SENSYLINK Microelectronics

(CT1705)
Single-Wire Digital Temperature Sensor

CT1705 is a Low Cost Digital Temperature Sensor with ±0.5°C Accuracy with Single-wire Interface.

It is ideally used in General Temperature Monitor, White Appliance, Temperature Logger etc.

Table of Contents

Figures and Tables

±0.5°C Accuracy Digital Temperature Sensor with Single-wire Interface

Description

CT1705 is a low cost digital temperature sensor with ±0.5°C accuracy over -30°C to 50°C. Temperature data can be read out directly via Single-Wire interface by MCU.

It includes a high precision band-gap circuit, a 12-bit analog to digital converter that can offer 0.0625°C resolution, a calibration unit with non-volatile memory and a digital interface block.

The chip is calibrated for ±0.5°C(Max.) accuracy over -30°C to 50°C range in factory before shipment to customers.

Available Package: SOT-23 package

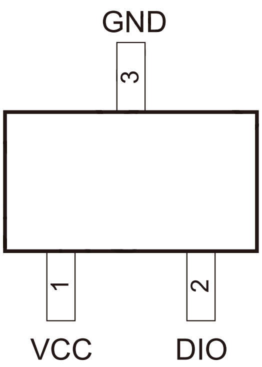
Features

- Operation Voltage: 2.2V to 5.5V
- Operating Current: 15uA(Typ.) during Temperature Conversion;
- Average Current Consumption: 0.55uA(Typ.) with reading once temperature per second
- Standby Current: 0.1uA(Typ.), 0.3uA(Max.)
- Temperature Conversion time: 35ms(Typ.)
- Temperature Accuracy:
  ±0.5°C(Max.) from -30°C to 50°C
- 12 bit ADC for 0.0625°C resolution
- Temperature Range: -40°C to 125°C

Applications

- General Temperature Monitor
- White Appliances
- Temperature Logger

PIN Configurations (Top View)



SOT-23 (package code K)

Typical Application

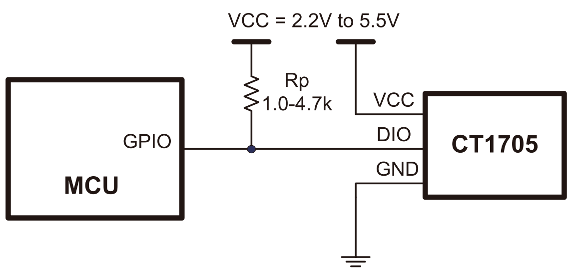


Figure 1. Typical Application of CT1705

Pin Description

PIN No.	PIN Name	Description
1	VCC	Power supply input pin, it should connect a 100nF to 1.0uF ceramic cap at least to ground.
2	DIO	Digital interface data input and output pin, Generally it needs a pull-up resistor to VCC in most applications, between 1.0k and 4.7k.
3	GND	Ground pin.

Function Block

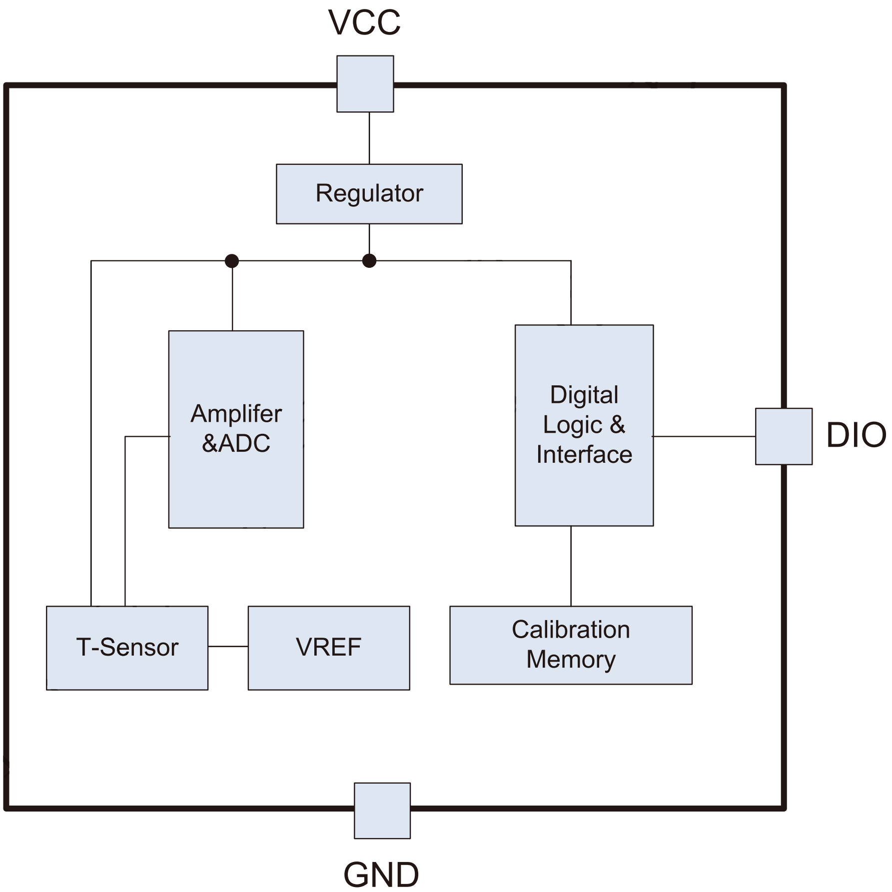


Figure 2. CT1705 function block

Ordering Information

CT1705 X X

Package Type
K : SOT-23

Packing
R : Tape & Reel

Order PN	Accuracy	Green[1]	Package	Marking ID[2]	Packing	MPQ	Operation Temperature
CT1705KR	±0.5°C	Halogen free	SOT-23	ADWW	Tape&Reel	3,000	-40°C~+125°C

Notes

Absolute Maximum Ratings (Note 3)

Parameter	Symbol	Value	Unit
Supply Voltage	V_{CC} to GND	-0.3 to 5.5	V
I/O pin Voltage	V_{IO} to GND	-0.3 to 5.5	V
Operation junction temperature	T_J	-50 to 150	ºC
Storage temperature Range	T_{STG}	-65 to 150	ºC
Lead Temperature (Soldering, 10 Seconds)	T_{LEAD}	260	ºC
ESD MM	ESD_{MM}	600	V
ESD HBM	ESD_{HBM}	6000	V
ESD CDM	ESD_{CDM}	1000	V

Note 3

1. *Stresses greater than those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at the "Absolute Maximum Ratings" conditions or any other conditions beyond those indicated under "Recommended Operating Conditions" is not recommended. Exposure to "Absolute Maximum Ratings" for extended periods may affect device reliability.*
2. *Using 2oz dual layer (Top, Bottom) FR4 PCB with 4x4 mm^2 cooper as thermal PAD*

Recommended Operating Conditions

Parameter	Symbol	Value	Unit
Supply Voltage	V_{CC}	2.2 ~ 5.5	V
Ambient Operation Temperature Range	T_A	-40 ~ +125	°C

Electrical Characteristics (Note 4)

Test Conditions: V_{CC} = 3.3V, T_A= -40 to 125°C unless otherwise specified. All limits are 100% tested at T_A=25°C.

Parameter	Symbol	Test Conditions	Min	Typ	Max	Unit
Supply Voltage	V_{CC}		2.2		5.5	V
Temperature Accuracy	T_{AC}	T_A = -30°C to 50°C	-0.5		0.5	°C
		T_A = -40°C to 125°C	-1.2		1.2	°C
Temperature Resolution		12-bit ADC		0.0625		°C
Operating Current	I_{OC}	during Temperature conversion		15		uA
Shutdown Current	$I_{SHUTDOWN}$	Idle, not temperature conversion		0.1	0.3	uA
Average Operating Current	I_{AOC}	1 time reading temperature per second		0.55		uA
Conversion time	t_{CON}	From active to finish completely		35		ms
Digital Interface						
Logic Input Capacitance	C_{IL}	I/O pin		20		pF
Logic Input High Voltage	V_{IH}	I/O pin	0.7*VCC		VCC	V
Logic Input Low Voltage	V_{IL}	I/O pin	0		0.2*VCC	V
Logic Input Current	I_{INL}	I/O pin	-2.0		2.0	uA
Communication Timing						
Single-Wire Communication Clock	T_{CLK}			12		us
Recovery time	t_{REC}		3.0			us
Time slot for "0" or "1"	t_{SLOT}		4*T_{CLK}+ t_{REC}			us
Initialization Low Time	t_{INIT}			32*T_{CLK}		us
Initialization Low Response Time	t_{PDL}			8* T_{CLK}		us
Initialization Response Sampling Time	t_{HSP}		2*T_{CLK}		10*T_{CLK}	us
Write '0' Low Time'	t_{W0L}		4*T_{CLK}		8*T_{CLK}	us
Write '1' Low Time'	t_{W1L}		2.0		1*T_{CLK}	us
Read bit Low Time	t_{RL}		2.5		1*T_{CLK}	us
Read bit sampling Time	t_{HSR}		t_{RL}		2*T_{CLK}	us

Part 1	Part 2	Part 3
Initialization the slave device, pull-low DIO pin with 450us to 650us duration time.	Send below Command Code respectively with LSB first, 1). 0xCC, [Device Ready]; 2). 0x44, [Temperature Conversion]; 3). Pull up DIO, and waiting for 35ms at least until the chip finish temperature conversion. 4). 0xBE, [Read Temperature];	Then the chip will output 2 bytes temperature data, 1). LSB byte in first; 2). MSB byte secondly.

Figure 3. CT1705 Communication Protocol Operation Diagram

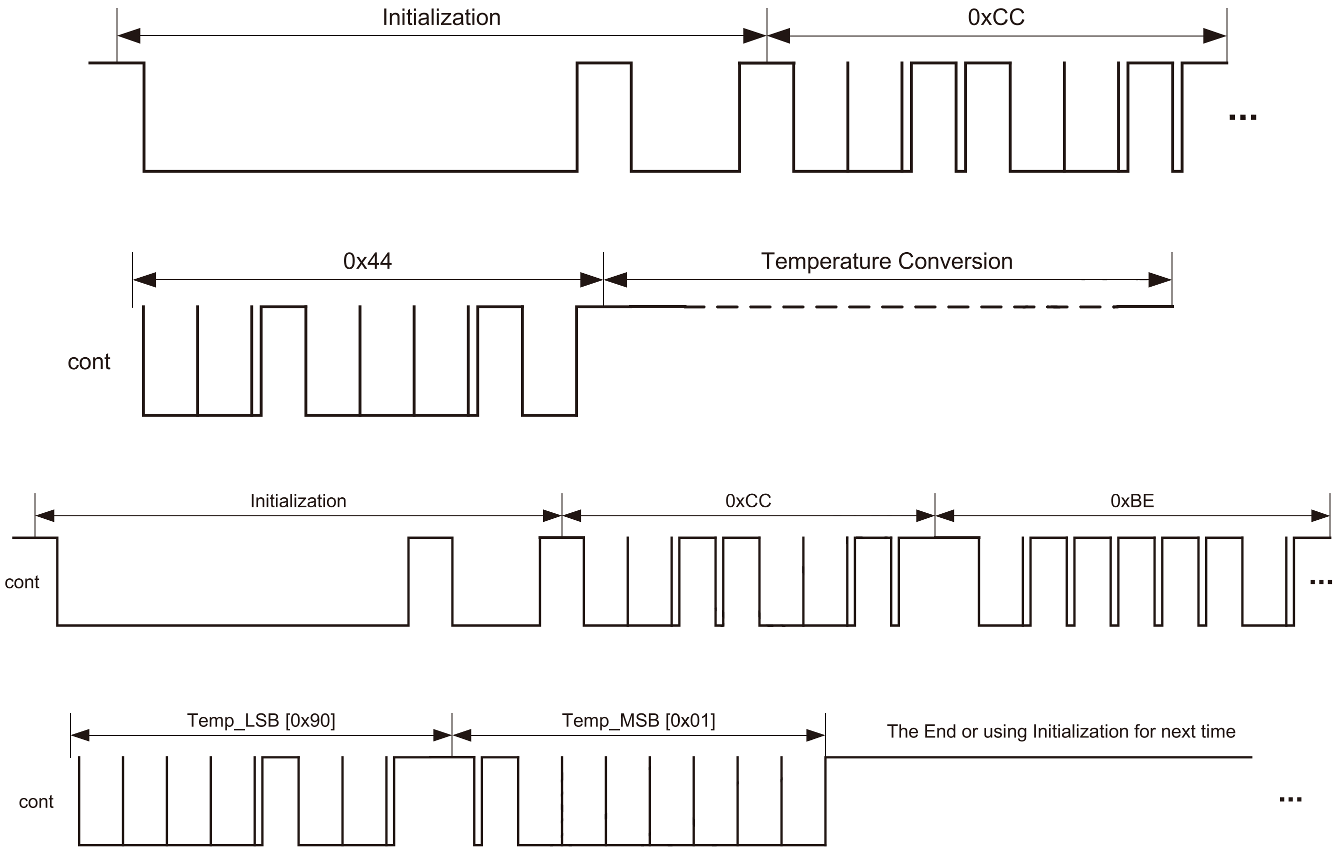


Figure 4. Read Temperature Reference Diagram [Temperature Data is 0x90, 0x01, means 25°C]

Note:

1.During initialization, MCU has to pull-low DIO pin with time range: 450us to 650us;

2. For temperature conversion, it will spends 35ms time in typical, user can check the finish FLAG by monitoring DIO pin every certain time, like 5ms, or just wait for 40ms.

1 Function Descriptions

The chip can sense temperature and convert it into digital data by a 12-bit ADC. Single-Wire interface's protocol shown in Figure-3. Generally, one complete communication with host or reading temperature by host, like MCU, include Part1, Part2 and Part3. And the time diagram is shown as below Figure-4. In general, user can obtain the temperature by following below example operation procedure.

Function	Data transmission direction (host)	Data on line (LSB first)	Comments
Step 1, Initialization & temperature conversion			
Initialization	Tx	low pulse with 450us to 650us duration.	The host generate valid initialization low pulse
	Rx	Low presence pulse	The device response this by presenting low pulse
Force the device do temperature conversion	Tx	0xCC	Make sure the chip is ready for Temp conversion.
	Tx	0x44	Temperature conversion
	Rx	Finish checking	Host read the status bit is '0 ', then the temperature conversion has not been completed; If '1', then the temperature conversion is complete. This step can also wait the required time of the temperature conversion (>35ms in Typ.).
Step 2, read temperature data from single-wire			
Initialization	Tx	low pulse with 450us - 650us duration.	The host generate valid low pulse
	Rx	Low presence pulse	The device response this by presenting low pulse
Read temperature data	Tx	0xCC	Make sure the chip is ready for reading temperature.
	Tx	0xBE	Reading Temperature Data command
	Rx	2 bytes, LSB in first, then MSB.	The chip sends 2 bytes of temperature data, with LSB first, then MSB.
		0xFF/Initialization Again	Once the chip finished sending data [0xFF], the chip will enter into standby mode.
The ends			

1.1 Digital Temperature Data

The major function of the chip is to measure temperature. The A-to-D converter resolution of the sensor is 12 bit, corresponding to 0.0625°C resolution. The CT1705 powers up in a low-power idle state. To initiate a temperature measurement and A-to-D conversion, the host has to issue a Temperature Conversion command [0x44h]. After the conversion, the temperature data is stored in the 2-byte temperature register in the registers [0x00, 0x01], and then the chip returns to idle state. The temperature data is stored in the temperature register as a 16-bit sign-extended two's complement format in degrees Celsius. The sign bits(S) indicate if the temperature is positive or negative: for positive numbers S = 0 and for negative numbers S = 1. Table 1 and Table 2 show examples of digital output data and the corresponding temperature (^{o}C). The default temperature data is 85oC after power-on, and LSB byte is 0x50 and MSB byte is 0x05.

Table 1. 12-bit Temperature Data

Temperature (°C)	12-bit Digital Output (HEX)	12-bit Digital Output (BIN)
+125.0000	0x07D0	0 0 0 0, 0 1 1 1, 1 1 0 1, 0 0 0 0
+85.0000	0x0550	0 0 0 0, 0 1 0 1, 0 1 0 1, 0 0 0 0
+25.0625	0x0191	0 0 0 0, 0 0 0 1, 1 0 0 1, 0 0 0 1

+10.1250	0x00A2	0 0 0 0, 0 0 0 0, 1 0 1 0, 0 0 1 0
+0.5000	0x0008	0 0 0 0, 0 0 0 0, 0 0 0 0, 1 0 0 0
0.0000	0x0000	0 0 0 0, 0 0 0 0, 0 0 0 0, 0 0 0 0
-0.5000	0xFFF8	1 1 1 1, 1 1 1 1, 1 1 1 1, 1 0 0 0
-10.1250	0xFF5E	1 1 1 1, 1 1 1 1, 0 1 0 1, 1 1 1 0
-25.0625	0xFE6F	1 1 1 1, 1 1 1 0, 0 1 1 0, 1 1 1 1
-55.0000	0xFC90	1 1 1 1, 1 1 0 0, 1 0 0 1, 0 0 0 0

Table 2. Temperature Data in Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LSB	2^{3}	2^{2}	2^{1}	2^{0}	2^{-1}	2^{-2}	2^{-3}	2^{-4}
MSB	S	S	S	S	S	2^{6}	2^{5}	2^{4}

1.2 Device Ready Command [0xCC]

After the host detects a presence pulse, it can issue a Device Ready command. The command code length is 8-Bit. Before perform Temperature conversion command [0x44] is issued, the host must submit this command. Please note, if user issue a Device Ready command followed by a Read temperature [0xBE] command, the temperature data will be output at single-wire.

1.3 Temperature Conversion [0x44]

This command forces the chip to do temperature conversion. After the conversion is complete, the measured temperature data will be stored into the register temporary. CT1705 then returns to a low-power idle state. The host monitors the conversion process in each time slot. When the host reads the logic '1 'instead of '0', it indicates the temperature conversion is complete.

1.4 Read Temperature Command [0xBE]

This command allows the host to read the temperature data measured in last time. Data transmission always starts from LSB byte in first, the MSB byte. Also the host can send a initialization low pulse signal to end this reading operation. The detail timing diagram is shown as above Figure-4.

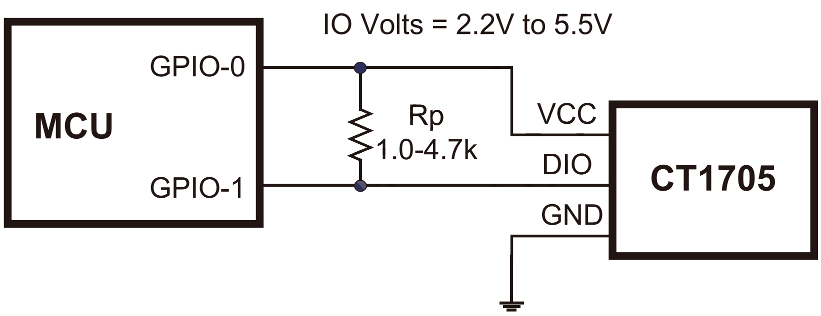
2 Single-Wire Communication Protocol

Single-Wire Protocol consists of a host and a salve device. In any case, CT1705 is used as slave device. The host could be a microcontroller or SoC. Discussion of Single-Wire Protocol is divided into three parts: the hardware configuration, the operation sequence and Single-Wire timing.

2.1 Hardware Configuration

According to the definition of Single-Wire Protocol, it has only one data line physically. In order to facilitate this, the slave device on the line needs to have open-drain or tri-state output, and CT1705's DIO pin uses an open-drain output. A typical circuit is shown in above Figure 1.

CT1705 supports fixed 15kbps (default rate) communication rate. For other communication rate, please contact Sensylink sales. Pull-up resistor depends primarily on the distance between the host and CT7310. For example with the communication distance of less than 20cm, a single node and an independent power supply condition, CT1705 requires an external 4.7kΩ (typical) pull-up resistor. If the communication distance is greater than 30m, you need a 1.0k or smaller pull-up resistor. Single-Wire line in idle state is high. It is ok to use another IO pin (GPIO-0) of host instead of VDD, shown as below. Both GPIO pins can be set low once finished temperature conversion & reading, which can saved power consumption further.



2.2 Operation Sequence

To access CT1705 through Single-Wire Line (DIO pin), the complete procedure is shown as previous Figure 3 and Figure 4, it includes:

- Part 1, Initialization.
- Part 2, Device Ready command, Temperature Conversion command and Read temperature command.
- Part3, Data Receiving/Transmitting, includes receive data from Single-Wire device or send data to Singe-Wire device.

2.3 Initialization

All operations of Single-Wire line always begins with a Initialization. It consists of a low pulse sent by the host and a device responses pulse shown in below Figure. The presence pulse is used to notify the host that CT1705 is already connected on the line. When the slave device sends a response pulse to the host, it tells the host that it is on the line and ready to work. During the initialization process, host pulls the line low for t_{INIT} period time (450us to 650us), thus produces (Tx) Initialization pulse. Then, the host releases the line and goes into receive mode (Rx). When the line is released, the line is pulled up by an external pull-up resistor. Single-Wire device operating clock T is set in production. When a Single-Wire device detects a rising edge, it will remain high for t_{PDH} (2T in typical), then the Single-Wire device generates a presence pulse by pulling the line low for t_{PDL} (8T in typical). After that the line is released and pulled back high by the external pull-up resistor, at least keeping the 6T time. Thus, the entire Single-Wire device response cycle is at least t_{PD} (16T in typical). After that, the host can begin to perform Device Ready command [0xCC]. If user needs more precise communication time match, the host can measure the Single-Wire device response t_{PDL} (8T in typical) low pulse, and adjust the time of the original Initialization pulse, t_{INIT}, and the read sampling timing. Once the device successfully captures the initialization low pulse, it will use it to set the communication speed. Single-Wire communication rate is fixed and NOT be change.

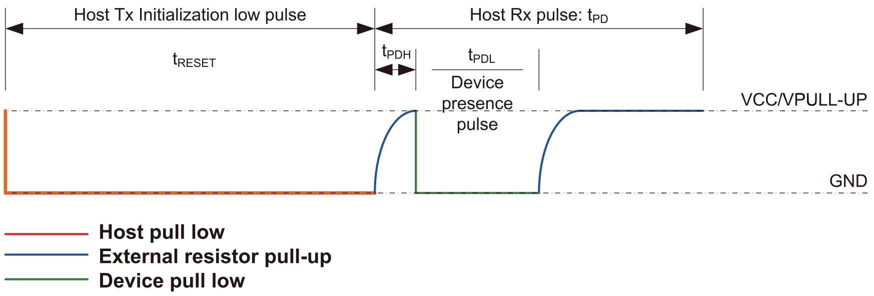


Figure 5 Initialization Timing Diagram

2.4 Single-Wire Timing

After complete initialization successfully on Single-Wire line, the next step is to perform Device ready command. The following section is to descript the bit transmission. The protocol defines several signal types: initialization pulse, presence pulse, write “0”, write “1”, read “0”, and read “1”. All these signals are synchronous signals issued by the host. And all the commands and data are the low byte first. For each byte data transmission, it is always LSB first.

During Write Time Slot the host writes data to a the slave device; and during the Read Time Slot, the host reads the data from the slave device.

Write Time Slot

There are two write time slot modes: write "1" and write "0" slot. The host writes into Single-Wire device "1" by using a write "1" slot, and host write into Single-Wire device "0" by using write "0". All write time slots are at least t_{SLOT} (4*T + t_{REC} in typical), and need the recovery time at least 3μs between two separate time slots. Two kinds of write slots start with pull-down line by the host shown in below Figure 6. To produce a write "1" slot, the host must release the line within t_{W1L} (<= 1*T) after pulling down for 1us, and then the line is pulled-up by an external pull-up resistors on the line. To produce a write "0" slot, after the host is pulling the line low, it maintains a low level during the entire time slot, that is t_{W0L} (> = 4T). During the write time slots, Single-Wire device samples line level status at t_{SSR} (2*T in typical) time. If sampling results at this time is high, then the logic "1" is written to the device; If "0", the write logic is "0".

Read Time Slot

Single-Wire device can only transmit data to the host after the host issues read time slots. After the host issues a read temperature command, a read time slot must be generated in order to read data from the Single-Wire device. A complete read time slot is at least t_{SLOT} (4*T + t_{REC}), and requires at least 3us recovery time between two separate time slots. Each time slot is generated by the host to initiate the read bit, a low level period is required to be at least 1μs shown in Figure 6. Once the device detects a Single-Wire line low, the device immediately sent bit "0" or "1" on the line. If Single-Wire device sends "1", the line is pulled-up high by a pull-up resistor after the short low period; if sent "0", then the line is keeping low for t_{DRV} (2*T in typical). After that the device releases the line from pull-up resistors and back to idle high. Therefore, the data issued by Single-Wire device after read time slot at the beginning stay effective during time t_{DRV} (2*T in typical). During the read time slots the host must release the line, and samples the line states at 2T after the start of a slot (optimum sampling time point 1T).

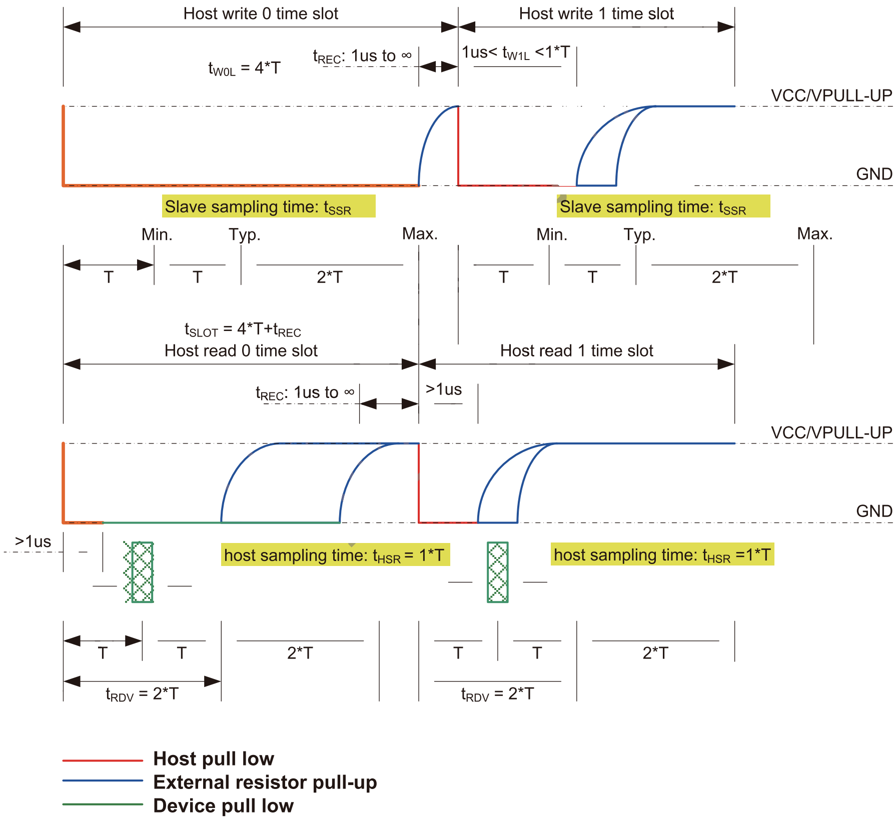
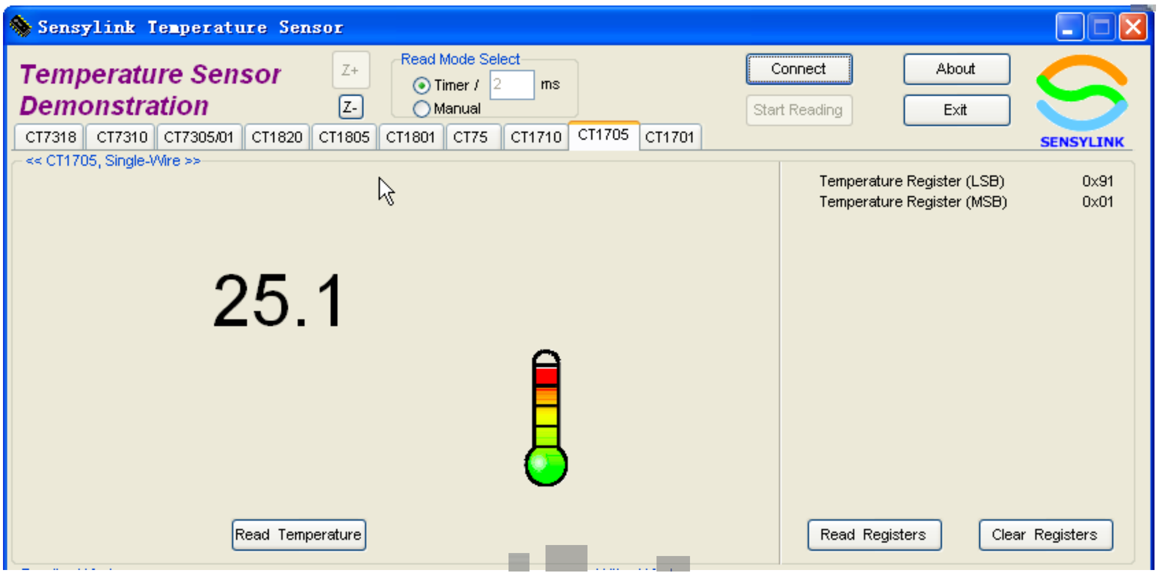


Figure 6 Read/Write Timing Slot Diagram

3 Software Reference Code

Below are windows based GUI of demo application for Sensylink temperature sensor, select CT1705 page, there are 2 Buttons. Press [connect] button, then press [Read Temperature] button, temperature data will be displayed, as shown on the right side. Below lists reference software code based on C++ language.



<u>For more information about software source code support, please contact our sales.</u>

```
/**********************************************/
uchar CT1705Init()
{
    uchar i;
    DIOPORT = 0;          //pull-low line
    delay600us();          //delay 450us to 650us
    DIOPORT = 1;         //pull-high line
    i = 0;  //
    while(DIOPORT)   //waiting for CT1705 pull-low line, once CT1705 give response.
    {
        delay500us();
        i++;
        if(i>1)//if waiting time > 5ms
        {
            return 0;//Return 0, initialization fail
        }
    }
    return 1;// Return 1, initialization success
}

void CT1705_Write_Byte(uchar dat)
{
    uint j;
    for(j=0; j<8; j++)
    {
        DIOPORT = 0;          //pull-low line with 1us
        i++;
        delay7us();
        DIOPORT = dat & 0x01;  //write one-bit data with LSB in first
        delay50us();
        DIOPORT = 1;   //release single-wire line, to be ready for next byte
        dat >>= 1;
    }
}

uchar CT1705_Read_Byte()
{
    uchar byte, bi;
    uint i, j;
    for(j=8; j>0; j--)
    {
        DIOPORT = 0;//pull-low line with 1us
        DIOPORT = 1;//then release line
```

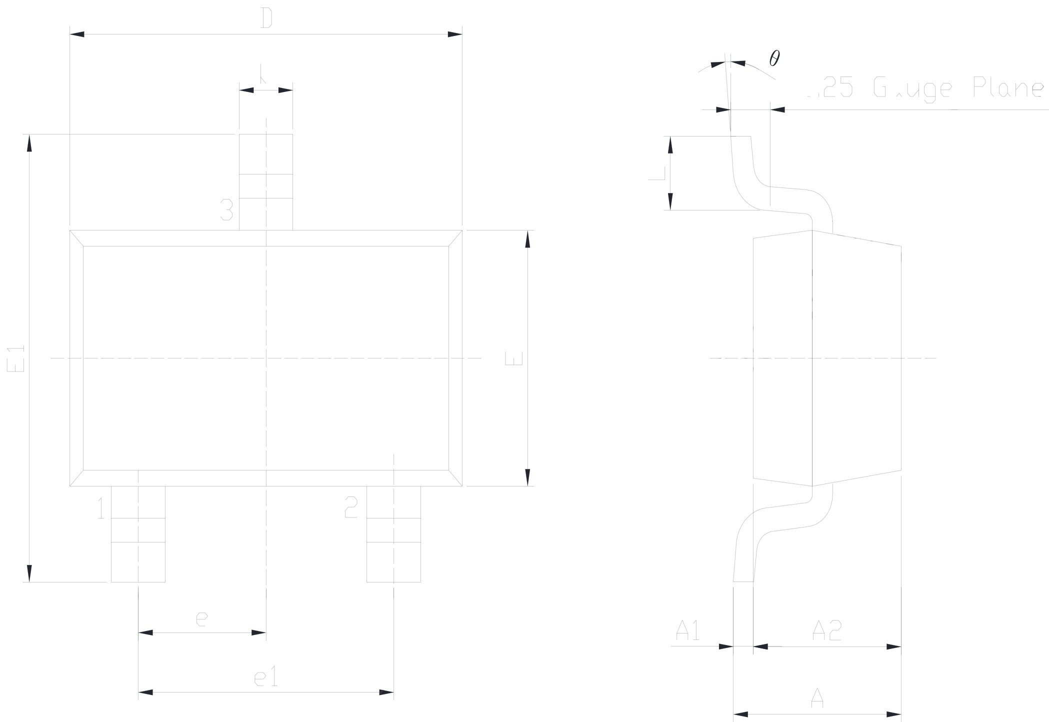
```
            bi = DIOPORT;     //Read Data from line, LSB in first
            /*move byte 1-bit to right, & move bi 7-bit to left*/
            byte = (byte >> 1) | (bi << 7);
            delay48us();
        }
        return byte;
    }
    void  CT1705_Temp_Conv()
    {
        CT1705Init();
        CT1705_Write_Byte(0xcc);      // make the chip ready command
        delay50us();
        CT1705_Write_Byte(0x44);      // Temp converter command
        delay50ms();
        }

    void  CT1705_Read_Temp_Com()
    {
        CT1705Init();
        delay200us();
        CT1705_Write_Byte(0xcc);   // make the chip ready Command
        delay200us();
        CT1705_Write_Byte(0xbe);   // Read temperature command
    }

    int CT1705_Read_Temp_Degree()
    {
        int temp = 0;
        uchar tmh, tml;
        CT1705_Temp_Conv();           //Send Temp converter command, 0x44
        CT1705_Read_Temp_Com();      //Read Temp
        delay200us();
        tml = CT1705_Read_Byte();     //Read LSB for Temperature in first
        delay200us();
        tmh = CT1705_Read_Byte();     //Then read MSB, Reg1_T_MSB
        temp = tmh;
        temp <<= 8;
        temp |= tml;
        return temp;
    }
```

Package Outline Dimensions

SOT-23 Unit (mm)



Symbol	Dimensions in Millimeters		Dimensions in Inches	
	Min.	Max.	Min.	Max.
A	0.900	1.150	0.035	0.045
A1	0.000	0.100	0.000	0.004
A2	0.900	1.050	0.035	0.041
b	0.300	0.500	0.012	0.020
c	0.080	0.200	0.003	0.008
D	2.800	3.000	0.110	0.118
E	1.200	1.400	0.047	0.055
E1	2.200	2.600	0.087	0.102
e	0.950（BSC）		0.037（BSC）	
e1	1.800	2.000	0.071	0.079
L	0.300	0.600	0.012	0.024
θ	0°	8°	0°	8°

SENSYLINK

SENSYLINK Microelectronics Co.,Ltd

www.sensylink.com

IMPORTANT NOTICE

SENSYLINK Microelectronics Co., Ltd reserves the right to make modifications, enhancements, improvements, corrections or other changes without further notice to this document and any product described herein or to discontinue any product or service. Customers should obtain the latest relevant information before placing orders and should verify the latest and complete information. SENSYLINK Microelectronics does not assume any responsibility for use of any product, nor does SENSYLINK Microelectronics any liability arising out of the application or use of this document or any product or circuit described herein. SENSYLINK Microelectronics assumes no liability for applications assistance or the design of Customers' products. Customers are responsible for their products and applications using SENSYLINK Microelectronics components. SENSYLINK Microelectronics does not convey any license under its patent or trademark rights nor the other rights.