GUANGZHOU WAYTRONIC TECHNOLOGY CO.,LTD

WT588D
Data Sheet

Special Features:

- Modules with 16pin, 28pin COB package, built-in flash memory
  Chips with DIP18 or SSOP20 or LQFP32 package, external flash required
- Playback duration depending on external SPI flash size (2M-32M SPI flash supported)
- Integrated high-speed audio processor
- User selectable DAC/PWM output modes
- Integrated 13Bit D/A converter, 12Bit PWM output with excellent quality
- PWM output directly drives a 0.5W/8Ω speaker
- Robust design for industrial environments
- Optional BUSY output

Electrical Features:

- Operating voltage: 2.8VDC - 5.5VDC
- Sampling frequency: 6KHZ - 20KHZ
- Automatic sleep mode after playback
- Sleep mode current: 10uA
- 10ms key trigger time
- 17ms reset time
- Serial baud rate 40us - 4000us

Peripheral Features:

- Download via USB port, online and offline download supported (online download: data my be written to SPI flash with WT588D connected and powered)
- Support for
  - mp3 control mode
  - key control mode
  - 3x8 key combination control mode
  - parallel control mode
  - one- line serial control mode
  - three-line serial control mode
  - three-line control I/O extension output mode
- Switch from three-line serial control mode to three-line control I/O extension output mode by sending serial command possible even during active playback
- 220 controllable addresses, each address may be loaded with a maximum of 128 phrases; phrases within an address can be combined for playback
- 15 different key control modes may be assigned to each input individually

Support Software:

- PC software to arrange voice files and insert mute duration between 10ms and 25minutes (mute does not consume memory space)
- Preview mode for loaded voice files
- Loaded voice phrases can be reused at multiple addresses
- 500 voice phrases max for editing
- Support for 8 chord MIDI, mp3 and WAV

Functional Description

The WT588D is a powerful, re- programmable voice chip using external SPI FLASH memory to store voice data.
The easy to use PC software provides all necessary features to arrange voice files, set control modes and download the compiled project file to SPI FLASH memories. Even with the WT588D chip connected and powered, data may be downloaded to the external memory without damage. Updates, including operating modes, will take effect after a valid reset of the WT588D chip.

A total of 15 trigger modes and up to 10 trigger inputs (depending on package) are supp orted:

- In the Mp3 control mode, a familiar Mp3 control interface is emulated: play, pause, previous, next, volume up and volume down
- In the key trigger mode, the following functions may be assigned to each key individually: repeatable edge trigger, unrepeatable edge trigger, level hold, level unhold, non - level hold, single key forward repeat, single key backward repeat, volume up, volume down,
  play/pause, stop, play/stop
- In the 3x8 key combination control mode, a tot al of 24 addresses may be triggered in edge trigger repeatable mode
- In the parallel control mode, a maximum of 8 input ports may be used to trigger up to 256 addresses
- In the serial control mode, two different modes of operation are possible: se rial three line control mode and serial control I/O port expansion mode.
  - The serial three line control mode enables an external MCU to control play, replay, stop and volume level, or directly trigger up to 219 addresses using a simple serial protocol.
  - The serial control I/O expansion mode provides control over an 8bit output port.

  Both serial control modes may be switched during playback. Note: the current playback will remain in the previous control mode!
- The single- line serial control mode offers play, stop, replay and volume level control using a simple one- wire protocol

The integrated PWM output directly drives a 0.5W/8 Ω speaker and the DAC output must be used in combination with an external amplifier.

The versatile WT588D can be used to add high quality voice to a wide variety of applications, including public transportation information announcement systems, alarm systems, personal reminders, alarm clocks, household appliances, medical instruments, electronic toys, telecommunication systems and parking distance control.

Table of Contents

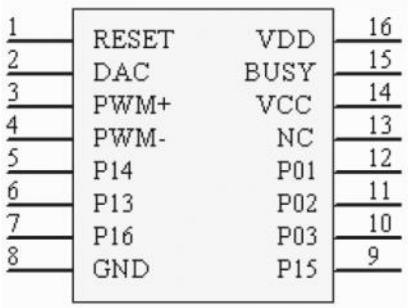
1.0 Device Varieties

TYPE	PACKAGE	MODEL	DURATION（@6K）	MEMORY
MODULE	16PIN MODULE (COB & HARD PACKAGE）	WTW30-16	32S	2M
		WTW100-16	102S	4M
		WTW230-16	238S	8M
		WTW510-16	516S	16M
		WTW1000-16	1054S	32M
	28PIN MODULE (COB & HARD PACKAGE）	WTW30-28	32S	2M
		WTW100-28	102S	4M
		WTW230-28	238S	8M
		WTW510-28	516S	16M
		WTW1000-28	1054S	32M
CHIP	DIP18	WT588D-18P	EXTERNAL 2M-32M (32-1054S)	
	SSOP20	WT588D-20SS	EXTERNAL 2M-32M (32-1054S)	
	LQFP32	WT588D-32L	EXTERNAL 2M-32M (32-1054S)	

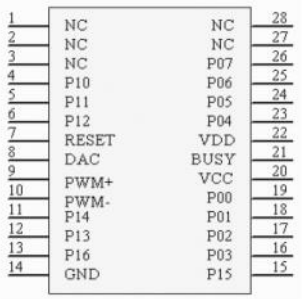
2.0 Relation between memory capacity (M), playback duration (D) and sample rate (S)

M / D / S	2M	4M	8M	16M	32M	64M
6K	33	101	238	511	1057	2149
8K	25	76	178	383	793	1612
10K	20	61	143	307	634	1290
12K	17	51	119	255	529	1075
14K	14	43	102	219	453	921
16K	12	38	89	192	396	806
18K	11	34	79	170	352	716
20K	10	30	71	153	317	645

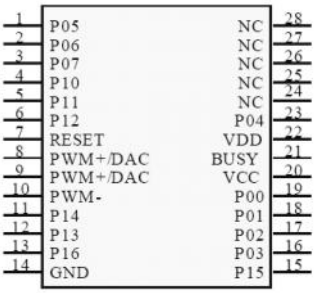
3.0 Pin Diagrams



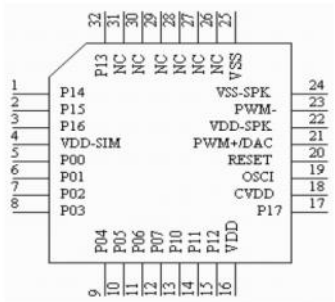
16PIN MODULE



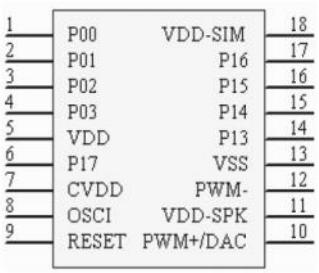
28PIN MODULE (V1.1)



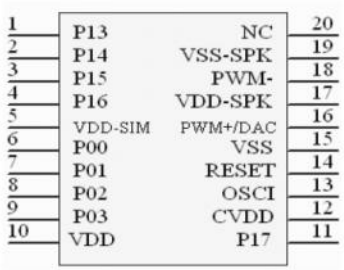
28PIN MODULE (V1.2)



LQPF32 CHIP



DIP18 CHIP



SSOP20 CHIP

4.0 Pin Descriptions

4.1 16PIN Module

Pin	Name	Description	Function
1	RESET	RESET	RESET Pin
2	PWM+/DAC	PWM+/DAC	PWM+/DAC audio output
3	PWM+/DAC	PWM+/DAC	PWM+/DAC audio output
4	PWM-	PWM-	PWM- audio output
5	P14	DI	Programming input
6	P13	DO	Programming output
7	P16	CLK	Clock
8	GND	GND	Ground
9	P15	CS	Programming Chip- Select
10	P03	K4/CLK/DATA	KEY/Three- Line: Clock; One-Line: data Input
11	P02	K3/CS	KEY/Three- Line Chip- Select Input
12	P01	K2/DATA	KEY/Three- Line Data Input
13	P00	K1	KEY Input
14	VCC	VCC	External memory power supply input
15	BUSY	BUSY	PLAY/ BUSY output
16	VDD	VDD	Positive Power Supply

4.2 28PIN Module (V1.1)

Pin	Name	Description	Function
1	NC	NC	NC
2	NC	NC	NC
3	NC	NC	NC
4	P10	K9/A7/R1	KEY/Parallel Address/Matrix Row Input/Three - Line Serial Extension Address
5	P11	K10/R2	KEY/Matrix Row Input/Three- Line Serial Extension Address
6	P12	R3	Matrix Row Input/Three- Line Extension Address
7	RESET	RESET	RESET
8	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
9	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
10	PWM-	PWM-	PWM- Audio Output
11	P14	DI	Programming Data Input
12	P13	DO	Programming Data Output
13	P16	CLK	Programming Clock
14	GND	GND	Ground
15	P15	CS	Programming Chip- Select
16	P03	K4/A2/L3/CLK/DATA	KEY/Parallel Address/Matrix Column/Three - Line Clock/One- Line Data Input
17	P02	K3/A1/L2/CS	KEY/Parallel Address/Matrix Column/Three - Line Chip-Select Input
18	P01	K2/A0/L1/DATA	KEY/Parallel Address/Matrix Column/Three - Line Data Input
19	P00	K1/L0/SBT	KEY/Matrix Column/Single button trigger (Parallel Address)/Three- Line Serial Extension Output Address
20	VCC	VCC	External memory power supply input
21	BUSY	BUSY	PLAY/ BUSY output
22	VDD	VDD	Digital Power Supply
23	P04	K5/A3/L4	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
24	P05	K6/A4/L5	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
25	P06	K7/A5/L6	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
26	P07	K8/A6/L7	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
27	NC	NC	NC
28	NC	NC	NC

4.3 28PIN Module (V1.2)

Pin	Name	Description	Function
1	P05	K6/A4/L5	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
2	P06	K7/A5/L6	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
3	P07	K8/A6/L7	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
4	P10	K9/A7/R1	KEY/Parallel Address/Matrix Row Input/Three - Line Serial Extension Address
5	P11	K10/R2	KEY/Matrix Row Input/Three- Line Serial Extension Address
6	P12	R3	Matrix Row Input/Three- Line Extension Address
7	RESET	RESET	RESET
8	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
9	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
10	PWM-	PWM-	PWM- Audio Output
11	P14	DI	Programming Data Input
12	P13	DO	Programming Data Output
13	P16	CLK	Programming Clock
14	GND	GND	Ground
15	P15	CS	Programming Chip- Select
16	P03	K4/A2/L3/CLK/DATA	KEY/Parallel Address/Matrix Column/Three - Line Clock/One- Line Data Input
17	P02	K3/A1/L2/CS	KEY/Parallel Address/Matrix Column/Three - Line Chip-Select Input
18	P01	K2/A0/L1/DATA	KEY/Parallel Address/Matrix Column/Three - Line Data Input
19	P00	K1/L0/SBT	KEY/Matrix Column/Single button trigger (Parallel Address)/Three- Line Serial Extension Output Address
20	VCC	VCC	External memory power supply in put
21	BUSY	BUSY	PLAY/ BUSY output
22	VDD	VDD	Digital Power Supply
23	P04	K5/A3/L4	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
24	NC	NC	NC
25	NC	NC	NC
26	NC	NC	NC
27	NC	NC	NC
28	NC	NC	NC

4.4 LQFP32 Chip

Pin	Name	Description	Function
1	P14	DI	Programming Data Input
2	P15	CS	Programming Chip- Select
3	P16	CLK	Programming Clock
4	VDD-SIM	VDD-SIM	External memory power supply input
5	P00	K1/L0/SBT	KEY/Matrix Column/Single button trigger (Parallel Address)/Three- Line Serial Extension Output Address
6	P01	K2/A0/L1/DATA	KEY/Parallel Address/Matrix Column/Three - Line Data Input
7	P02	K3/A1/L2/CS	KEY/Parallel Address/Matrix Column/Three - Line Chip-Select Input
8	P03	K4/A2/L3/CLK/DATA	KEY/Parallel Address/Matrix Column/Three- Line Clock/One- Line Data Input
9	P04	K5/A3/L4	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
10	P05	K6/A4/L5	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
11	P06	K7/A5/L6	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
12	P07	K8/A6/L7	KEY/Parallel Address/Matrix Column Input/Three - Line Serial Extension Output Address
13	P10	K9/A7/R1	KEY/Parallel Address/Matrix Row Input/Three- Line Serial Extension Address
14	P11	K10/R2	KEY/Matrix Row Input/Three- Line Serial Extension Address
15	P12	R3	Matrix Row Input/Three- Line Extension Address
16	VDD	VDD	Digital Power Supply
17	P17	BUSY	PLAY/ BUSY output
18	CVDD	CVDD	VDD power supply adjustment input
19	OSCI	OSCI	Crystal oscillator input
20	RESET	RESET	RESET
21	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
22	VDD-SPK	VDD-SPK	Audio power supply
23	PWM-	PWM-	PWM-/DAC Audio Output
24	VSS-SPK	VSS-SPK	Audio power ground
25	VSS	VSS	Ground
26	NC	NC	NC
27	NC	NC	NC
28	NC	NC	NC
29	NC	NC	NC
30	NC	NC	NC
31	NC	NC	NC
32	P13	DO	Programming Data Output

4.5 DIP18 Chip

Pin	Name	Description	Function
1	P00	K1/SBT	KEY/Matrix Column/Single button trigger (Parallel Address)/Three- Line Serial Extension Output Address
2	P01	K2/A0/DATA	KEY/Parallel Address/Matrix Column/Three - Line Data Input
3	P02	K3/A1/CS	KEY/Parallel Address/Matrix Column/Three - Line Chip-Select Input
4	P03	K4/A2/CLK/DATA	KEY/Parallel Address/Matrix Column/Three- Line Clock/One- Line Data Input
5	VDD	VDD	Digital Power Supply
6	P17	BUSY	PLAY/ BUSY output
7	CVDD	CVDD	VDD power supply adjustment input
8	OSCI	OSCI	Crystal oscillator input
9	RESET	RESET	RESET
10	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
11	VDD-SPK	VDD-SPK	Audio power supply
12	PMW-	PWM-	PWM-/DAC Audio Output
13	VSS	VSS	Ground
14	P13	DO	Programming Data Output
15	P14	DI	Programming Data Input
16	P15	CS	Programming Chip- Select
17	P16	CLK	Programming Clock
18	VDD-SIM	VDD-SIM	External memory power supply input

4.6 SSOP20 Chip

Pin	Name	Description	Function
1	P13	DO	Programming Data Output
2	P14	DI	Programming Data Input
3	P15	CS	Programming Chip- Select
4	P16	CLK	Programming Clock
5	VDD-SIM	VDD-SIM	External memory power supply input
6	P00	K1/SBT	KEY/Matrix Column/Single button trigger (Parallel Address)/Three- Line Serial Extension Output Address
7	P01	K2/A0/DATA	KEY/Parallel Address/Matrix Column/Three - Line Data Input
8	P02	K3/A1/CS	KEY/Parallel Address/Matrix Column/Three- Line Chip-Select Input
9	P03	K4/A2/CLK/DATA	KEY/Parallel Address/Matrix Column/Three - Line Clock/One- Line Data Input
10	VDD	VDD	Digital Power Supply
11	P17	BUSY	PLAY/ BUSY output
12	CVDD	CVDD	VDD power supply adjustment input
13	OSCI	OSCI	Crystal oscillator input
14	RESET	RESET	RESET
15	VSS	VSS	Ground
16	PWM+/DAC	PWM+/DAC	PWM+/DAC Audio Output
17	VDD-SPK	VDD-SPK	Audio power supply
18	PWM-	PWM-	PWM-/DAC Audio Output
19	VSS-SPK	VSS-SPK	Audio power ground
20	NC	NC	NC

5.0 I/O Port Description

5.1 Use of K1 - K10:

K1 - K10 are defined as key trigger inputs of the I/O port, corresponding to I/O port P00 – P07 and P10 - P11. The trigger mode of keys K1 – K10 can be set to edge retrigger, edge non retrigger, level unloop, level loop , level hold loop, prev unloop, next unloop, prev loop, next loop, no function, play/pause, stop, vol+, vol - and on/off.

KEY MODE: P00→K1
P01→K2
P02→K3
P03→K4
P04→K5
P05→K6
P06→K7
P07→K8
P10→K9
P11→K10

ONE-LINE SERIAL MODE: P00→K1
P01→K2
P02→K3
P04→K5
P05→K6
P06→K7
P07→K8
P10→K9
P11→K10

5.2 Use of R1 - R3 and L0 - L7

In the 3x8 matrix key control mode, I/O ports are defined as R1 - R3 as matrix row inputs and L0 - L7 as matrix column inputs. A phrase can be triggered and played when row (R) and co lumn (L) inputs are shorted. In this control mode, all keys trigger modes are set to edge non retrigger.

5.3 Use of SBT and A0 - A7

In the parallel control mode, I/O ports P01 - P07 and P10 correspond to address inputs A0 – A7 and P00 is defined as single button trigger (SBT).

6.0 Control modes

6.1 MP3 control mode

In the MP3 control mode, the following standard functions are assigned to the I/O port:

I/O	P00	P01	P02	P03	P04	P05
FUNCTION	STOP	PLAY/PAUSE	NEXT	PREVIOUS	VOL+	VOL-

6.2 KEY CONTROL MODE

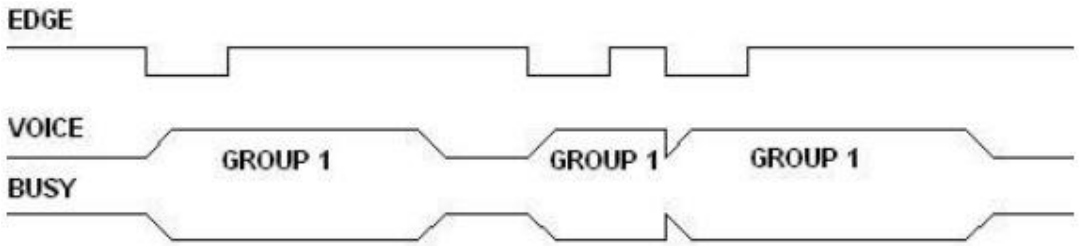
Each I/O pin may be assigned a separate key trigger mode with a debounce time of 10ms each.

The following trigger modes can be selected:

- edge retrigger
- edge non retrigger
- level unloop
- level loop
- level hold loop
- prev unloop
- next unloop
- prev loop
- next loop
- no function
- play/pause
- stop
- vol+
- vol-
- on/off

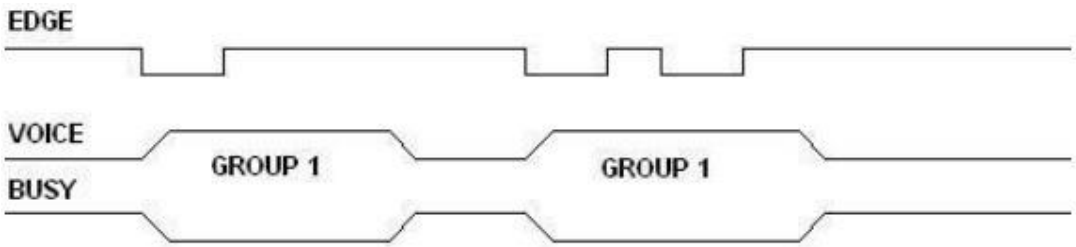
See the following section for detailed descriptions of all trigger modes.

6.2.1 Edge retrigger



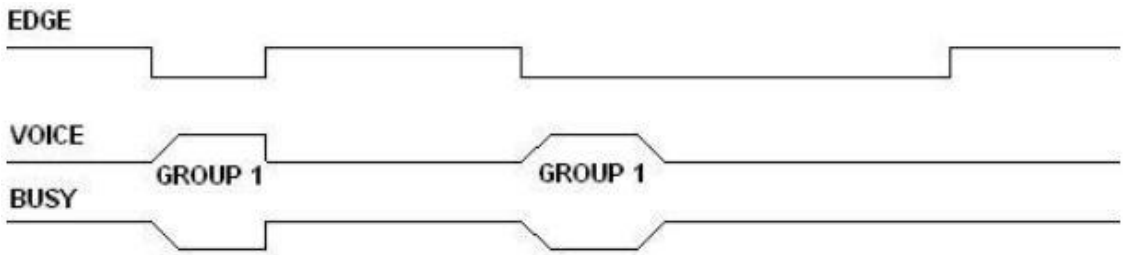
Remark: A falling edge on the I/O pin will trigger voice playback. A negati ve edge during active playback will stop the current playback and restart playback of the same voice file.

6.2.2 Edge non retrigger



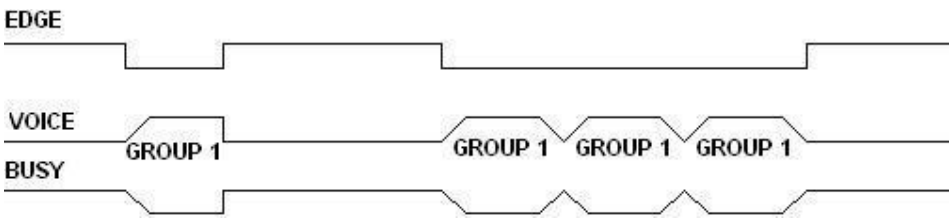
Remark: A falling edge on the I/O pin will trigger voice playback. A negative edge during active playback will not interrupt the current playback.

6.2.3 Level unloop



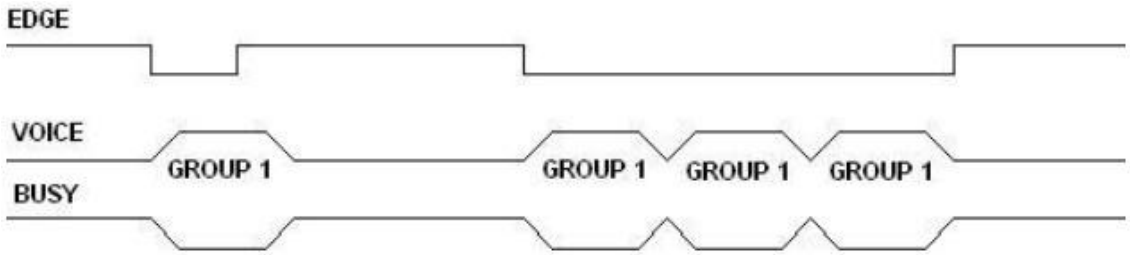
Remark: A low level on the I/O pin will trigger voice playback. Playback can be interrupted by a high level during playback. If the low level exceeds the playback time of the voice file, the voice file will not be replayed.

6.2.4 Level loop



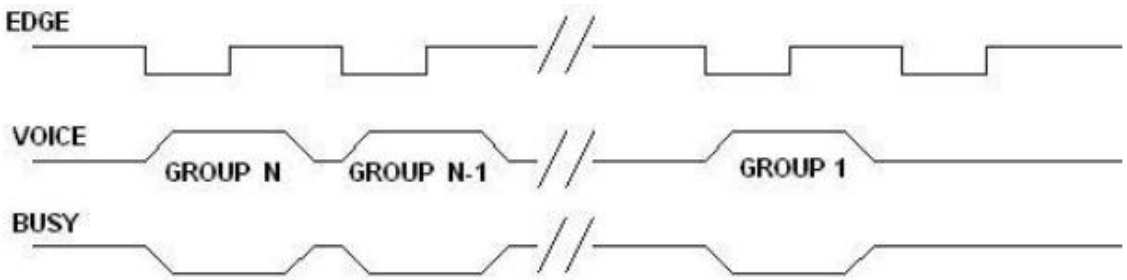
Remark: A low level on the I/O pin will trigger voice playback. Playback can be interrupted by a high level during playback. If the low level exceeds the playback time of the voice file, the voice file will be replayed until the I/O port reverts to a high level.

6.2.5 Level hold loop



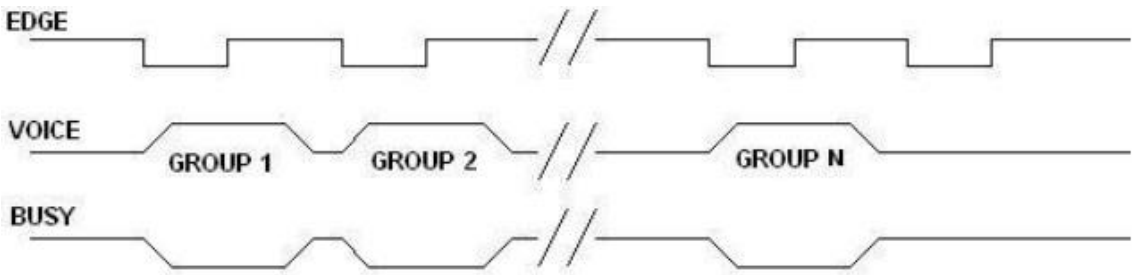
Remark: A low level on the I/O pin will trigger voice playback. Playback can not be interrupted by a high level during playback – the voice file will be played back until the end of the file is reached. If the low level exceeds the playback time of the voice file, the voice file will be replayed until the I/O port reverts to a high level.

6.2.6 Prev unloop



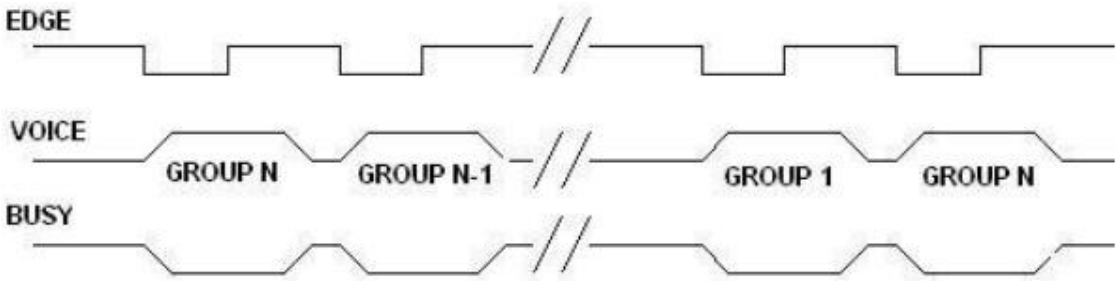
Remark: A falling edge on the I/O pin will trigger voice playback. Each consecutive falling edge will trigger the voice file stored before the previously played voice file in memory until the first voice file is reached.

6.2.7 Next unloop



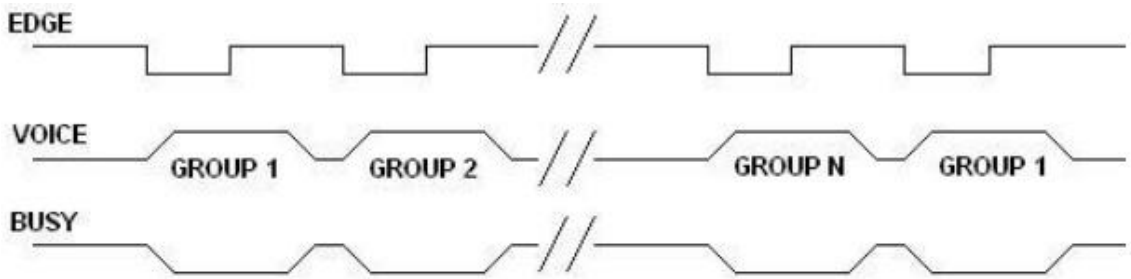
Remark: A falling edge on the I/O pin will trigger voice playback. Each cons ecutive falling edge will trigger the voice file stored after the previously played voice file in memory until the last voice file is reached.

6.2.8 Prev loop



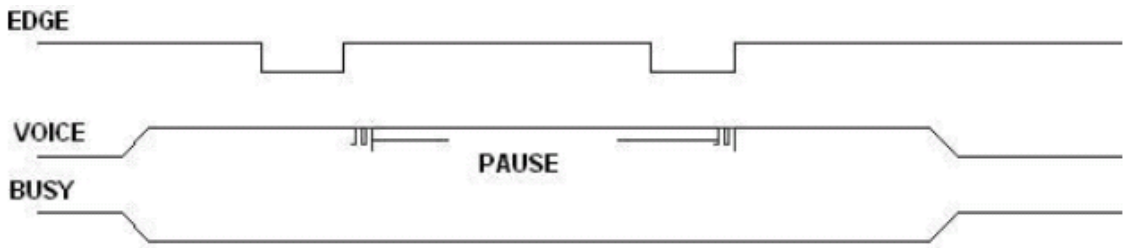
Remark: A falling edge on the I/O pin will trigger voice playback. Each consecutive fa lling edge will trigger the voice file stored before the previously played voice file in memory until the first voice file is reached and then roll over to the last voice file stored in memory.

6.2.9 Next loop



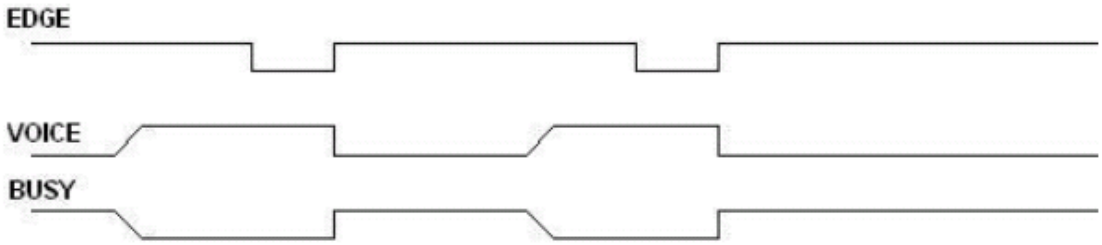
Remark: A falling edge on the I/O pin w ill trigger voice playback. Each consecutive falling edge will trigger the voice file stored after the previously played voice file in memory until the last voice file is reached and then roll over to the first voice file stored in memory.

6.2.10 Play/Pause



Remark: A falling edge on the I/O pin will trigger voice playback. Each consecutive falling edge will toggle between pause and play.

6.2.11 Stop



Remark: A falling edge on the I/O pin will stop active voice playback. A consecutive trigger will be ignored if no playback is active.

6.3 3X8 Matrix Key Control Mode

In this mode, all keys are set to edge retrigger mode. A total of 24 addresses can be selected.

I/O	P00	P01	P02	P03	P04	P05	P06	P07
P10	KEY 1	KEY 2	KEY 3	KEY 4	KEY 5	KEY 6	KEY 7	KEY 8
P11	KEY 9	KEY 10	KEY 11	KEY 12	KEY 13	KEY 14	KEY 15	KEY16
P12	KEY 17	KEY 18	KEY 19	KEY 20	KEY 21	KEY 22	KEY 23	KEY24

6.4 Parallel Control Mode

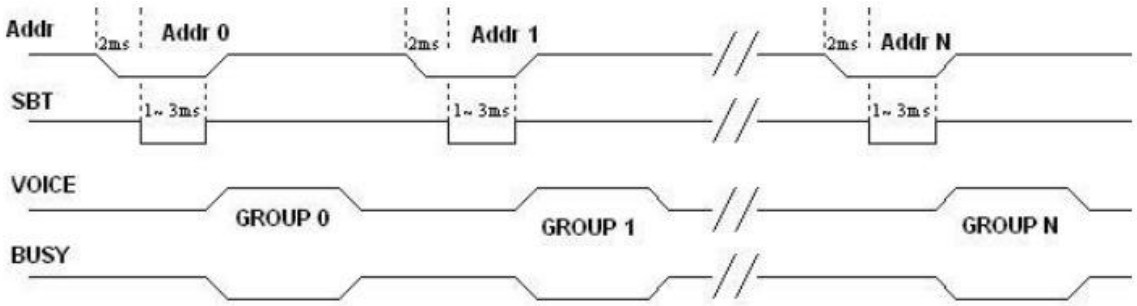
In the parallel control mode, I/O ports P01 - P07 and P10 correspond to address inputs A0 – A7 and P00 is defined as single button trigger (SBT). The trigger mode of SBT can be set to edge retrigger, edge on retrigger, on/off, prev unloop, next unloop, prev loop, next loop,level unloop, level loop, level hold loop, play/pause.

PACKAGE	PIN								
	P00	P01	P02	P03	P04	P05	P06	P07	P10
DIP18	SBT	A0	A1	A2	--	--	--	--	--
SSOP20	SBT	A0	A1	A2	--	--	--	--	--
LQFP32	SBT	A0	A1	A2	A3	A4	A5	A6	A7

6.4.1 Addresses

GROUP N	Address inputs							
	A7	A6	A5	A4	A3	A2	A1	A0
GROUP0	0	0	0	0	0	0	0	0
GROUP1	0	0	0	0	0	0	0	1
GROUP2	0	0	0	0	0	0	1	0
……	……	……	……	……	……	……	……	……
GROUP217	1	1	0	1	1	0	0	1
GROUP218	1	1	0	1	1	0	1	0
GROUP219	1	1	0	1	1	0	1	1

6.4.2 Timing diagram for external MCU



6.5 ONE- LINE CONTROL MODE

In the one- line mode, the user has full control over the device via the serial interface. The key inputs from P00 - P10 can be set to any trigger function.

6.5.1 I/O assignment

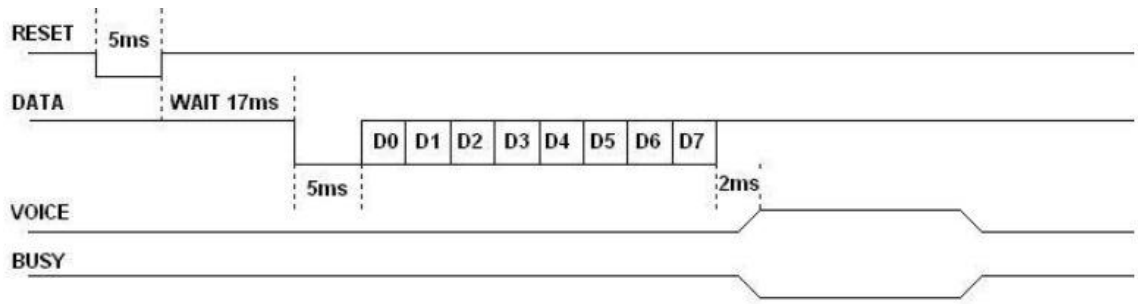
PACKAGE	I/O									
	P00	P01	P02	P03	P04	P05	P06	P07	P10	P11
SOP28	KEY K1	KEY K2	KEY K3	DATA	KEY K5	KEY K6	KEY K7	KEY K8	KEY K9	KEY K10
SSOP28	KEY K1	KEY K2	KEY K3	DATA	KEY K5	KEY K6	KEY K7	KEY K8	KEY K9	KEY K10

6.5.2 Instruction summary

CODE	FUNCTION	DESCRIPTION
E0H	VOLUME ADJUSTMENT	During playback or standby, the volume level may be adjusted in eight levels, E0H is minimum, E7H is maximum
F2H	LOOP PLAY	During active playback, a voice file may be set to loop
FEH	STOP	Stop active playback

6.5.3 Timing diagram for external MCU

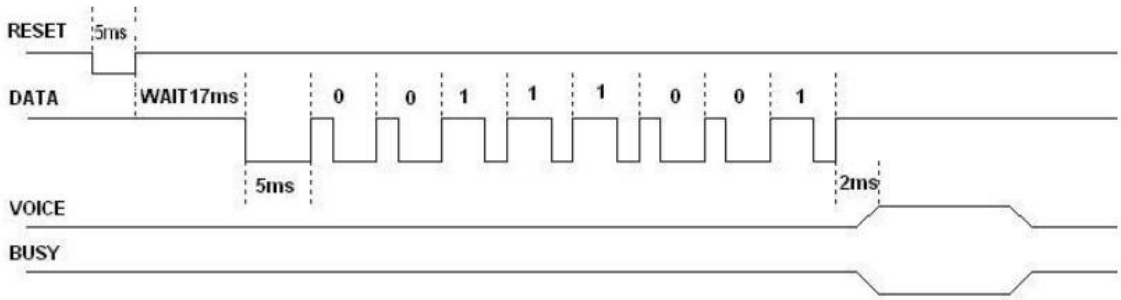
In the One Line Serial Mode the information of the data bit depends on the duty cycle. The data bit is 0 if the duty cycle is 1/3 and 1 if the duty cycle is 2/3.



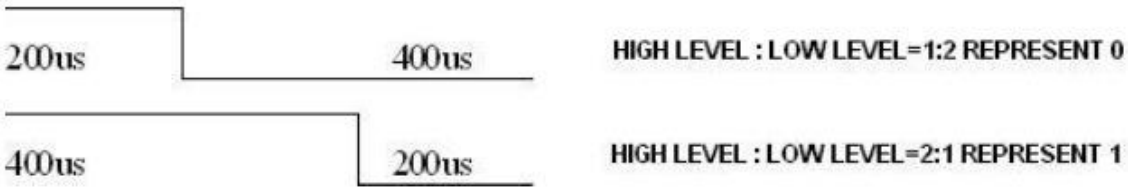
Remark: Pull RESET low for at least 5ms then wait another 17ms until first data bit is sent to ensure proper operation in noisy environments.

6.5.4 Example timing diagram

The following example shows the timing diagram for code 9CH:



6.5.5 Duty cycle versus data representation



6.5.6 Code example

```
MICROCONTROLLER: PIC16F54, CLOCK SPEED:4MHz
Send oneline(unsigned char addr)
{
        rst=0; /*reset the chip */
        delay1ms(2); /*delay 1ms to 5ms */
        rst=1;
        delay1ms(6); /*delay 6ms */
        sda=0;
        delay1ms(5); /* delay 5ms */

                for(i=0;i<8;i++)
                        {
                                sda=1;

                                        if(addr & 1)
                                        {
                                                delay1us(600); /* 600us */
                                                sda=0;
                                                delay1us(300); /* 300us */
                                        }
                                        else
                                        {
                                                delay1us(300); /* 300us */
                                                sda=0;
                                                delay1us(600); /* 600us */
                                        }
                                addr>>=1;
```

```
                    }
        sda=1;
}
```

6.6 Three- Line Serial Control Mode

This mode uses a standard SPI protocol to control device operation.

6.6.1 Pin assignment

PACKAGE	I/O									
	P00	P01	P02	P03	P04	P05	P06	P07	P10	P11
SOP28	--	DATA	CS	SCK	--	--	--	--	--	--
SSOP28	--	DATA	CS	SCK	--	--	--	--	--	--

6.6.2 Instruction summary

CODE	FUNCTION	DESCRIPTION
E0H-E7H	VOLUME ADJUSTMENT	During playback or standby, the volume level may be adjusted in eight levels, E0H is minimum, E7H is maximum
F2H	LOOP PLAY	During active playback, a voice file may be set to loop
FEH	STOP	Stop active playback
F5H	ENTER I/O EXTENSION OUTPUT MODE	Change from Serial Three Line Control Mode to I/O Extension Output Mode
F6H	EXIT I/O EXTENSION OUTPUT MODE	Change from I/O Extension Output Mode to Serial Three Line Control Mode

6.6.3 Address setup

DATA(HEX)	FUNCTION
00H	Play voice stored in address 0
01H	Play voice stored in address 1
02H	Play voice stored in address 2
…..	…….
D9H	Play voice stored in address 217
DAH	Play voice stored in address 218
DBH	Play voice stored in address 219

6.7 Three- Line Serial Control I/O Extension Output

In the Three- Line Serial Control Mode, sending F5H will change the device into Three - Line Serial Control I/O Expansion Output Mode. Sending F6H will exit this mode and change back to the Three- Line Serial Control Mode.

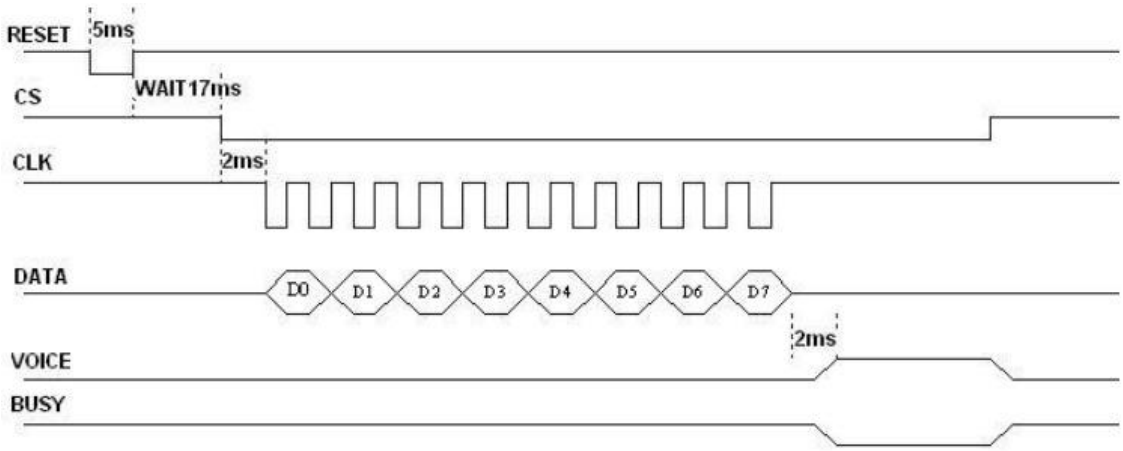
The Three- Line Serial Control I/O Expansion Mode offers up to eight individually controllable I/Os. All eight bits have to be sent in this mode.

GROUP	I/O							
	P12	P11	P10	P07	P06	P05	P04	P00
GROUP 0	0	0	0	0	0	0	0	0
GROUP 1	0	0	0	0	0	0	0	1
GROUP 2	0	0	0	0	0	0	1	0
……	……	……	……	……	……	……	……	……
GROUP217	1	1	0	1	1	0	0	1
GROUP218	1	1	0	1	1	0	1	0
GROUP219	1	1	0	1	1	0	1	1

Note that in the above table, 0 represents low level output and 1 represents high level output. A switch to I/O extension output from three-line serial control mode will not interrupt the currently active playback. E.g. If loop play is set, the last triggered voice file will keep looping until the control mode is switched back to Three- Line Serial Control Mode and the playback is stopped or paused. If I/O Extension Mode is changed to Thre e- Line Serial Control Mode, the last active I/O setting will remain active until it is changed again in I/O Extension Mode.

6.7.1 Example timing diagram

The Three- Line Serial Control Mode uses CS, SCK and DATA pins with standard SPI protocol timings. Pull RESET low for 1ms - 5ms before sending data. CS should be kept low for 2ms - 10ms to wake up the chip. Data is to be shifted out LSB first upon rising clock edge. The clock cycle should be between 40us - 2ms. The BUSY signal will become active 2ms afte r valid data was received.



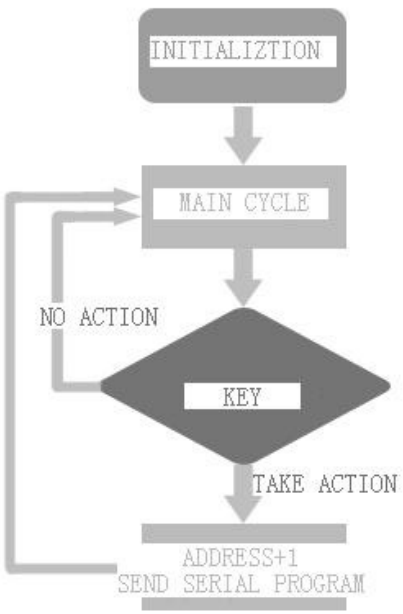
6.7.2 Program Example

(Microcotroller PIC16F54,system speed 4MHz)

```
Send threelines(unsigned char addr)
{
        rst=0;
        delay1ms(2); /* 2ms */
        rst=1;
        delay1ms(5); /* 5ms */
        cs=0;
        delay1ms(2);

                for(i=0;i<8;i++)
                {
                        scl=0;

                                if(addr & 1) sda=1;
                                else sda=0;

                        addr>>=1;
                        delay1us(30); /* 30us */
                        scl=1;
                        delay1us(30); /* 30us */
                }
        cs=1;
}
```

7.0 Control Timing

7.1 Control Timing Diagram



7.2 One- Line Serial Control program example

This program works for the one-line serial control mode application circuit given in “WT588D CHIP&MODULE DETAILED INFORMATION”.

```
ORG 0000H

        KEY EQU P1.1 ;
        RST EQU P1.4 ;
        SDA EQU P3.0 ;
        DAIFAZHI EQU 50H ;
        MOV  DAIFAZHI,#0H;
        MOV R5,#8 ;

MAIN:

        JB KEY,MAIN ;
        MOV R6,#20 ;
        LCALL DELAY1MS
        JB KEY,MAIN ;
        JNB KEY,$ ;
        LCALL one_line ;
        INC DAIFAZHI ;
        MOV A,DAIFAZHI
        CJNE A,#210,XX2 ;
XX2:
        JC XX3
        MOV DAIFAZHI,#0H
XX3:
        LJMP MAINone_line: ;
        CLR RST
        MOV R6,#5 ;
        LCALL DELAY1MS
        SETB RST
        MOV R6,#17 ;
        LCALL DELAY1MS
        CLR SDA
        MOV R6,#5 ;
        LCALL DELAY1MS
        MOV A,DAIFAZHI
LOOP:

        SETB SDA
        RRC A
        JNC DIDIANPIN ;
        LCALL DELAY200US
        LCALL DELAY200US
        CLR SDA
        LCALL DELAY200US
        LJMP LOOP1
DIDIANPIN:

        LCALL DELAY200US
        CLR SDA
        LCALL DELAY200US
```

```
        LCALL DELAY200US
LOOP1:

        DJNZ R5,LOOP
        MOV R5,#08H
        SETB SDA
        RET
DELAY200US:

        MOV R6,#100 ;
        DJNZ R6,$ RET
DELAY1MS:

        L1:
                MOV R7,#248
                DJNZ R7,$
                DJNZ R6,L1 RET
        END
```

7.3 One- Line Serial Control C- language program example

This program works for the one-line serial control mode application circuit given in “WT5 88D CHIP&MODULE DETAILED INFORMATION”.

```
#include <at89x2051.H>

sbit KEY = P1^1;
sbit RST = P1^4;
sbit SDA = P3^0;

void delay1ms(unsigned char count)
{
        unsigned char i,j,k;

        for(k=count;k>0;k--)
                for(i=2;i>0;i--)
                        for(j=248;j>0;j--);
}

void delay100us(unsigned char count)
{
        unsigned char i;
        unsigned char j;

        for(i=count;i>0;i--)
                for(j=50;j>0;j--);
}

Send_oneline(unsigned char addr)
{
        unsigned char i;

        RST = 0;
        delay1ms(5);
```

```
        RST = 1;
        delay1ms(17);
        SDA = 0;
        delay1ms(5);

        for(i=0;i<8;i++)
        {
                SDA=1;

                if(addr & 1)
                {
                        delay100us(4);
                        SDA = 0;
                        delay100us(2);
                }
                else
                {
                        delay100us(2);
                        SDA = 0;
                        delay100us(4);
                }

                addr>>=1;
        }

        SDA = 1;
}

main()
{
        unsigned char FD = 0;
        P3 = 0XFF;

        while(1)
        {
                if(KEY==0)
                {
                        delay1ms(10);

                        if(KEY==0)
                        {
                                Send_oneline(FD);
                                FD++;

                                if(FD==210)
                                {
                                        FD=0;
                                }
                        while(KEY==0);
                        }
                }
        }
}
```

7.4 Three- Line Serial Control Program example

This program works for the three - line serial control mode application circuit given in “WT588D CHIP&MODULE DETAILED INFORMATION”.

```
ORG 0000H
        KEY EQU P1.1 ;
        RST EQU P1.4 ;
        CS EQU P3.1 ;
        SCL EQU P3.2 ;
        SDA EQU P3.0 ;
        DAIFAZHI EQU 50H ;
        MOV DAIFAZHI,#0H;
        MOV R5,#8 ;

MAIN:
        JB KEY,MAIN
        MOV R6,#20 ;
        LCALL DELAY1MS
        JB KEY,MAIN ;
        JNB KEY,$ ;
        LCALL THREE_LINE;
        INC DAIFAZHI ;
        MOV A,DAIFAZHI
        CJNE A,#220,XX2 ;

XX2:
        JC XX3
        MOV DAIFAZHI,#0H

XX3:
        LJMP MAIN

THREE_LINE:
        CLR RST
        MOV R6,#5 ;
        LCALL DELAY1MS
        SETB RST
        MOV R6,#17 ;
        LCALL DELAY1MS
        CLR CS
        MOV R6,#2 ;
        LCALL DELAY1MS
        MOV A,DAIFAZHI

LOOP:
        CLR SCL
        RRC A
        MOV SDA,C
        LCALL DELAY50US
        SETB SCL
        LCALL DELAY50US
        DJNZ R5,LOOP
        MOV R5,#08H
        SETB CS
```

```
        RET
        DELAY50US:
        MOV R6,#25 ;
        DJNZ R6,$
        RET

DELAY1MS:

        L1:
                MOV R7,#248
                DJNZ R7,$
                DJNZ R6,L1
                RET
END
```

7.5 Three- Line Serial Control C- Language program example

This program works for the three-line serial control mode application circuit given in "WT588D CHIP&MODULE DETAILED INFORMATION".

```
#include <at89x51.H>
sbit KEY=P1^1;
sbit  RST=P1^4;
sbit CS=P3^1;
sbit SCL=P3^2;
sbit SDA=P3^0;
//sbit DENG=P3^7;

void delay1ms(unsigned char count)
{
        unsigned char i,j,k;

                for(k=count;k>0;k--)
                        for(i=2;i>0;i--)
                                for(j=248;j>0;j--);
}

void delay100us(void)
{
        unsigned char j;

                for(j=50;j>0;j--);
}

Send_threelines(unsigned char addr)
{
        unsigned char i;
        RST = 0;
        delay1ms(5);
        RST = 1;
        delay1ms(17);
        CS = 0;
        delay1ms(2);

        for(i=0;i<8;i++)
```

```
                {
                        SCL = 0;

                        if(addr & 1) SDA = 1;
                        else SDA = 0;

                        addr>>=1;
                        delay100us();
                        SCL = 1;
                        delay100us();
                }

        CS = 1;
}

main()
{
        unsigned char FD = 0;
        P3 = 0XFF;

        while(1)
        {
                if(KEY == 0)
                {
                        delay1ms(20);

                        if(KEY == 0)
                {

                Send_threelines(FD);
                FD++;

                if(FD==220
                {
                        FD=0;
                }

        while(KEY == 0);
        }
}
```

8.0 Revision history

VERSION	DATE	DESCRIPTION
V1.0	2008-5-17	ORIGINAL VERSION
V1.1	2008-7-15	REVISION
V1.2	2008-8-22	REVISION
V1.3	2008-8-30	REVISION
V1.4	2008-9-12	REVISION
V2.0	2008-11-29	REVISION

9.0 Manufacturer Information

Guangzhou Waytronic Technology Co., Ltd.