# SP03 Text to Speech Synthesizer

The robotics community has been without a low cost speech synthesizer chip for a long time. The ever popular SP0256-AL2 has long gone out of production, though there are a few still available. There are other multi-chip products around but none have captured the popular imagination like the Winbond WTS701. And with good reason, the WTS701 is not only a complete single chip synthesizer but also includes a text to speech processor. Given the near impossibility of producing a definitive set of rules for text to speech processing, the WTS701 performs impressively. The only downer for the hobbyist is that the WTS701 comes in a 56 lead TSOP package with pins on a 0.5mm pitch.

The SP03 module includes an audio amplifier, a 3volt regulator and level conversion to 5volts, a PIC processor to provide easy communication with your host processor and even a small 40mm speaker, along with the WTS701. Interfaces include an RS232 serial interface, I2C bus interface and a parallel interface to speak up to 30 predefined phrases. A PC program SP03.EXE is available to load the 30 predefined phrases into the SP03.

---

**Connections to the Speech  Module**
There are two connectors on the SP03 module. The 5v power supply may be applied to either of them - the other may be left unconnected.

**PL1**



|  | |
|---|---|
| +5V - | 5V Power Supply - up to 100mA |
| SDA - | IC2 bus SDA connection |
| SCL - | I2C bus SCL connection |
| No Connect - | Do not connect this pin |
| GND - | The 0 volt Ground line |
| Spare - | Undefined pin - do not connect |
| GND - | The 0 volt Ground line |
| RS232 Rx - | Connect to Tx on the PC |
| RS232 Tx - | Connect to Rx on the PC |

**PL2**

| | |
|---|---|
| +5V - | 5V Power Supply - up to 100mA |
| Status - | High when speaking, Low when done |
| Sel 4 - | These are the binary select inputs. They select one of the 30 predefined phrases |
| Sel 3 - | |
| Sel 2 - | |
| Sel 1 - | |
| Sel 0 - | |
| GND - | The 0 volt Ground line |
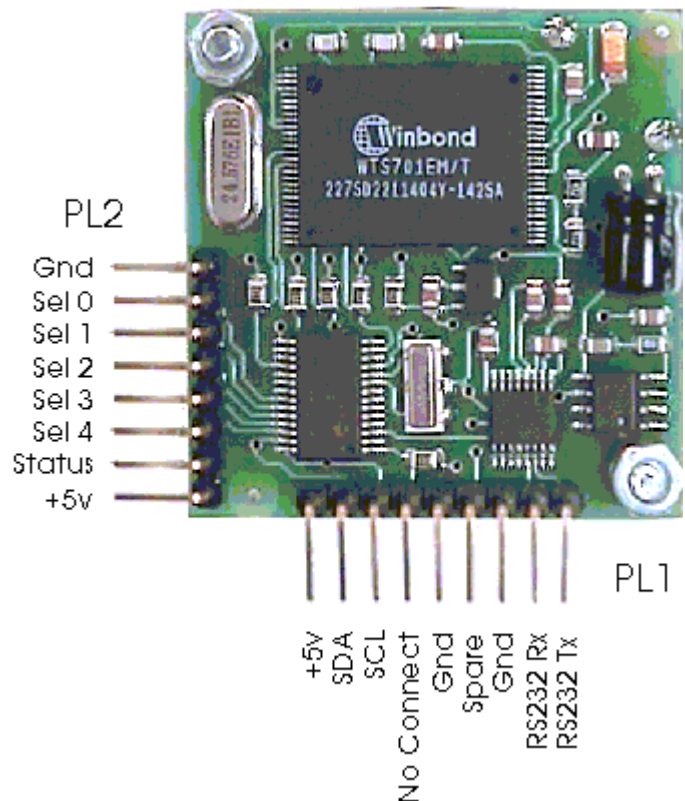
**RS232 Communications**
To use the RS232 serial port to control the SP03, just three connections to the PC (or other host) and a 5v power supply is required -  Ground, Rx, Tx and 5Volts. The modules RS232 Rx pin should be connected to the PC's Tx line (pin 3 on the DB9 socket). The SP03 RS232 Tx pin should be connect to the PC's Rx line (pin 2 on the DB9 socket). The SP03 Ground line is connected to pin 5 on the DB9

socket. The 5volt supply is not shown in the photo below.
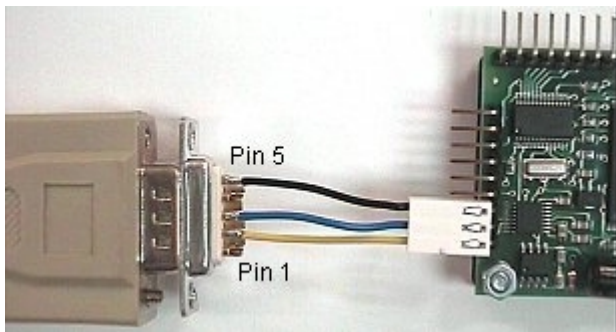


The serial data format is 8bits, No Parity, 2 stop bits, 38400 baud.

**RS232 Commands**
There are 32 serial commands that can be sent to the SP03. Thirty of these (commands 1 to 30, or 0x01 to 0x1E) are used to speak one of the thirty predefined phrases. To speak any of these phrases, just send a single command byte to the SP03. When the SP03 has finished speaking it will send the command back to the PC as an acknowledgement. Don't send any commands to the module whilst it is speaking, as they will be ignored. When the command is acknowledged, the module will be ready to receive another command.

Command 128 (0x80)
This command is used to speak a line of text. It is followed by 3 control bytes, then the ASCII text, then an 0x1a character and finally a zero (0x00 or NULL). The three control bytes are the volume, pitch and speed values for the text. For every byte sent, here will be an acknowledge byte sent back from the module. It is essential that you wait for the acknowledge byte before sending the next character. Here is the sequence to speak the word "Hello" at full volume and moderate pitch and speed. Note that the volume value 0-7 is from loudest to quietest. The PIC's text buffer is 85 bytes in size, so that is the limit for a single phrase. The volume, pitch, speed and trailing NULL character take 4 bytes leaving 81 for the text.

| Command byte Transmitted to SP03 Module | Acknowledge byte returned from SP03 |
|---|---|
| Command 0x80 | 0x01 |
| Full Volume 0x00 | 0x00 |
| Speech Pitch 0x04 | 0x04 |
| Speech Speed 0x02 | 0x02 |
| Text 'H' | 'H' |
| Text 'e' | 'e' |
| Text 'l' | 'l' |
| Text 'l' | 'l' |
| Text 'o' | 'o' |
| NULL 0x00 | 0x00 |
| SP03 will now speak the text | 0x00 indicates text loading is complete |

Command 129 (0x81)
This command is used to read back the WTS701's status register. After sending 0x81 to the module it will respond by sending back first the low byte and then the high byte of the WTS701 status register. This may be used to determine when the WTS701 has actually finished speaking the text. You should

consult the WTS701 data sheet for full details of the status bits, however a simple way to do this in C is;

```
do {
    SerOut(0x81);               // send Get Status command
    sts = SerIn();              // get low byte
    sts += SerIn()<<8;          // get high byte
    sts &= 0x8003;              // select bits to test
}while(sts != 0x8001);         // and loop until finished speaking
```

Command 130 (0x82)
This command is used by SP03.EXE to download the 30 predefined phrases into the PIC's Flash memory. Do not try to use it as the protocol requires that the text is sent in a compressed format. Use the configuration utility SP03.EXE instead.

All other values sent as commands will be ignored and no acknowledgement will be returned.

**I2C Bus Communications**
Along with the 5Volt power supply, the I2C bus just requires the SDA and SCL lines. The I2C interface does not have any pull-up resistors on the board, these should be provided elsewhere, most probably with the bus master. They are required on both the SCL and SDA lines, but only once for the whole bus, not on each module. I suggest a values of 4k7 for 100KHz and 1k8 if you are going to be working up to 400KHz or higher. If your going for higher speeds than 400KHz then you must separate *all* bytes transmitted over the I2C bus to the SP03 with a 40uS delay. This is to give the processor time to transfer the incoming data to the buffer. By doing this we have tested the SP03 with SCL up to 1MHz.

I2C communication protocol with the speech module is the same as popular eeprom's such as the 24C04. The SP03 only has two registers, the command register and the software revision number. To read the software revision number, first send a start bit, the module address (0XC4) with the read/write bit low, then the register number you wish to read (0x01). This is followed by a repeated start and the module address again with the read/write bit high (0XC5). You now read one byte which is the PIC software revision number and follow this with the stop bit.

| Register | Function |
|---|---|
| 0 | Command Register |
| 1 | Software Revision Number |

All commands and text to be spoken are sent to the command register. There are 32 valid commands, listed below:

| Command | Action |
|---|---|
| NOP (0x00) | No Action |
| SPKPRE 1 to 30, or 0x01 to 0x1E) | Speak pre-defined phrase |
| SPKBUF 64 or 0x40 | Speak text previously stored in buffer |

The NOP command is followed by the text that you want spoken. You may send as little or as much (up to the 85 byte limit) as you wish. A number of NOP sequences may be used to build up the buffer before the SPKBUF command is issued. The buffer is flushed after a SPKPRE or SPKBUF command is used. A text sequence is the same as for the RS232 protocol, that is, 3 control bytes, then the ASCII text, and

finally a zero (0x00 or NULL). The text in the buffer may then be spoken by sending a SPKBUF command. The PIC's text buffer is 85 bytes in size, so that is the limit for a single phrase. The volume, pitch, speed and trailing NULL characters take 4 bytes leaving 81 for the text.
To say "Hello" send the following sequence over the I2C bus:

| | |
|---|---|
| Start Bit | I2C Start Sequence |
| 0xC4 | SP03 I2C Address |
| 0x00 | SP03 Command Register |
| 0x00 | SP03 NOP Command |
| 0x00 | Volume (Max.) |
| 0x05 | Speech Speed |
| 0x03 | Speech Pitch |
| 'H' (0x48) | Text |
| 'e' (0x65) | Text |
| 'l' (0x6C) | Text |
| 'l' (0x6C) | Text |
| 'o' (0x6F) | Text |
| 0x00 | NULL |
| Stop Bit | I2C Stop Sequence |

Note - there is no command to "speak the following text", as the RS232 interface does. If you wish to speak a single phrase then you should send a NOP command followed by the text sequence. Then, in a separate I2C bus transaction, the SPKBUF command on its own.

To check to see when the SP03 has finished speaking, you can read back the command register. Whilst speaking, it will be the command that initiated the speaking, either 1-30 (0x01-0x1E) or 64 (0x40). It will be cleared to zero (0x00) when speaking is complete and the module is ready for the next phrase.
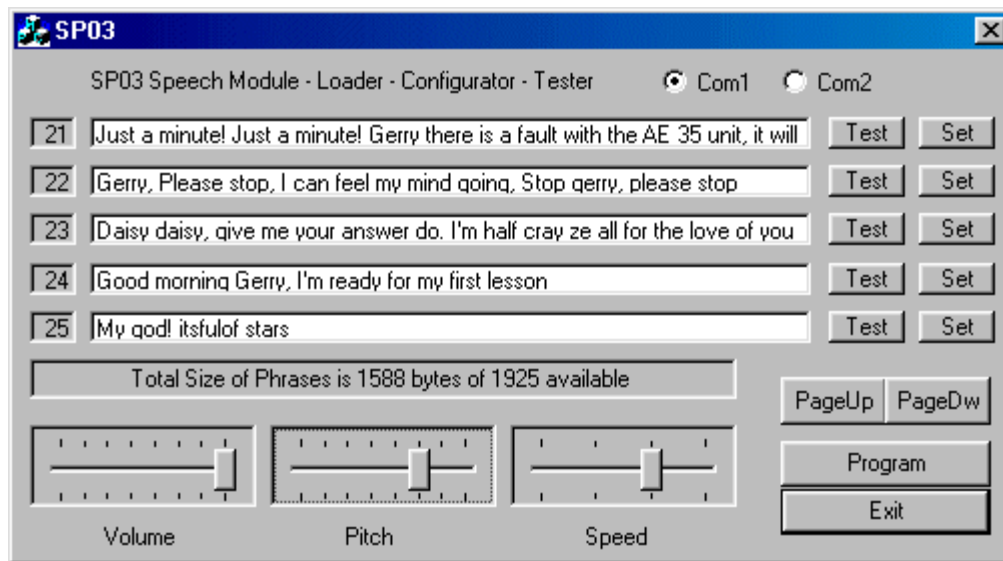
**Parallel Port Communications**
PL2 is used to speak one of the 30 pre-defined phrases. Text cannot be sent to the parallel port for speaking. To speak any of the 30 (1-30) pre-defined phrases, apply the phrase number to the 5 bit input port SEL4 - SEL0. The numbers 0 and 31 (0x00 and 0x1F) are just parking values and to not cause any phrase to be spoken. The SP03 has pull-up resistors on the inputs so they can be left unconnected if not used. As soon as the CPU has recognised and confirmed the new input (a few uS at most) it will raise the STATUS bit to a logic 1 (high) and speak the phrase. As soon as the STATUS bit goes high, the input code may be removed by returning the input to 0x00 or 0x1F. This must happen before the unit has finished speaking or the phrase will be repeated. The STATUS bit will go to a logic 0 (low) when the SP03 has finished speaking, and this can be monitored by the host.

**Power Up**
On power up, the module will speak phrase #1, if one has been stored by the configuration program SP03.EXE

**Configuration program SP03.EXE**

The SP03 configuration program is shown below

SP03

SP03 Speech Module - Loader - Configurator - Tester   ● Com1   ○ Com2

| 21 | Just a minute! Just a minute! Gerry there is a fault with the AE 35 unit, it will | Test | Set |
| 22 | Gerry, Please stop, I can feel my mind going, Stop gerry, please stop | Test | Set |
| 23 | Daisy daisy, give me your answer do. I'm half cray ze all for the love of you | Test | Set |
| 24 | Good morning Gerry, I'm ready for my first lesson | Test | Set |
| 25 | My god! itsfulof stars | Test | Set |

Total Size of Phrases is 1588 bytes of 1925 available

PageUp  PageDw

Program

Exit

Volume        Pitch        Speed

When you run the program for the first time, you should select the communications port you will be using, either COM1 or COM2. This will be remembered for the next time you run the program.
The SP03 configuration program can store 30 phrases in 6 pages of 5 phrases each. Press the PageUp and PageDw buttons to change pages. Below the 5 edit boxes for the phrases is a message/status bar and 3 sliders for volume, pitch and speed. The volume and Speed sliders work as expected but the pitch is a little strange, try the seventh position! The pitch seems out of sequence with the rest of the positions, but that's the WTS701. Below the PageUp and PageDw buttons is the program button. This will program all 30 phrases into the Flash memory of the PIC16F872 processor on the SP03.
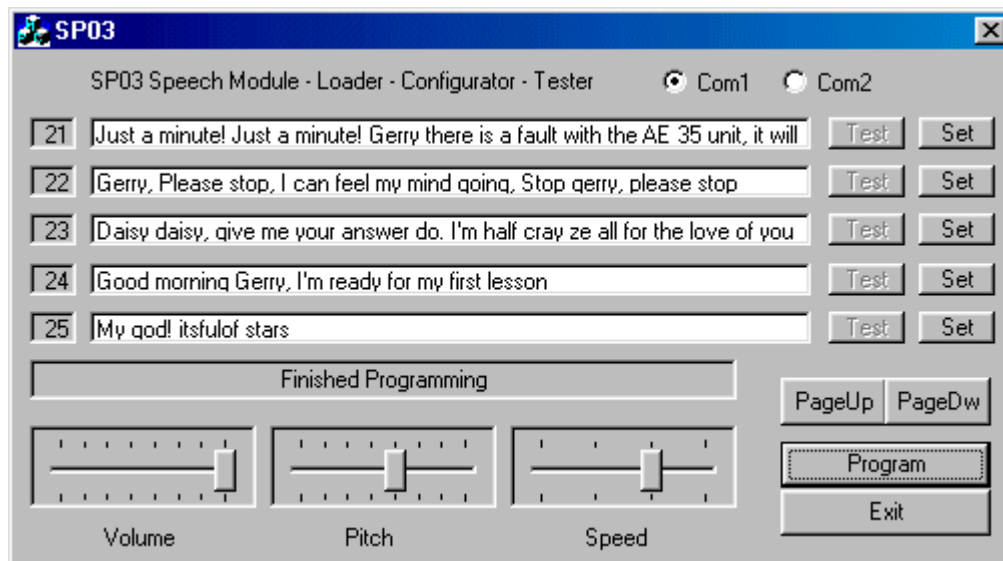
**SP03.EXE Operation**
The operation of the program is fairly easy. Start by selecting the Com port and setting the Volume, Pitch & Speed sliders as shown above. Now type something into one of the edit boxes and press the "Test" button to the right of that edit box. The words you typed will be spoken. Notice that the message bar keeps track of the number of characters used so far. This is the total for all 30 phrases.

**The "Test" and "Set" buttons**
Both of these buttons cause the phrase to be spoken. The "Test" button uses and also stores the Volume, Pitch & Speed values set on the sliders. The "Set" button uses the Volume, Pitch & Speed values stored from the previous use of the "Test" button. Therefore you can have different Volume, Pitch & Speed settings for each of the 30 phrases. When you've all your phrases set-up and tested using the "Set" buttons you're ready to program them into the PIC16F872.

**Programming the Phrases**
Easy! Just press the "Program" button. Your phrases will be compressed and stored in the Flash memory of the PIC16F872 processor. The message bar will report the progress of programming. When programming is complete you will see the following screen:

SP03 Speech Module - Loader - Configurator - Tester    ○ Com1   ○ Com2

| 21 | Just a minute! Just a minute! Gerry there is a fault with the AE 35 unit, it will | Test | Set |
| 22 | Gerry, Please stop, I can feel my mind going, Stop gerry, please stop | Test | Set |
| 23 | Daisy daisy, give me your answer do. I'm half cray ze all for the love of you | Test | Set |
| 24 | Good morning Gerry, I'm ready for my first lesson | Test | Set |
| 25 | My god! itsfulof stars | Test | Set |

Finished Programming

PageUp  PageDw

Program

Exit

Volume    Pitch    Speed

Notice that the "Test" buttons are de-selected. The remaining "Set" buttons have now changed mode and when pressed will cause the phrase to be spoken directly from the pre-defined phrases that you just programmed. Click in any of the edit boxes to restore the "Test" and "Set" buttons to normal.

**Baffle Cutout and Dimensions**



## SP03 Mounting Dimentions

3mm Dia.

35mm Circular Cutout
in Baffle for Speaker

3.2mm

36.8mm

40mm

3.2mm    36.8mm    40mm