is no special description in the figure and text, the temperature condition of the voltage characteristic curve is T=25ÿ, and the voltage condition of the temperature characteristic curve is VDD=5V;**

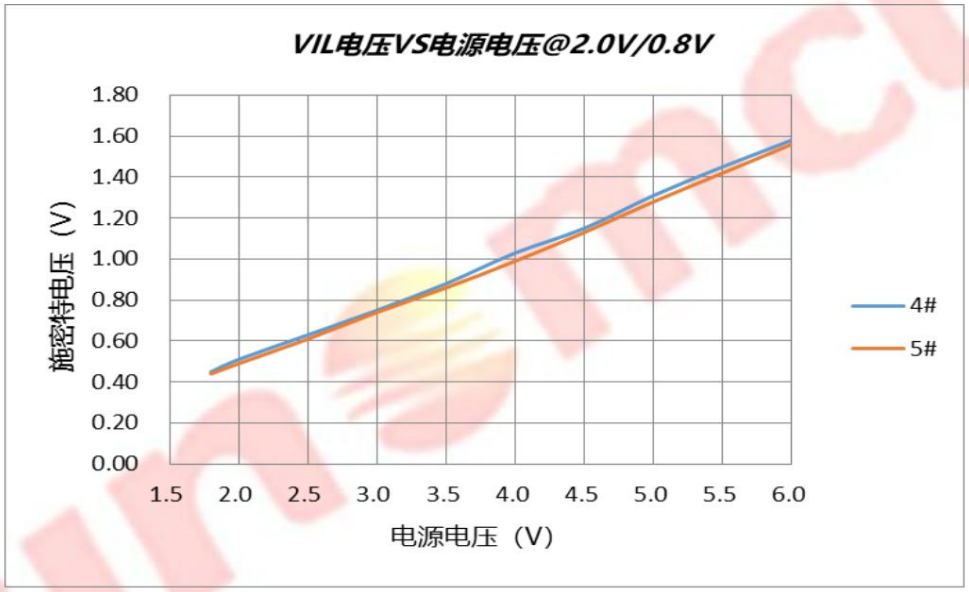
10.1 I/O Characteristics

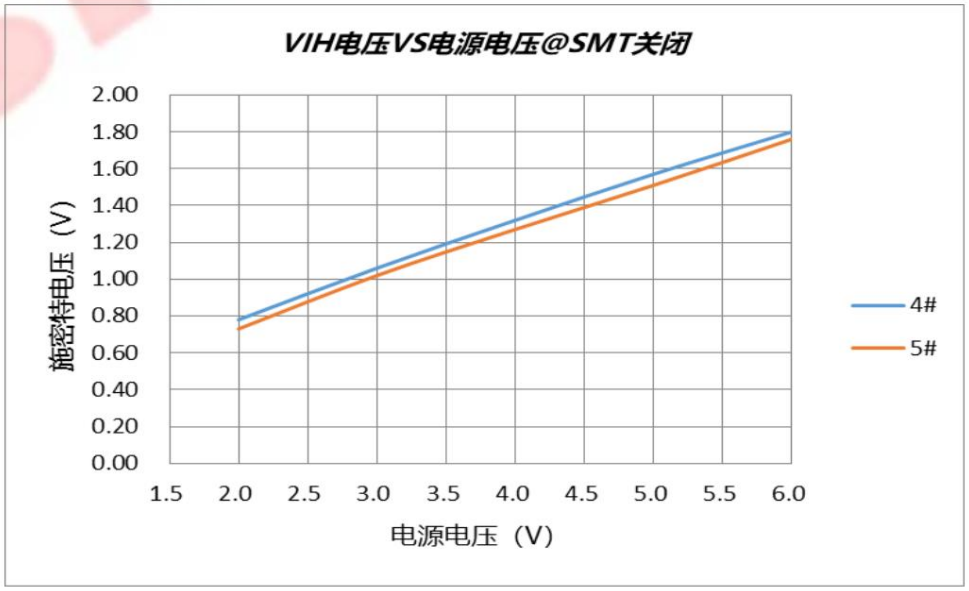
Input SMT Threshold Voltage VS Supply Voltage

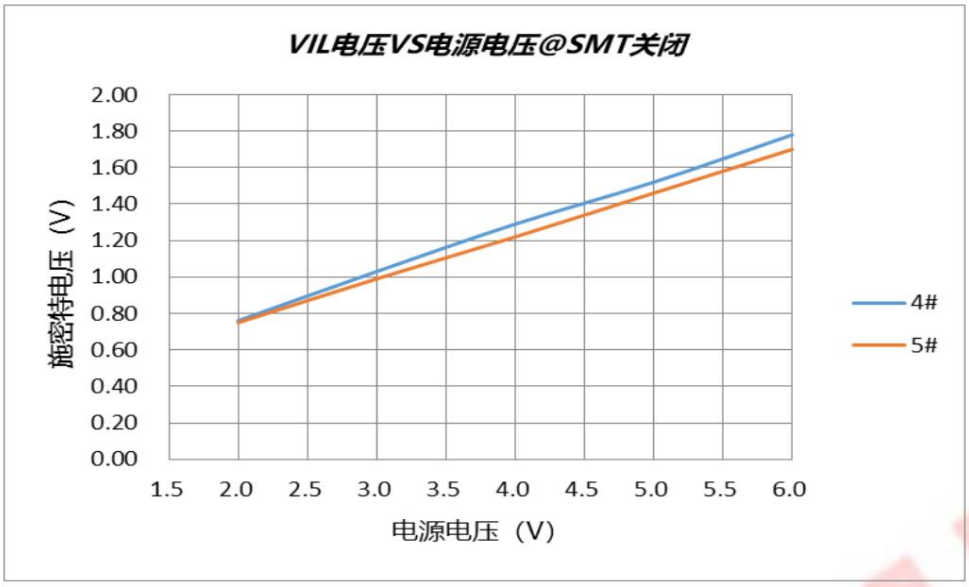




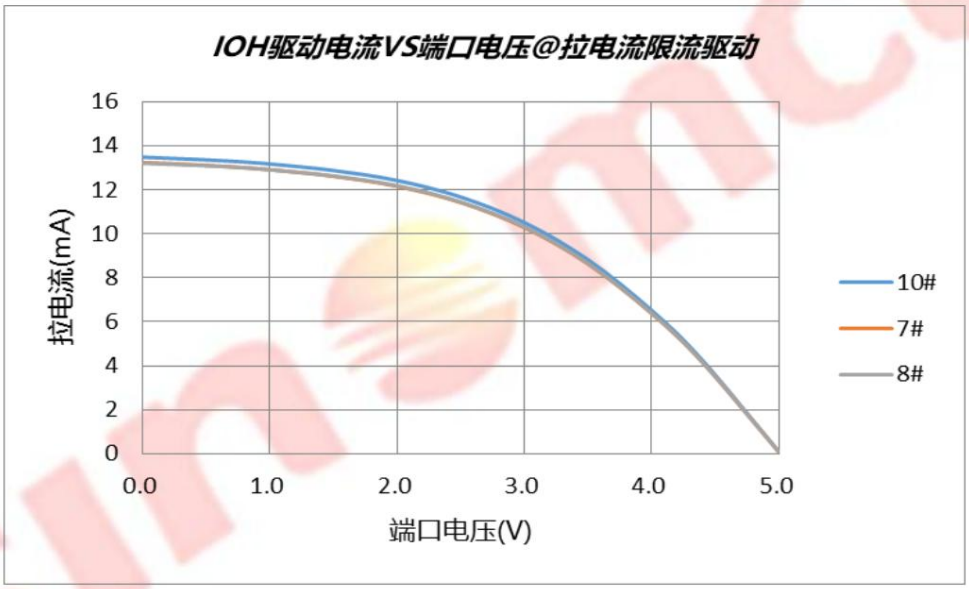


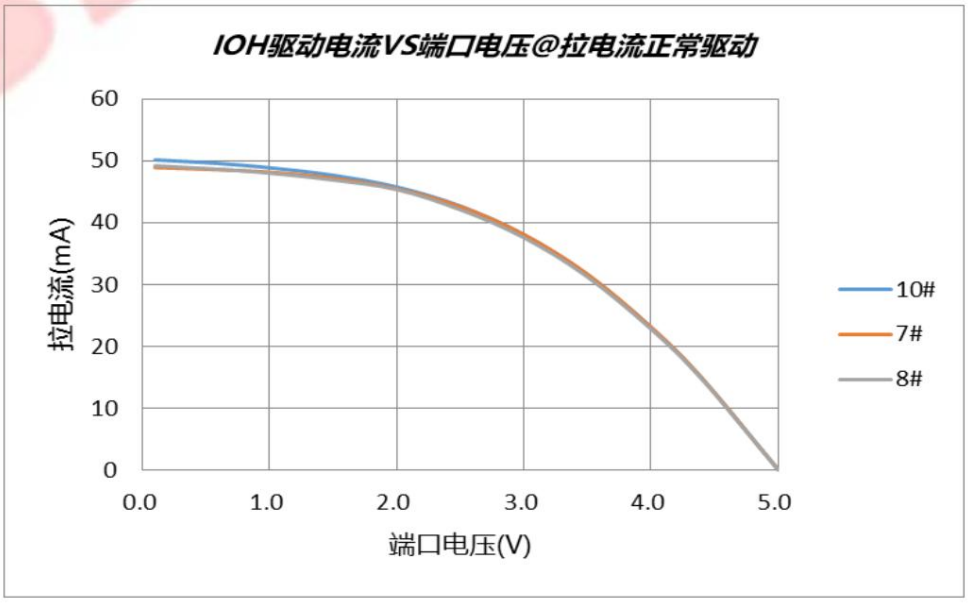


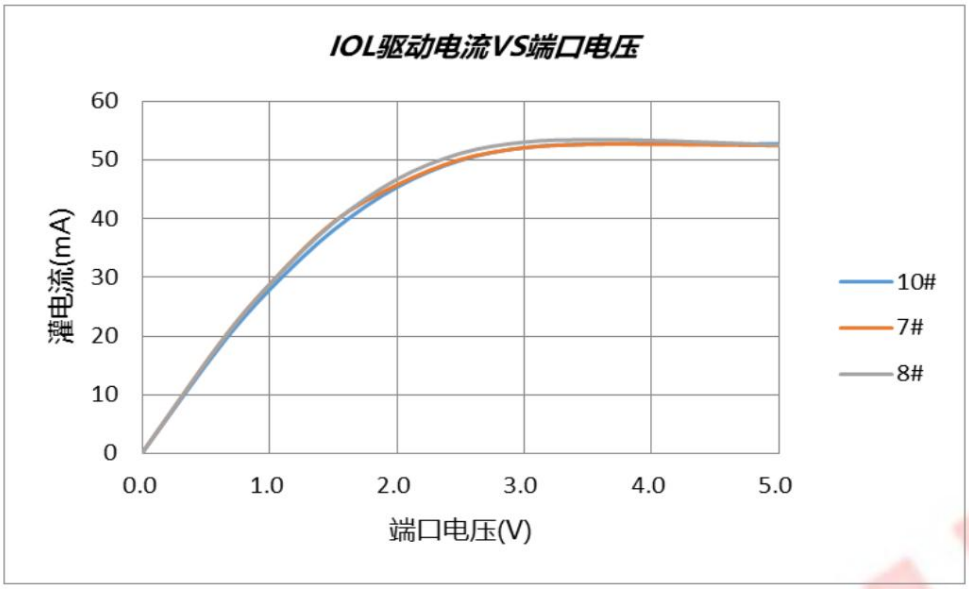




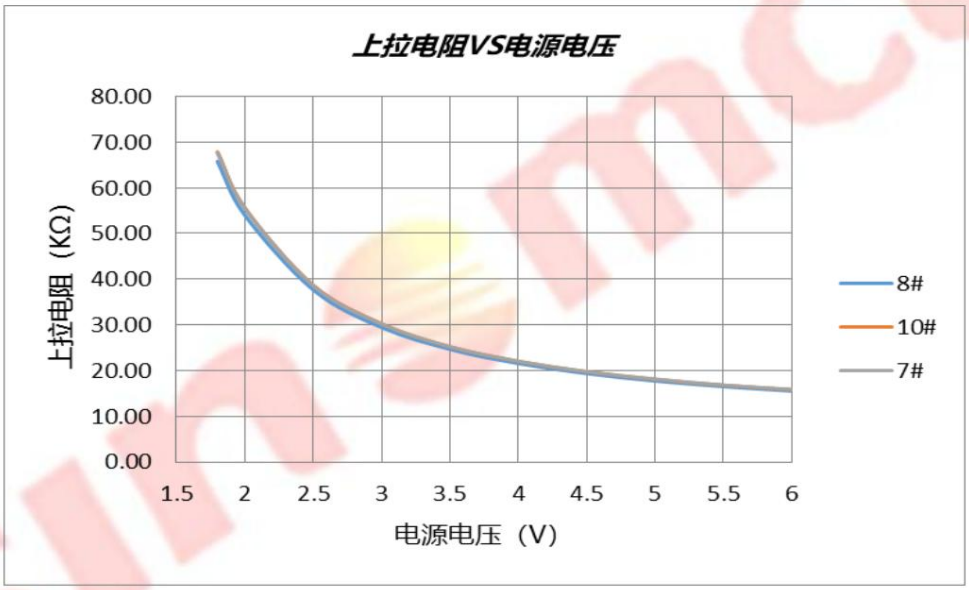
I/O output drive current VS port voltage (VDD=5V)

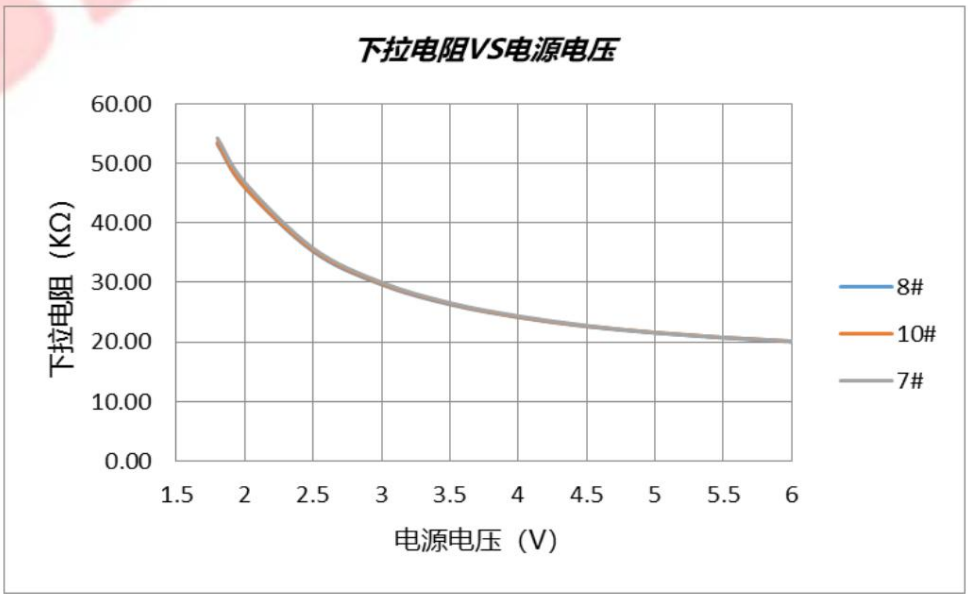






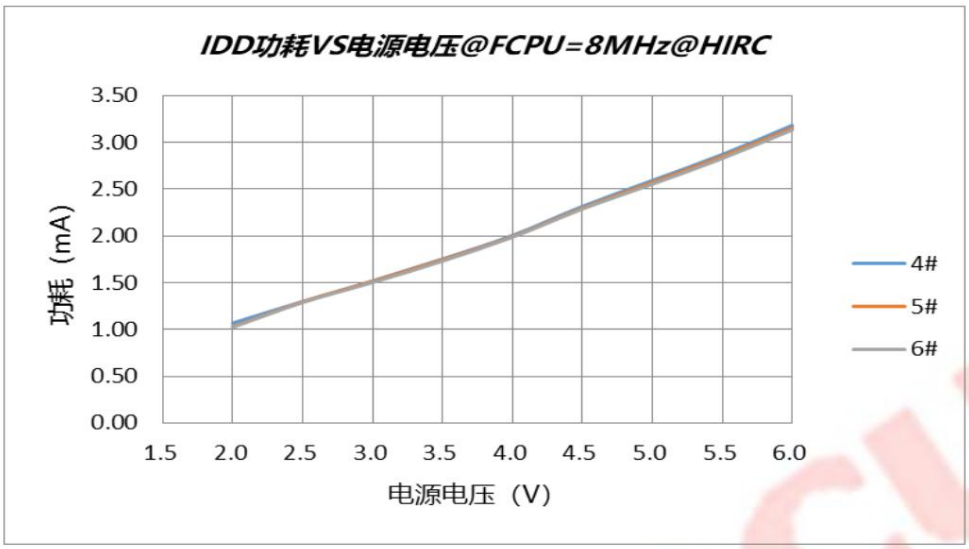
Pull-up/pull-down resistor value vs supply voltage

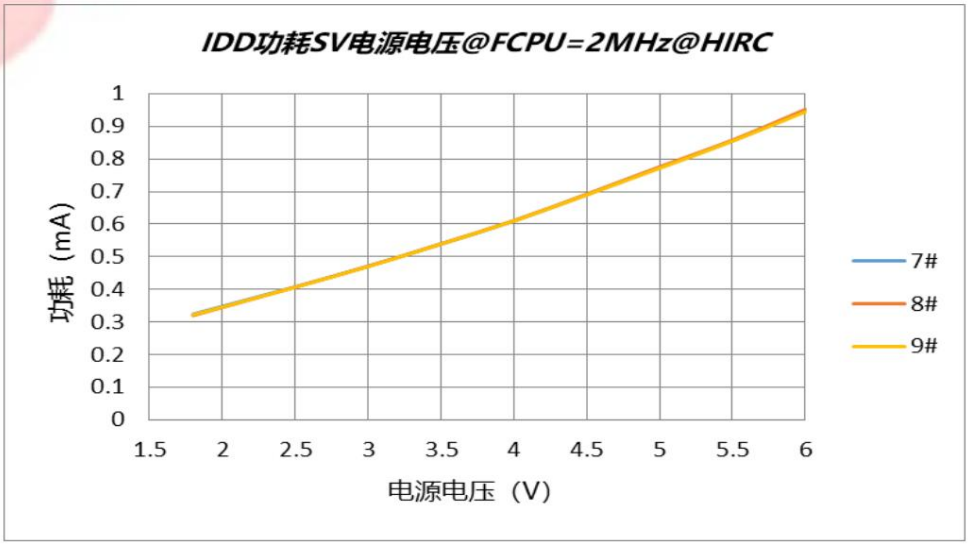


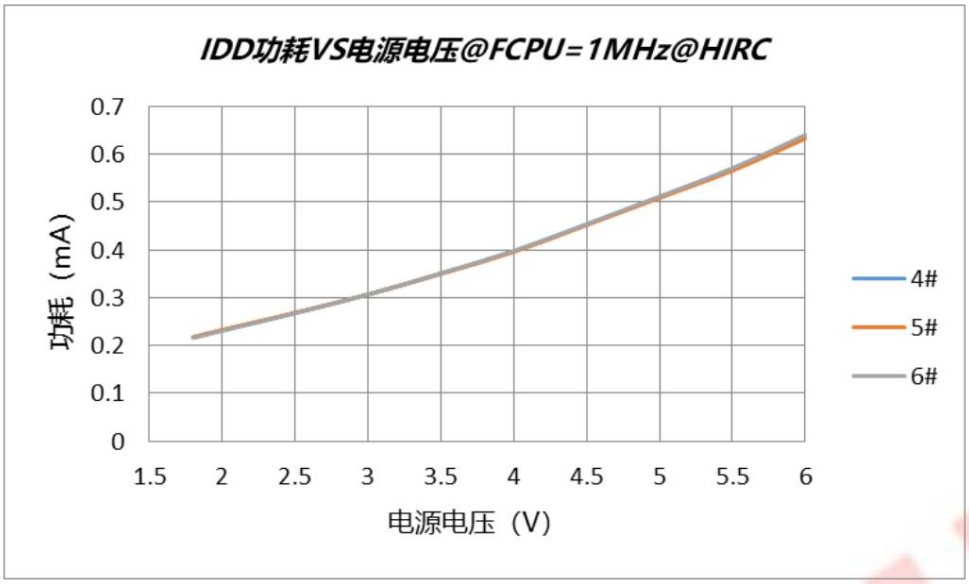


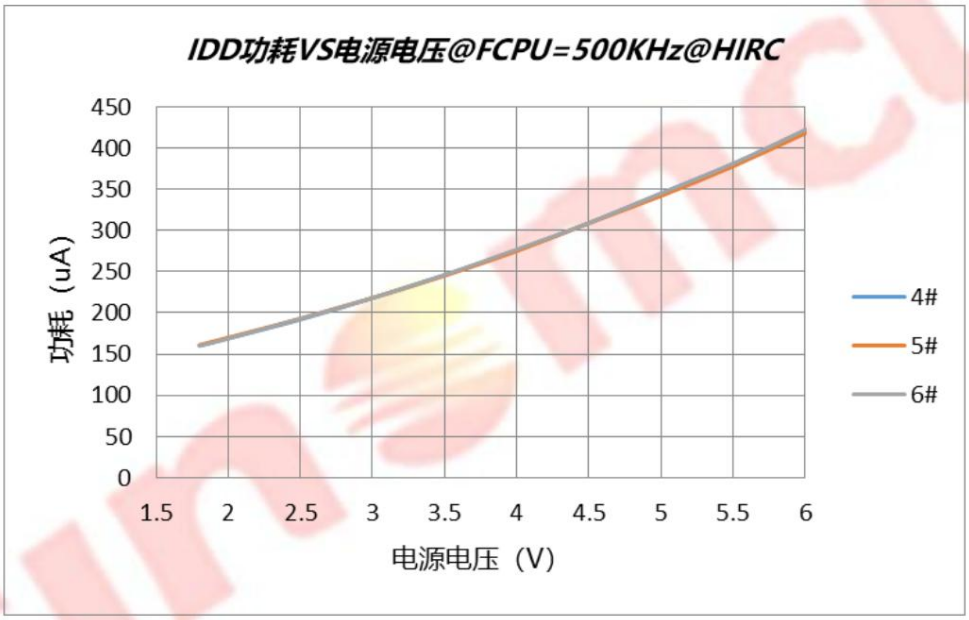
10.2 Power consumption characteristics

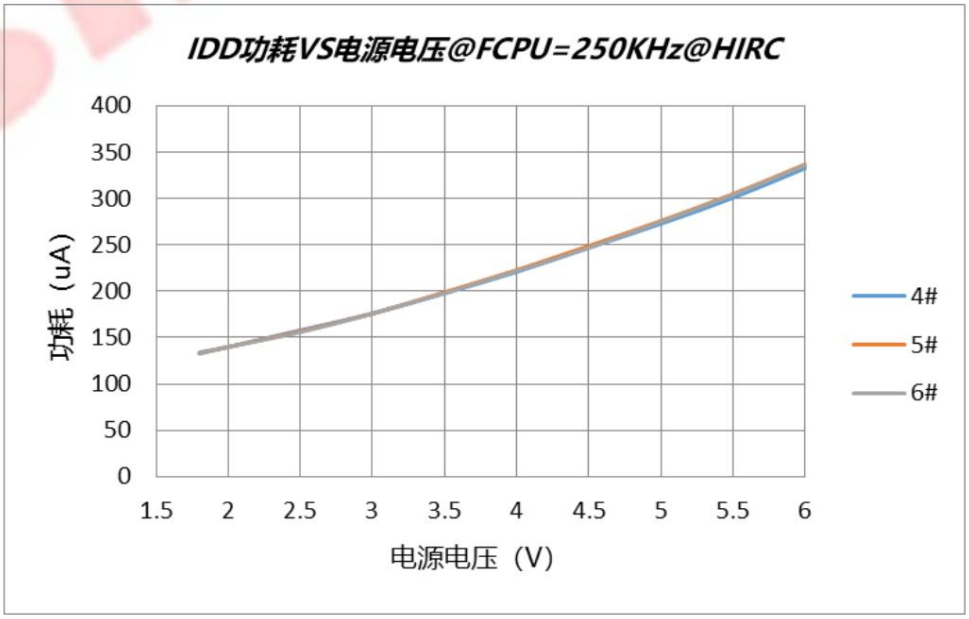
Run Mode Power Consumption vs Supply Voltage

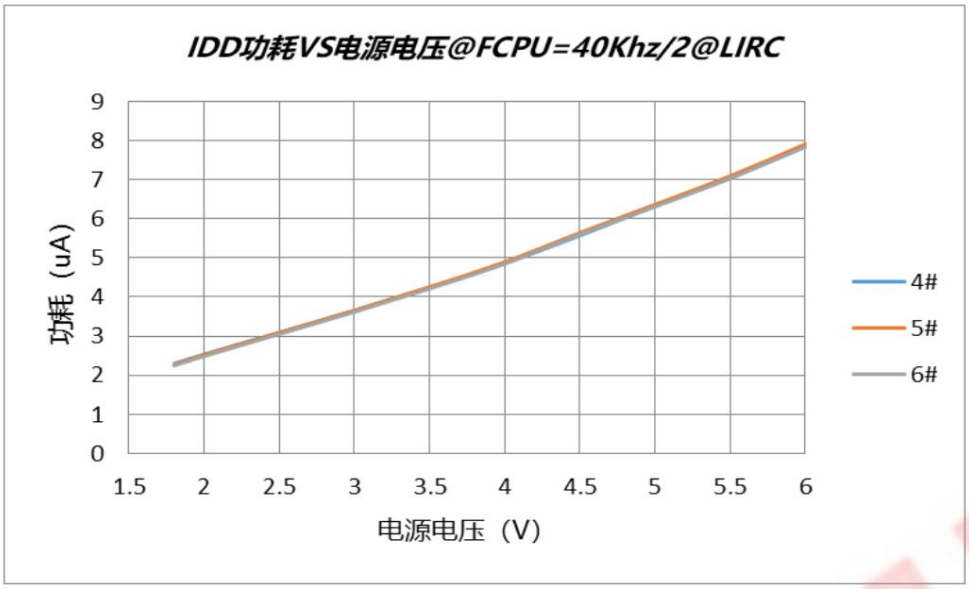




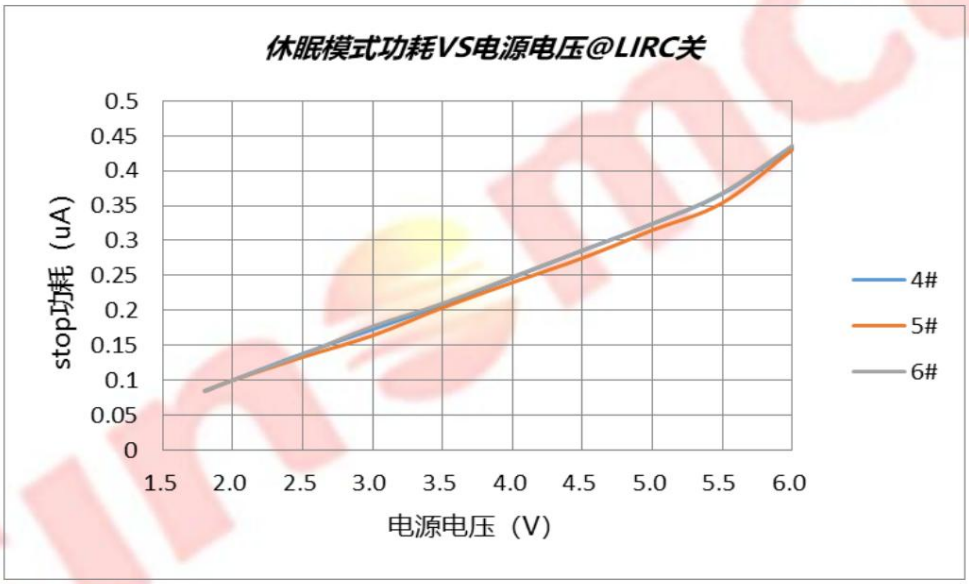


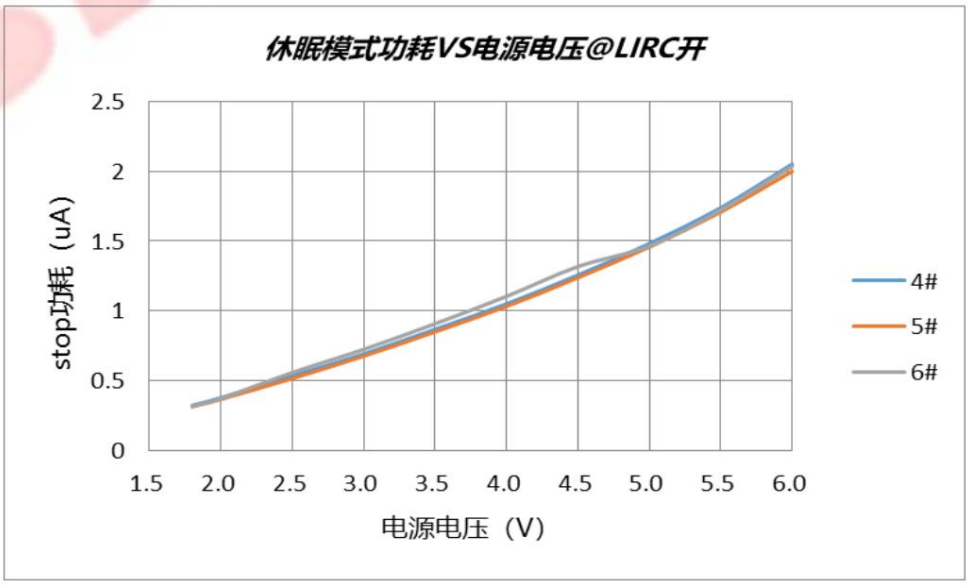






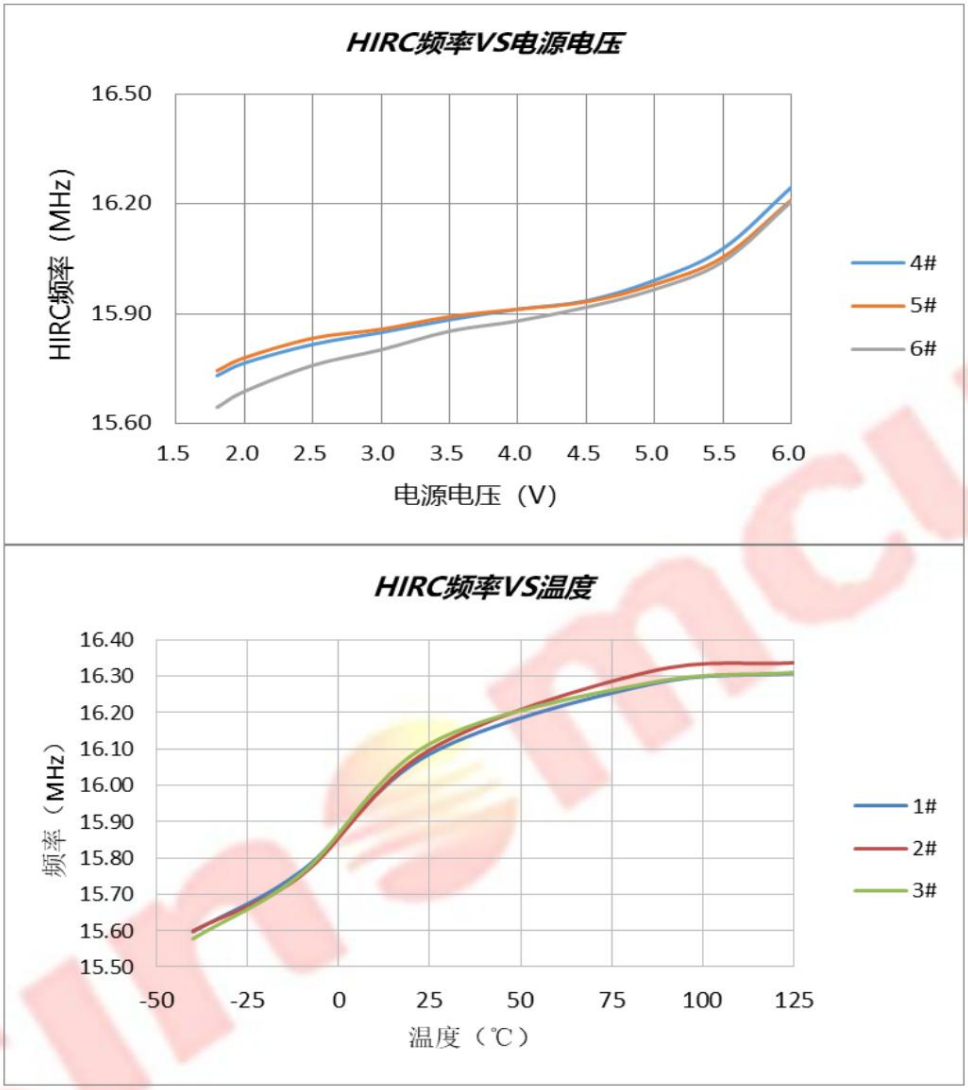
Sleep Mode Power Consumption vs Supply Voltage



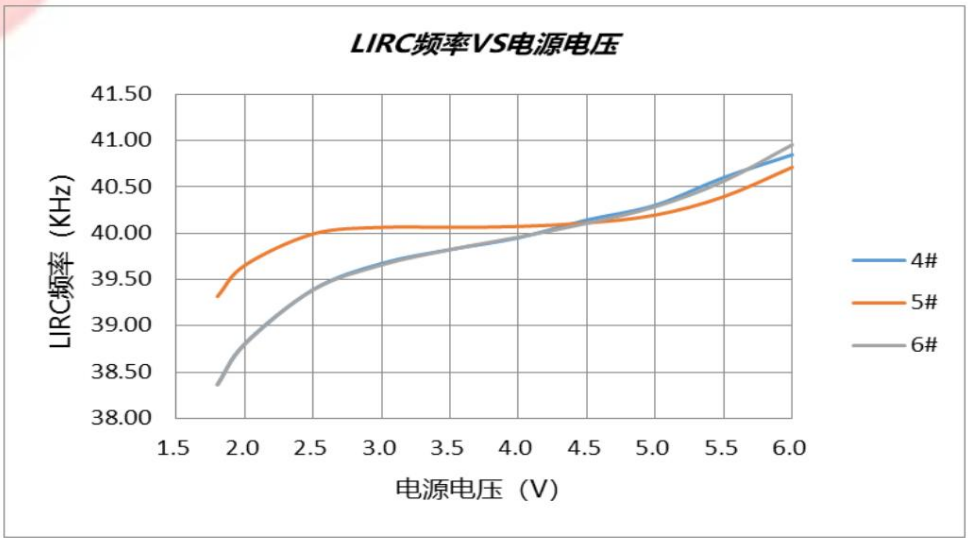


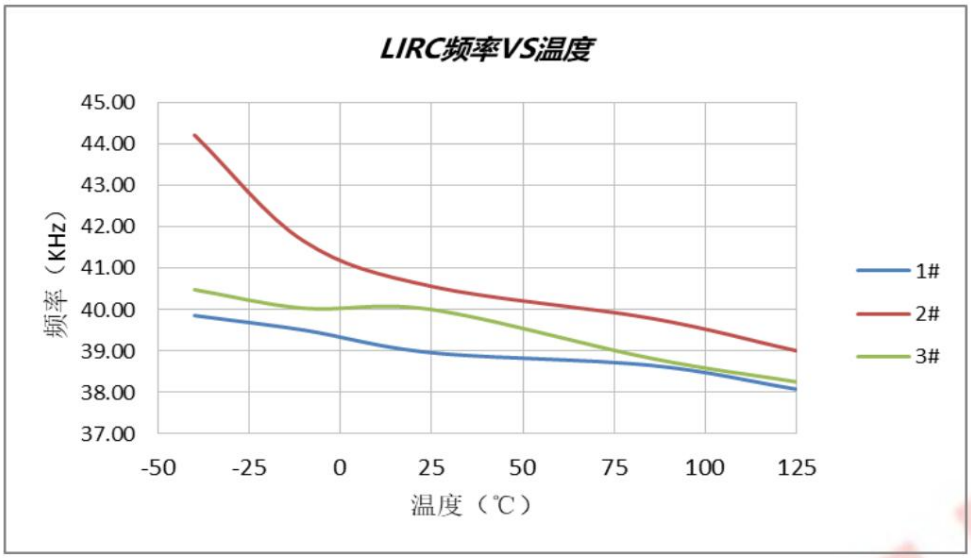
10.3 Analog Circuit Characteristics

HIRC Frequency vs Supply Voltage/Temperature



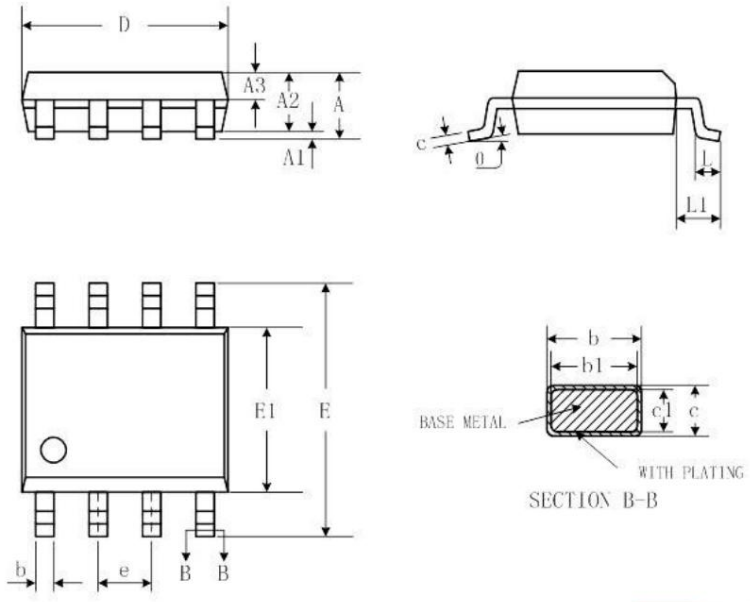
LIRC Frequency VS Supply Voltage/Temperature





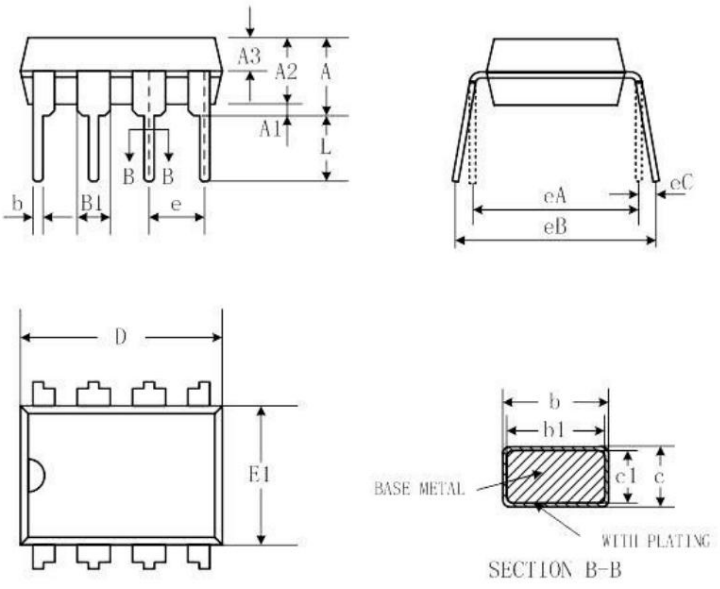
11 Package size

11.1 SOP8



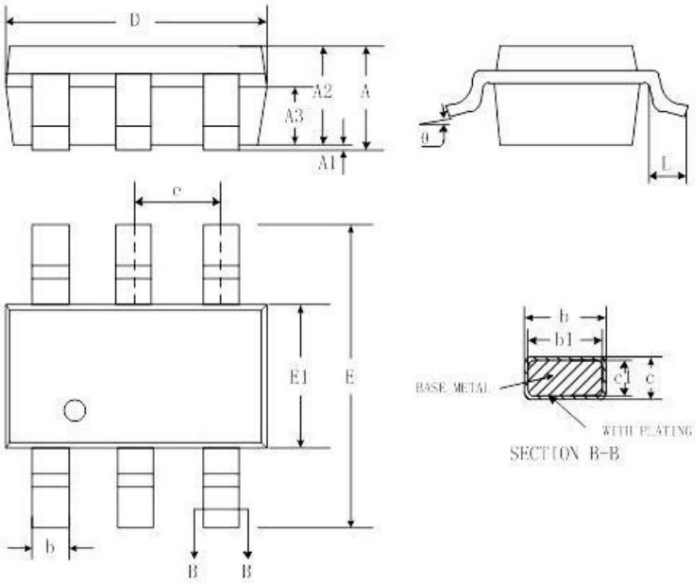
SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.77
A1	0.08	0.18	0.28
A2	1.20	1.40	1.60
A3	0.55	0.65	0.75
b	0.39	-	0.48
b1	0.38	0.41	0.43
c	0.21	-	0.26
c1	0.19	0.20	0.21
D	4.70	4.90	5.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
L	0.50	0.65	0.80
L1	1.05BSC		
θ	0	-	8°

11.2 DIP8



SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	3.60	3.80	4.00
A1	0.51	-	-
A2	3.10	3.30	3.50
A3	1.50	1.60	1.70
b	0.44	-	0.53
b1	0.43	0.46	0.48
B1	1.52BSC		
c	0.25	-	0.31
c1	0.24	0.25	0.26
D	9.05	9.25	9.45
E1	6.15	6.35	6.55
e	2.54BSC		
eA	7.62BSC		
eB	7.62	-	9.50
eC	0	-	0.94
L	3.00	-	-

11.3 SOT23-6



SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.35
A1	0.04	-	0.15
A2	1.00	1.10	1.20
A3	0.55	0.65	0.75
b	0.30	-	0.50
b1	0.30	0.40	0.45
c	0.08	-	0.22
c1	0.08	0.13	0.20
D	2.72	2.92	3.12
E	2.60	2.80	3.00
E1	1.40	1.60	1.80
e	0.95BSC		
L	0.30	-	0.60
θ	0	-	8°

12 Revision History

Version	Revision Date	Revisions
V1.0	2020-10-15	Initial version released;
V1.1	2021-01-07	Adjusted LVR parameters to ±15%;