CMOS LIS of Real Time Watch with Serial Interface

IN1363

FEATURES

- Count of seconds, minutes, hours, week days, date, months and years with consideration of leap years (until 2100);
- 400 kHz, double wire serial interface;
- Programmed orthogonal output signal;
- Function programming of alarm, timer and interruption;
- Automatic determination of the supply voltage drop;
- Consumption current of less, than 450 nA with supply of 2V with the operating oscillator;
- Operating temperature range: -40°C – +85°C.

SOP- 8

TA = - 40 ... + 85 °C
for all packages

ORDERING INFORMATION

Device	Operating Temperature Range	Package	Shipping
IN1363DT	T_A = - 40 ... + 85 °C	SOP-8	Tape & Reel
IN1363D		SOP-8	Tube

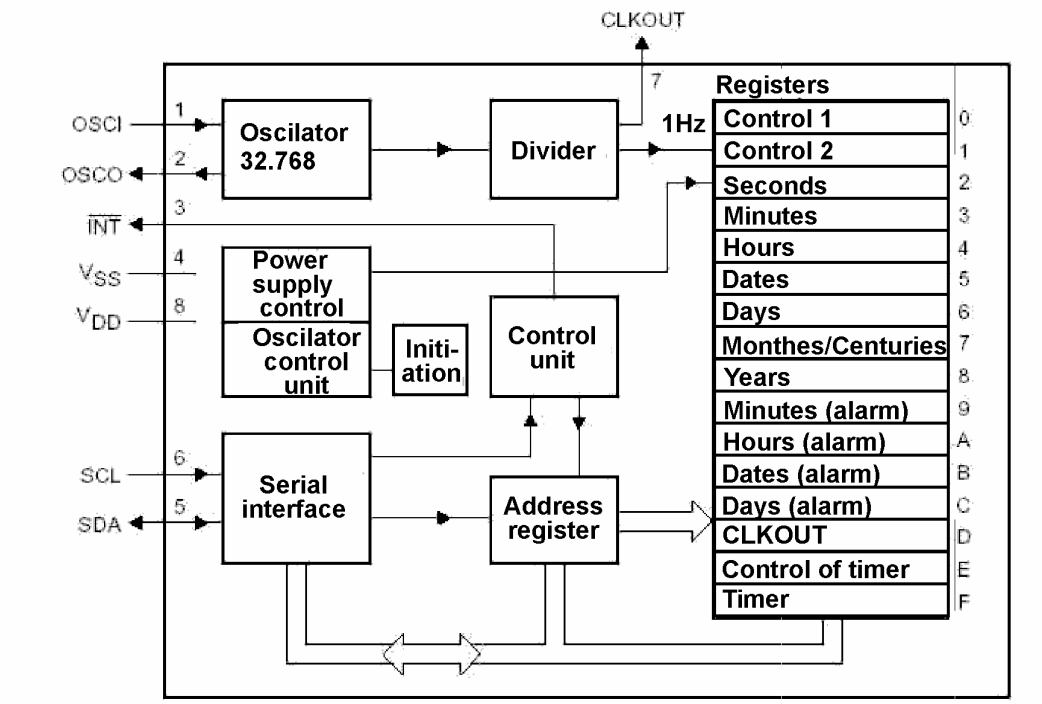
DESCRIPTION

Microcircuit IN1363 is essentially the complete binary-decimal digital watch with calendar, alarm, timer and possesses low power consumption. Addresses and data are transferred in series via the double wire bi-directional bus. The microcircuit is intended for count of real time in hours, minutes and seconds, count of week days, date, month and year. The last day of the month is automatically adjusted for the months with fewer, than 31 days, including correction for the leap year. The watch functions in the 24 hour mode. The microcircuit IN1363 has the built-in power control circuit, which determines the power level < 1V and forms the bit, signaling, that information about the real time may not be correct.

PINS DESCRIPTION

Pin Number	Symbol	Description
01	OSCI	Pin for connection of the quartz resonator
02	OSCO	Pin for connection of the quartz resonator
03	INT	Interruption output
04	V_{SS}	Common pin
05	SDA	Input / Output of data
06	SCL	Synchrosignal input
07	CLKOUT	Frequency divider output
08	V_{DD}	Supply source pin

BLOCK DIAGRAM



ASOLUTE MAXIMUM RATING

Limit and limit permissible operating modes of the microcircuit IN1363 are listed in the table

Characteristics	Symbol	Limit Permissible		Limit		Unit
		Min	Max	Min	Max	
Supply voltage	V_{DD}	1.0	5.5	–0.5	6.5	V
Dissipated power	P_{tot}	-	-	-	300	mWt
Input voltage SCL, SDA, OSCI	V_I	0	5.5	–0.5	6.5	V
Output voltage CLKOUT,INT	V_O	0	5.5	–0.5	6.5	V
Direct input or output current via any pin	I_{IO}	-	-	–10	10	mA

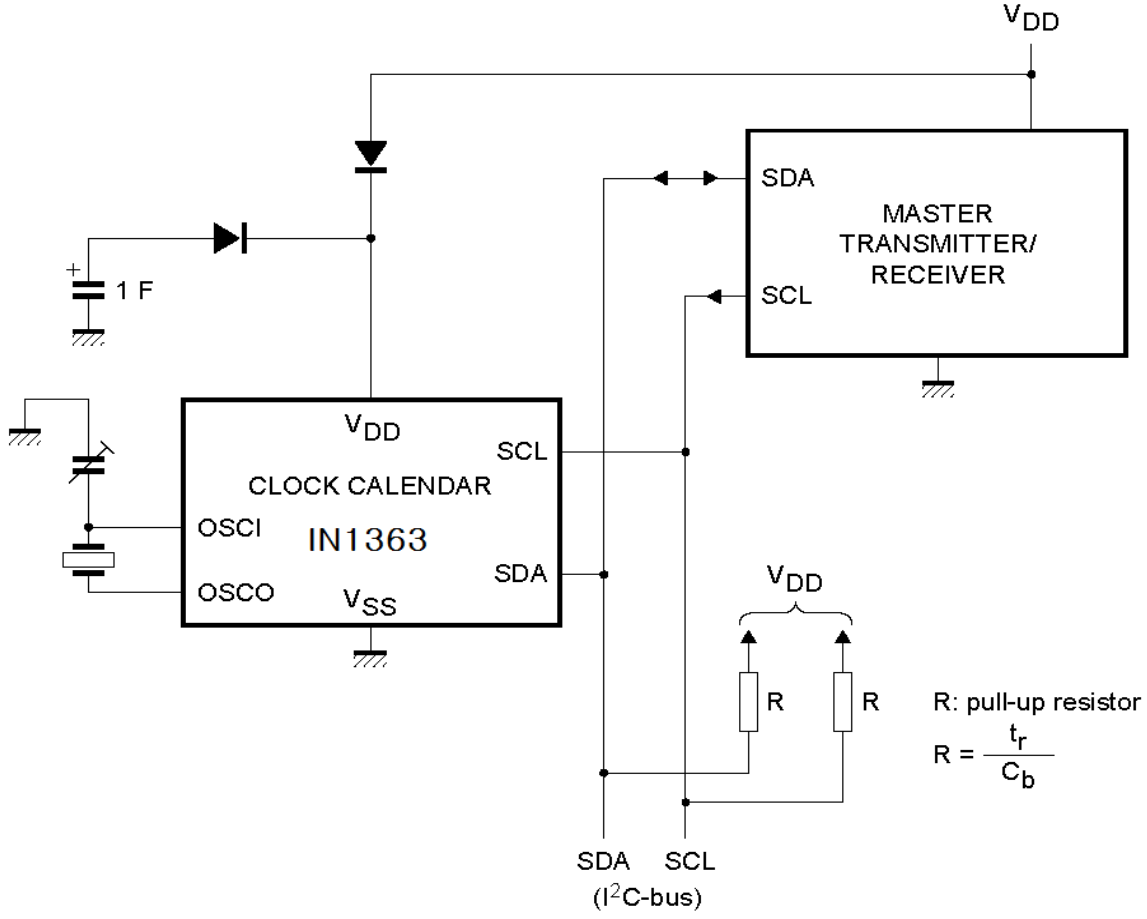
All voltages are listed relative to ground. Under influence of the limit mode serviceability of the microcircuits is not guaranteed. After plotting the limit mode serviceability is guaranteed in the limit permissible mode.

* Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device.

These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied.

Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Application



ELECTRICAL CHARACTERISTICS

(T_A = –40...+ 85°C, V_{CC} = 4,5 – 5,5 V)

Characteristics	Symbol	Test Condition	Min	Max	Note	Unit
Supply voltage	V_{DD}	I2C bus –active; F_{SCL} = 400 kHz	1.8	5.5		V
		in the non-active mode	1.0	5.5	1,2	
Input leakage current	I_{LI}	V_{IN} =V_{DD}; V_{IN} =V_{SS}	-	1		uA
Output leakage current	I_{LO}	V_{OUT} =V_{DD}; V_{OUT} =V_{SS}	-	1		uA
Consumption current	I_{DD1}	CLKOUT–off, F_{SCL} =400 kHz	-	800		uA
		CLKOUT–off, F_{SCL} =100 kHz	-	200		
		CLKOUT–off, F_{SCL} =0 kHz, V_{DD}=5V	-	0.55	1,2	
		CLKOUT–off, F_{SCL} =0 kHz, V_{DD}=2V	-	0.45	1,2	
Low level input voltage	V_{IL}		V_{SS}	0.3V_{DD}		V
High level input voltage	V_{IH}		0.7V_{DD}	V_{DD}		V
Low level output current at pin CLKOUT	I_{OL1}	V_{OL} = 0.4 V, V_{DD} =5 V	1	-		mA
High level output current at pin CLKOUT	I_{OH1}	V_{OH} = 4.6 V, V_{DD} =5 V	1	-		mA
Low level output current at pin INT	I_{OL2}	V_{OL} = 0.4 V, V_{DD} =5 V	1	-		mA
Low level output current at pin SDA	I_{OL3}	VOL = 0.4 V, VDD =5 V	3	-		mA
Supply low level, detected by the circuit	V_{LOW}			1	1	V

Note

1 Ta=(25 ± 5)^{o}C

2 Parameters of the quartz oscillator: f_{OSC}=32.768 kHz, R_S≤40 kOhm, C_L=8 пF

DYNAMIC CHARACTERISTICS

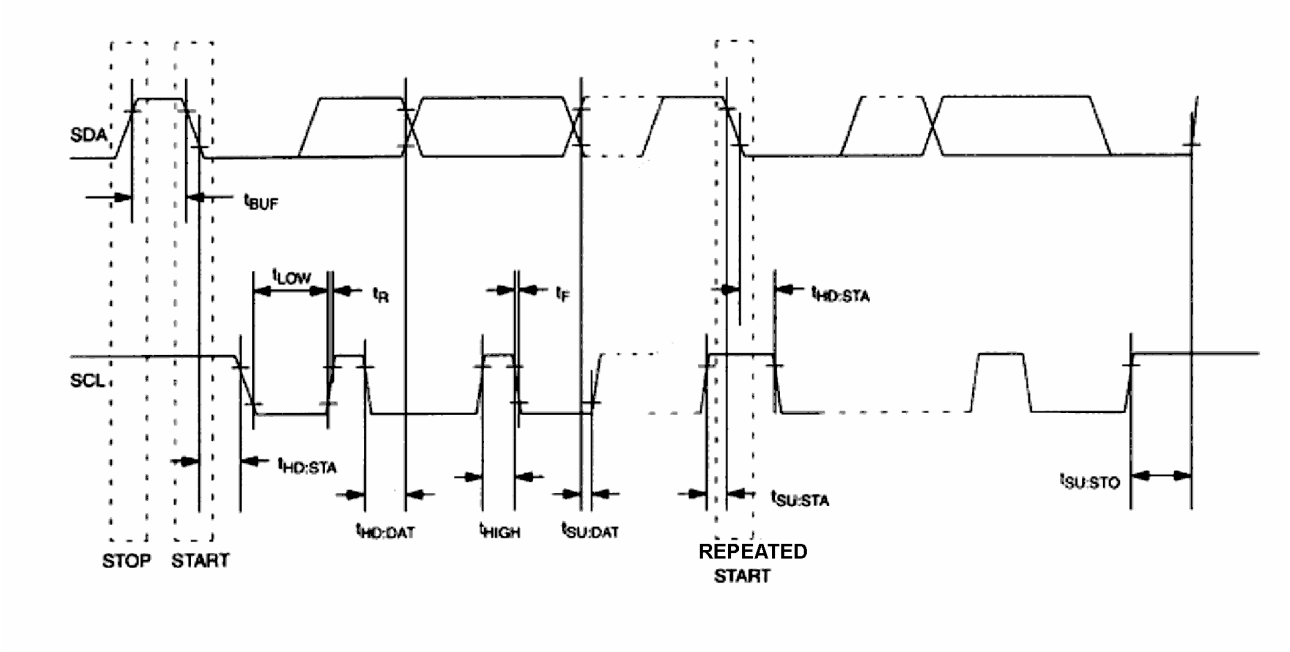
(T_A = –40...+ 85°C, V_{CC} = 4,5 – 5,5 V are listed in the table)

Characteristics	Symbol	Test Condition	Min	Max	Unit
Cycle frequency SCL	f_{SCL}	–	0	400	kHz
Time of bus vacant condition between the conditions STOP and START	t_{BUF}	–	4,7	–	us
Hold time (repeated) of the condition START	$t_{HD:STA}$ [1)]	–	0,6	–	us
Low condition duration of the cycle pulse SCL	t_{LOW}	–	1,3	–	us
High condition duration of the cycle pulse SCL	t_{HIGH}	–	0,6	–	us
Presetting time for the repeated condition START	$t_{SU:STA}$	–	0,6	–	us
Data hold time	$t_{HD:DAT}$ [2)]	–	0	–	us
Data presetting time	$t_{SU:DAT}$	–	100	–	ns
Rise time for signals SDA and SCL	t_R	–	–	300	ns
Drop time for signals SDA and SCL	t_F	–	–	300	ns
Presetting time for the condition STOP	$t_{SU:STO}$	–	0,6	–	us
Total capacitance load on each bus line	C_B	–	–	400	pF
Capacity input/output	$C_{I/O}$	–	10	10	pF
Load capacitance of the quartz resonator	C_{LX}	–	12,5	12,5	pF

1) After this time interval the first cycle signal is formed;
2) The device should internally ensure the hold time, at least, 300 nsec for the signal SDA (relative to V_{IHMIN} of the signal SCL) in order to overlap the indeterminacy area of the signal SCL drop front

Maximum value $t_{HD:DAT}$ should be definite in that case, if the device does not increase duration of the low condition (t_{LOW}) of the signal SCL

TIMING DIAGRAMM



OPERATION DESCRIPTION

IN1363 operates as the «slave» device on the serial bus. For access to it expedient to set the condition START and transfer after the register address the device identification code.

The next registers can be address in series till the condition STOP is preset. With V_{CC} below 1,8 V, access granting to the device by the serial interface is not guaranteed.

The current time is counted with the supply voltage 1 ÷ 5,5 V. When the supply level becomes lower, than 1V, the bit VL=1 is formed, signaling, that the information about the current time may be incorrect.

DESCRIPTION OF SIGNALS

V_{DD}– Positive supply. V_{DD} – input from +1 till +5 V. With supply < 1,8 V access to the interface circuit is not guaranteed.

INT – interrupt output. Interruption condition is formed with coincidence of the current time with the alarm settings, or with attainment of the condition «0» of the timer countdown. Interruption, formed from the alarm, forms the continuous signal, and from the timer can be both continuous and pulse one.

SCL (Serial Clock Input) – SCL is used for synchronization of the data transfer by the serial interface.

SDA (Serial Data I/O) – SDA is input/output for the double wire serial interface. Output SDA is the open drain, for which the external load resistor is required to be connected.

CLKOUT (Former output of the orthogonal signal) – For output activation the bit FE is preset to “1”. CLKOUT generates the orthogonal signal of four different frequencies (1 Hz, 32 Hz, 1 kHz, 32 kHz). Output CLKOUT is essentially the open drain, for which the external load resistor is required to be connected.

OSCI, OSCO – connection of the standard quartz resonator for the frequency 32,768 kHz. Capacitance load of the internal oscillator for the quartz resonator is equal to 12pF. IN1363 can operate from the external oscillator with the frequency 32,768 kHz. With this configuration the output OSCI is connected to the signal external oscillator, and OSCO is left unconnected.

WATCH AND CALENDAR

Acquisition of information on time and date is performed by means of reading the appropriate register bytes. Presetting and time and calendar initialization is performed by means of the appropriate bytes. Information, contained in the time, calendar and alarm registers, is essentially the binary-decimal code. Bit 7 of register 2 is essentially the indication bit of the supply level decrease. < 1V (VL). When this bit = “1” , this signifies, that the supply voltage was below the norm, and the information on the current time may be unreliable.

When switching power supply on, all register bits are preset to "0", with the exception of bits FE, VL, TD1, TD0, TESTC and AE, which are preset to "1".

When applying the signal “START” on the double wire bus, the current time transfer occurs from the counters to the auxiliary set of registers. The data on time are read out from these auxiliary registers, while the watch continue to operate. This eliminates the necessity in the repeated reading in case of updating the basic registers in the access process.

REGISTERS RTC IN1363

Address	Data								Registers / Range	
	D7	D6	D5	D4	D3	D2	D1	D0		
00H	TEST1	0	STOP	0	TESTC	0	0	0	Control 1	
01H	0	0	0	TI/TP	AF	TF	AIE	TIE	Control 2	
02H	VL	Tens of seconds			Units of seconds				Seconds	00 – 59
03H	x	Tens of minutes			Units of minutes				Minutes	00 – 59
04H	x	x	Tens of hours		Units of hours				Hours	00 – 23
05H	x	x	Tens of date		Units of date				Dates	01 – 31
06H	x	x	x	x	x	Day of week			Day of week	0 – 6
07H	C	x	x	10 M.	Units of month				Century / month	0-1/01-12
08H	Tens of years				Units of years				Year	00 – 99
09H	AE	Tens of minutes			Units of minutes				Minute alarm	00 – 59
0AH	AE	x	Tens of hours		Units of hours				Hour alarm	00 – 23
0BH	AE	x	Tens of date		Units of date				Date alarm	01 – 31
0CH	AE	x	x	x	x	Day of week			Weekday alarm	0 – 6
0DH	FE	x	x	x	x	x	FD1	FD0	Control of CLKOUT	
0EH	TE	x	x	x	x	x	TD1	TD0	Control of timer	
0FH	Value of timer								Timer	

CONTROL REGISTERS

Control Register 1

Address 00H : Control / Status 1 register bits

Bit	7	6	5	4	3	2	1	0
Symbol	TEST1	0	STOP	0	TESTC	0	0	0

TEST1 (activation of test mode) – This bit, preset to logic “1”, activates the test mode, with logic “0” normal functioning of the circuit.

STOP – This bit, preset to logic “1” in the test mode perform the zero setting of all dividers, with logic “0” – normal functioning of the circuit.

TESTC (activation of the test mode) – This bit, preset to logic “1”, activates the test mode, with logic “0” normal functioning of the circuit.

When the control register1 is programmed, The OSC generates.

Control Register 2

Address 01H : Control / Status 2 register bits

Bit	7	6	5	4	3	2	1	0
Symbol	0	0	0	TI/TP	AF	TF	AIE	TIE

TI/TP (formation of the pulse interruption signal at output INT) – This bit, preset to logic “0”, with appearance of the timer flag TF at output INT forms the constant interruption signal of the low level. The bit, preset to logic “1” at output INT, forms the interruption pulse signal (signal frequencies are listed in the table).

Timer Input Frequency (Hz)	**Period INT (sec).[1]**	
	N =1 [2]	**N > 1**
4096	1/8192	1/4096
64	1/128	1/64
1	1/64	1/64
1/60	1/64	1/64

[1] TF and INT become active simultaneously.
[2] N – value, loaded to the timer register. Timer is stopped with N = 0.

AF (alarm flag) - bit, in logic “1” informs about interruption by actuation of the alarm, by means of software the bit AF can be reset only.

TF (timer flag) - bit, in logic “1” informs about interruption by actuation of the timer, by the software means the bit TF can be reset only.

AIE (activation of alarm) - bit, preset to logic “1”, activates operation of the alarm.

TIE (activation of timer) - bit, preset to logic “1”, activates operation of the alarm.

	AF		TF	
Read	0	Alarm flag inactive	0	Timer flag inactive
	1	Alarm flag inactive	1	Timer flag active
Write	0	Alarm flag inactive	0	Timer flag is cleared
	1	Alarm flag inactive	1	Timere flag remains unchanged

Control Register CLKOUT.

Address 0DH : CLKOUT frequency register bits

Bit	7	6	5	4	3	2	1	0
Symbol	FE	x	x	x	x	x	FD1	FD0

FE (output activation CLKOUT) - This bit, preset to logic “1”, activates output CLKOUT. Frequency of the output orthogonal signal is determined by the bits FD0 and FD1. The source clock for the timer is also selected by the Timer Control register.

Other timer properties,e.g interrupt generation are controlled via the Control./ststus 2 register. For accurate read back of the countdown value, the I2C- bus clock SCL must be operating at a frequency of at least twice the selected timer clock.

FD1	FD0	Frequency CLKOUT
0	0	32,768 kHz
0	1	8,192 kHz
1	0	4,096 kHz
1	1	1 Hz

Timer Control Register

The Timer register is an 8 –bit binary countdown timer . It is enabled and disabled via the Timer control register bit TE .

The source clock for the timer is also selected by theTimer control register. Other timer properties, e.g. interru pt generation, are controlled via the Control/status 2 register. For accurate read back of the countdown value, the i2c bus clock SCL must be operating at a frequency of at least twice the selected timer clock.

Address 0EH : Timer control register bits

Bit	7	6	5	4	3	2	1	0
Symbol	TE	x	x	x	x	x	TD1	TD0

TE (timer activation) - This bit, preset to logic “1”, activates the frequency application to the timer input from the oscillator. The signal frequency is determined by bits TD0 and TD1.

TD1	TD0	Timer Input Frequency
0	0	4096 Hz
0	1	64 Hz
1	0	1 Hz
1	1	1 / 60 Hz

Address 0FH : Timer countdown value register bits

Bit	7	6	5	4	3	2	1	0
Symbol								

Timer countdown value - This register holds the loaded countdown value ‘n’

$$\text{Countdown period} = \frac{n}{\text{Source clock frequency}}$$

Seconds, Minutes and Hours Register

Address 02H : Seconds / VL register bits

Bit	7	6	5	4	3	2	1	0
Symbol	VL	seconds						

VL = 0 : reliable clock/calendar information is guaranteed
VL = 1 : reliable clock/calendar information is no longer guaranteed.

<seconds> These bits represent the current seconds value coded in BCD format; value = 00 to 59.

Address 03H : Minutes register bits

Bit	7	6	5	4	3	2	1	0
Symbol	x	minutes						

<minutes> These bits represent the current minutes value coded in BCD format; value = 00 to 59.

Address 04H : Hours register bits

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	hours					

<hours> These bits represent the current hours value coded in BCD format; value = 00 to 23.

Days, weekdays,Months/Century and Years Register

Address 05H : Dates register bits

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	days					

<days> These bits represent the current day value coded in BCD format; value=01to 31

Address 06H : Weekdays register bits

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	Week day		

<week days> These bits represent the current week day value 0 to 6

Address 07H : Century / Months registers bits

Bit	7	6	5	4	3	2	1	0
Symbol	C	-	-	Months				

< C > Century bit.
C = 0 ; indicates the century is 20xx.
C = 1 : indicates the century is 19xx. 'xx' indicates the value held in the Years register;
This bit is toggled when the years register overflow from 99 to 00.

<months> These bits represent the current month value coded in BCD format value 0 to12

	B4	B3	B2	B1	B0
January	0	0	0	0	1
February	0	0	0	1	0
March	0	0	0	1	1
April	0	0	1	0	0
May	0	0	1	0	1
June	0	0	1	1	0
July	0	0	1	1	1
August	0	1	0	0	0
September	0	1	0	0	1
October	1	0	0	0	0
November	1	0	0	0	1
December	1	0	0	1	0

Address 08H : Years register bits

Bit	7	6	5	4	3	2	1	0
Symbol	years							

<years> This register represent the current year value coded in BCD format :00 to 99

Alarm Registers

Address 09H : Minute alarm register bits

Bit	7	6	5	4	3	2	1	0
Symbol	AE	Minute alarm						

AE - AE = 0; minute alarm is enabled.
AE = 1; minute alarm is disabled.

<minute alarm> These bits represents the Minute alarm information coded in BCD format;
value = 00 to 59.

Address 0AH : Hour alarm register bits

Bit	7	6	5	4	3	2	1	0
Symbol	AE	-	hour alarm					

AE - AE = 0; hour alarm is enabled.
AE = 1; hour alarm is disabled.

<hour alarm> These bits represents the hour alarm information coded in BCD format; value = 00 to 23.

Address 0BH : Day alarm register bits

Bit	7	6	5	4	3	2	1	0
Symbol	AE	-	day alarm					

AE - AE = 0; day alarm is enabled.
AE = 1; day alarm is disabled.

<day alarm> These bits represents the day alarm information coded in BCD format; value = 01 to 31

Address 0CH : Weekday alarm register bits

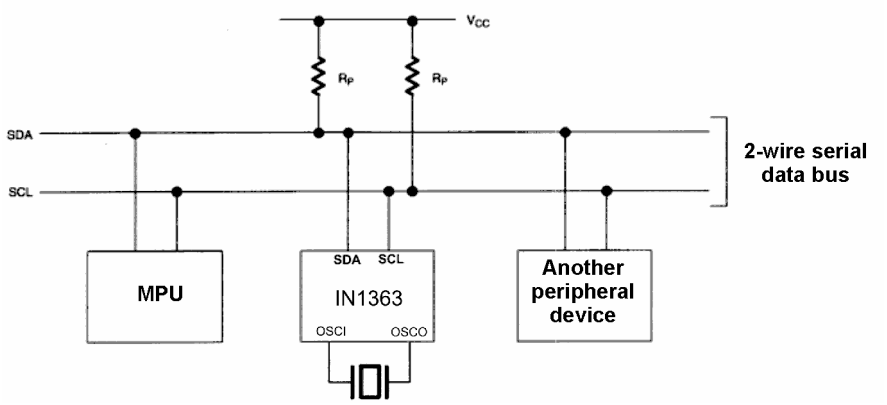
Bit	7	6	5	4	3	2	1	0
Symbol	AE	-	-	-	-	weekday alarm		

AE - AE = 0; weekday alarm is enabled.
AE = 1; weekday alarm is disabled.

< weekday alarm> These bits represents the weekday alarm information coded in BCD format;
value = 0 to 6

2-WIRE SERIAL DATA BUS

IN1363 supports the bi-directional double wire bus and the data transfer protocol. The bus can be controlled by the “master” device, which generates the cycle signal (SCL), controls access to the bus, generates the conditions START and STOP. Typical bus configuration with the double wire is indicated in the Figure.



Data transfer can be started only when the bus is not busy. In the process of the data transfer, the data line should remain stable, while the cycle signal line is in the HIGH condition. Alterations of the data line conditions at that moment, when the cycle line is in the high condition, will be regarded as the control signals.

In compliance with this the following conditions are determined:

Bus is not busy: both lines of data and cycle signal are in the HIGH condition.

Data transfer start: Alteration of the data line condition during transition from HIGH to LOW, while the cycle line is in the HIGH condition, is determined as the status START.

Data transfer stop: Alteration of the data line condition during transition from LOW to HIGH, while the cycle line is in the HIGH condition, is determined as status STOP.

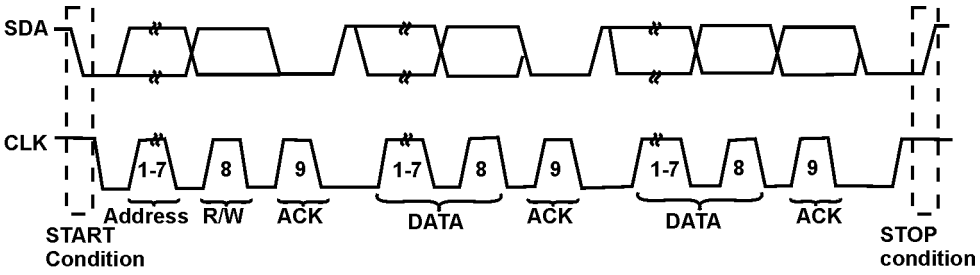
Valid data: Condition of the data line corresponds to the valid data, when after the condition START the data line is stable at the time of the HIGH status of the cycle signal. The data on the line should be altered at the time of the LOW condition of the cycle signal. One cycle pulse per one data bit.

Each data transfer starts with arrival of the status START and ceases with arrival of the status STOP. Number of data bytes, transferred between the statuses START and STOP, is not limited and is determined by the «master» device. Information is transferred byte by byte, and each reception is confirmed by the ninth bit.

Reception confirmation: Each receiving device, when being addressed, generates the reception confirmation bit after reception of each byte. The «master» device should generate the additional cycle pulses, which are set in compliance with the confirmation bits.

If the reception confirmation signal is in the high condition, then upon arrival of the confirmation cycle signal, confirming reception, the device should switch the SDA line to the low condition. Of course, the presetting time and the hold time should be taken into consideration.

The «master» device should signal about termination of the data transfer to the «slave» device, stopping generation of the confirmation bit, while receiving from the «slave» cycle pulse of the reception confirmation. In this case, the «slave» cycle pulse should switch the data line to the low condition for the «master» cycle pulse to generate the condition STOP.



Data transfer by the serial double wire bus

Depending on the status of the bit $R/\overline{W}$, two types of transfer are possible:

1. Data are transferred from the «master» transmitter to the «slave» receiver. The first byte, transferred by the «master» one, is the address of the «slave» one. Then follows sequence of the data bytes. The «slave» one returns the reception confirmation bits after each received byte. Order of the data transfer: the first one is the most senior digit (MSB).

2. Data are transferred from the «slave» transmitter to the «master» receiver. The first byte (address of «slave») is transferred to the «master» one. Then the «master» returns the confirmation bit. This follows after the «slave» one of the data sequence. The «master» one returns the reception confirmation bit after each received byte, with exception of the last byte. After reception of the last byte the reception confirmation bit does not return.

The «master» device generates all cycle pulse and the conditions START and STOP. Transmission completes with emergence of the condition STOP or the repeated emergence of the condition START. As the repeated condition START is the beginning of the next serial transmission, then the bus is not vacated. Data transfer order: the first one is the most senior digit (MSB).

IC IN1363 CAN WORK IN 2 NEXT MODE

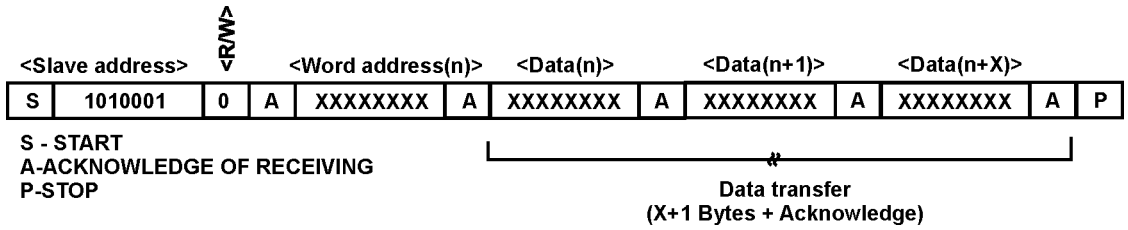
Mode of the «slave» receiver (writing mode of IN1363)

Serial data and cycles are received via SDA and SCL appropriately. After transfer of each byte the confirming bit is transferred.

Conditions START and STOP are understood as the start and end of the serial transmission. Address recognition is performed by the hardware means after reception of the address of the «slave» one and the direction bit.

The address byte is the first byte, received after emergence of the condition START, generated by the «master» one. Address byte contains seven address bits IN1363, equal to 1010001, accompanied by the direction bit ($R/\overline{W}$), which is equal to 0 for writing. After reception and decoding the address IN1363 provides confirmation on the line SDA. After confirmation by IN1363 of the «slave» address and the write bit, the «master» one transmits the register address of IN1363.

Thus, the register indicator will be set in IN1363. Then the «master» one will start to transfer each data byte with the subsequent confirmation reception of each byte receipt. Upon completion of writing the «master» one will form the condition STOP, for termination of the data transfer.



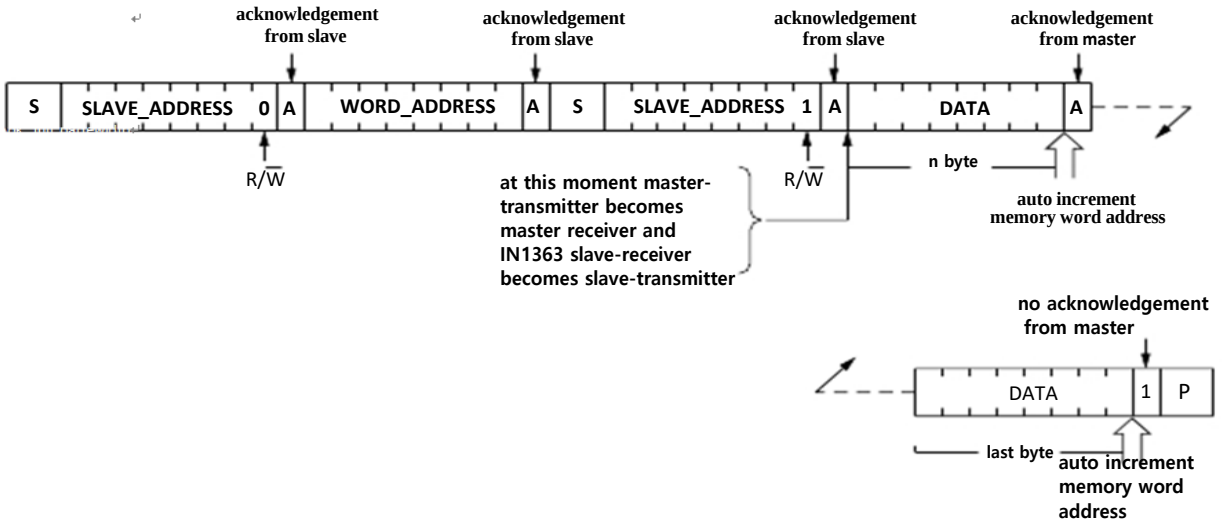
Data writing – mode of the «slave» receiver

Mode of the «slave» transmitter (read-out mode from IN1363)

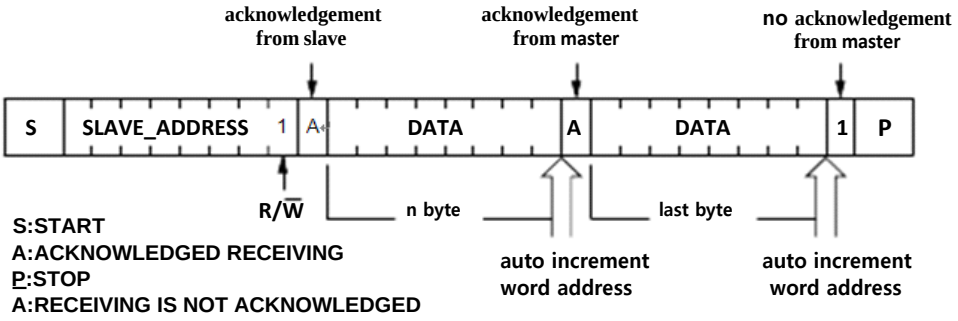
The first byte is accepted and is processed as in the mode of the «slave» receiver. But in this mode the direction bit will indicate, that the transmission direction is altered. The serial data are transferred by IN1363 by means of SDA, the cycle pulses – by means of SCL. The statuses of START and STOP are recognized as the start and end of transmission in series. The address byte is the first byte, received after emergence of the status START, generated by the «slave» one.

The address byte contains the seven address bits DS1363, equal to 1010001, accompanied with the direction bit ($R/\overline{W}$), which is equal to 1 for reading. After reception and decoding the address byte IN1363 accepts confirmation from the line SDA.

Then IN1363 starts to transmit the data from the address, to which the register indicator indicates. If the register indicator is not written prior to initialization of the writing mode, then the first read address will be the last address, stored in the register indicator. IN1363 should transmit the bit of «non-confirmation», in order to complete reading.



Master reads after setting word address (write word address ; read data)



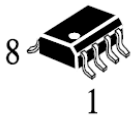
Master reads slave immediately after first byte (read mode)

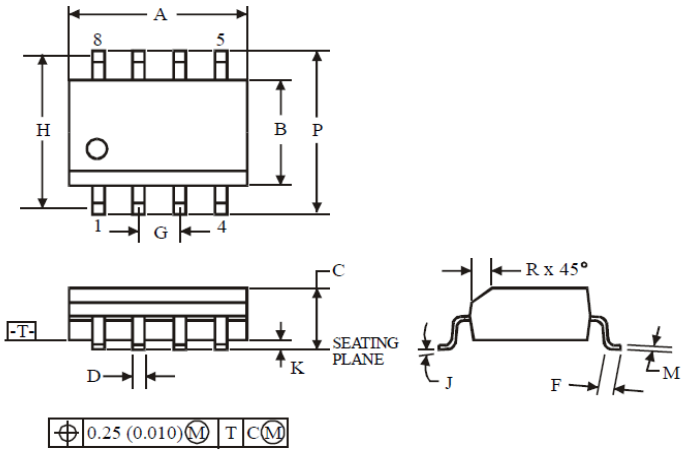
Data reading – mode of «slave» transmitter

PACKAGE DIMENSION

SOP 8 (MS-012AA)

D SUFFIX SOIC
(MS - 012AA)





	Dimension, mm	
Symbol	**MIN**	**MAX**
A	4.8	5
B	3.8	4
C	1.35	1.75
D	0.33	0.51
F	0.4	1.27
G	1.27	
H	5.72	
J	0°	8°
K	0.1	0.25
M	0.19	0.25
P	5.8	6.2
R	0.25	0.5

NOTES:

1. Dimensions A and B do not include mold flash or protrusion.
2. Maximum mold flash or protrusion 0.15 mm (0.006) per side for A; for B - 0.25 mm (0.010) per side.

TEST FIXTURE
for IN1363 & IN1307

Application Note

IK SEMICON

1. TEST FIXTURE OVERVIEW

In order to supply convenient method for RTC Test, IK-SEMICON prepared test fixture. It's controlled by Silabs MCU C8051F410, user can RTC test by using this fixture.

1.1. Installing the hardware

IN1363/IN1307 is an IC compatible real time clock (RTC).
As an application example, it demonstrates how to setup the RTC.
This application note explain how to set the register value into a date and time value that can be put in the following form [YY]:[MM]:[DD]:[HH]:[MM]:[SS].

1.2. Major Components

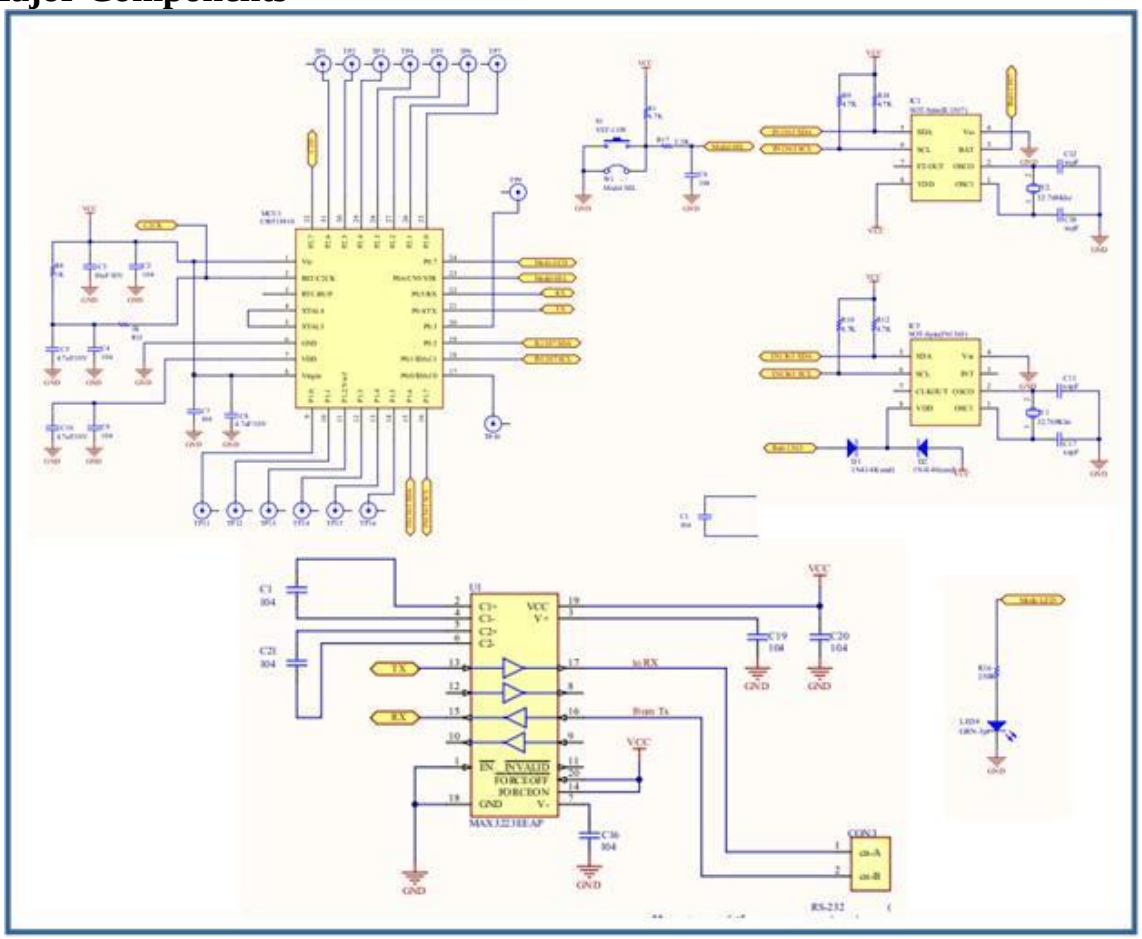


Fig 1. Test board Schematic

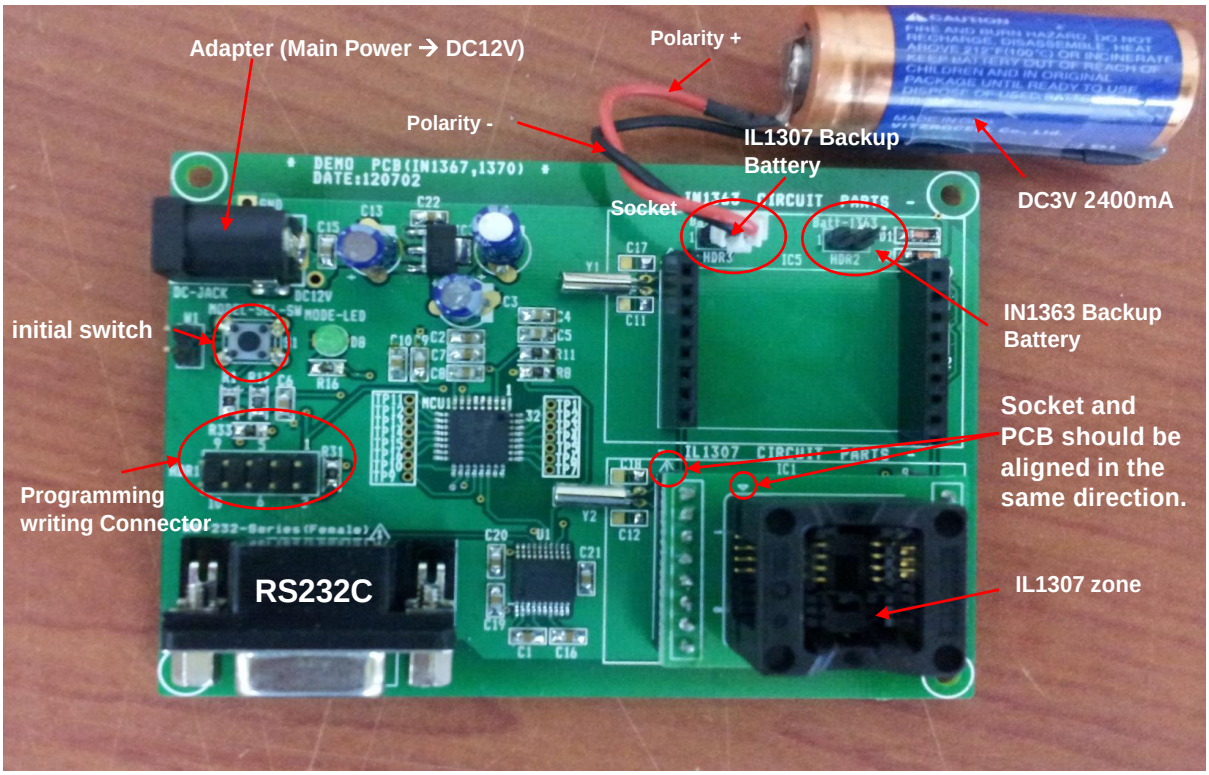


Fig 2 Top view of test fixture

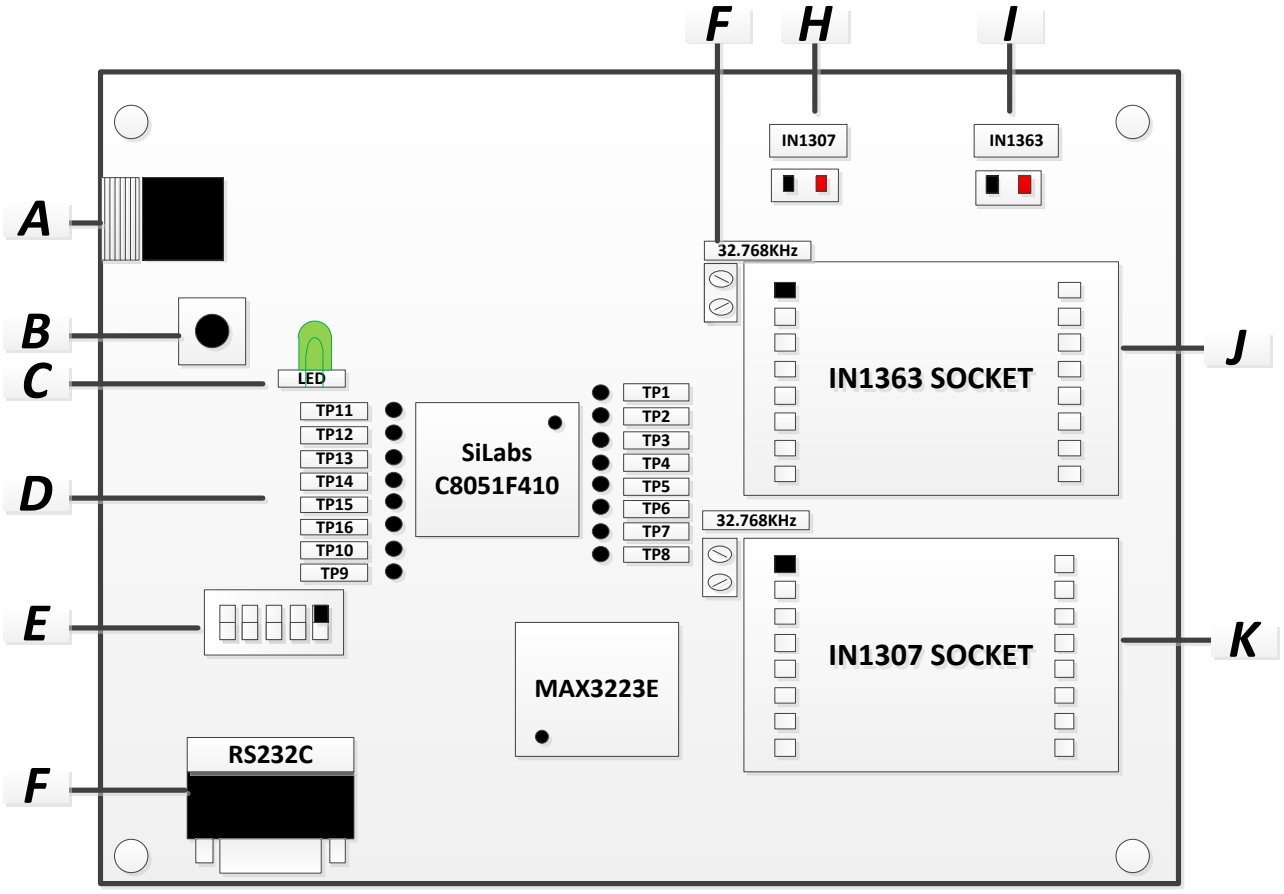


Fig 3 Layout of test fixture.

1.3. Basic Specification

A. 12V DC-JACK : Power Terminal
This power terminal normally is connected to the 12V power supply.

B. Init Switch
This switch initialize time to 23 July 2012, current time to 14:03:02(IN1363).

C. LED
This LED blinking every 1 seconds. if there is a power on the test fixture.

D. Test Point
In/Output pin of the MCU is connected.
If need the user, use as the test pin.

E. Debugger Connector
This test program exchanges with PC(Personal Computer) through this debugger connector.

F. RS232C Female Connector
User can see the test time through this connector.

G. 32.768KHz
Connection of the standard quartz resonator for the frequency 32.768KHz. IN1363 can operate from external oscillator with the frequency 32.768KHz

Note : To oscillation the 32.768KHz , Reset to the bit TEST1

H. IN1307_BAT
If you test the IN1307,Connect Battery on the IN1307_BAT.

I. IN1363_BAT
If you test the IN1363, Connect Battery on the IN1363_BAT

Note: Although the power eliminate, the Real Time Clock is working if the battery is connected on the IN1307/IN1363_BAT

J. IN1363_SOCKET
To test of IN1363 RTC, it should select IN1363_SOCKET.

K. IN1307_SOCKET
To test of IN1307 RTC, it should select IN1307_SOCKET.

Note: To avoid any confusion by program selector, only one IC should be selected by socket.

2. REAL TIME CLOCK PROGRAM

2.1. Flow Chart

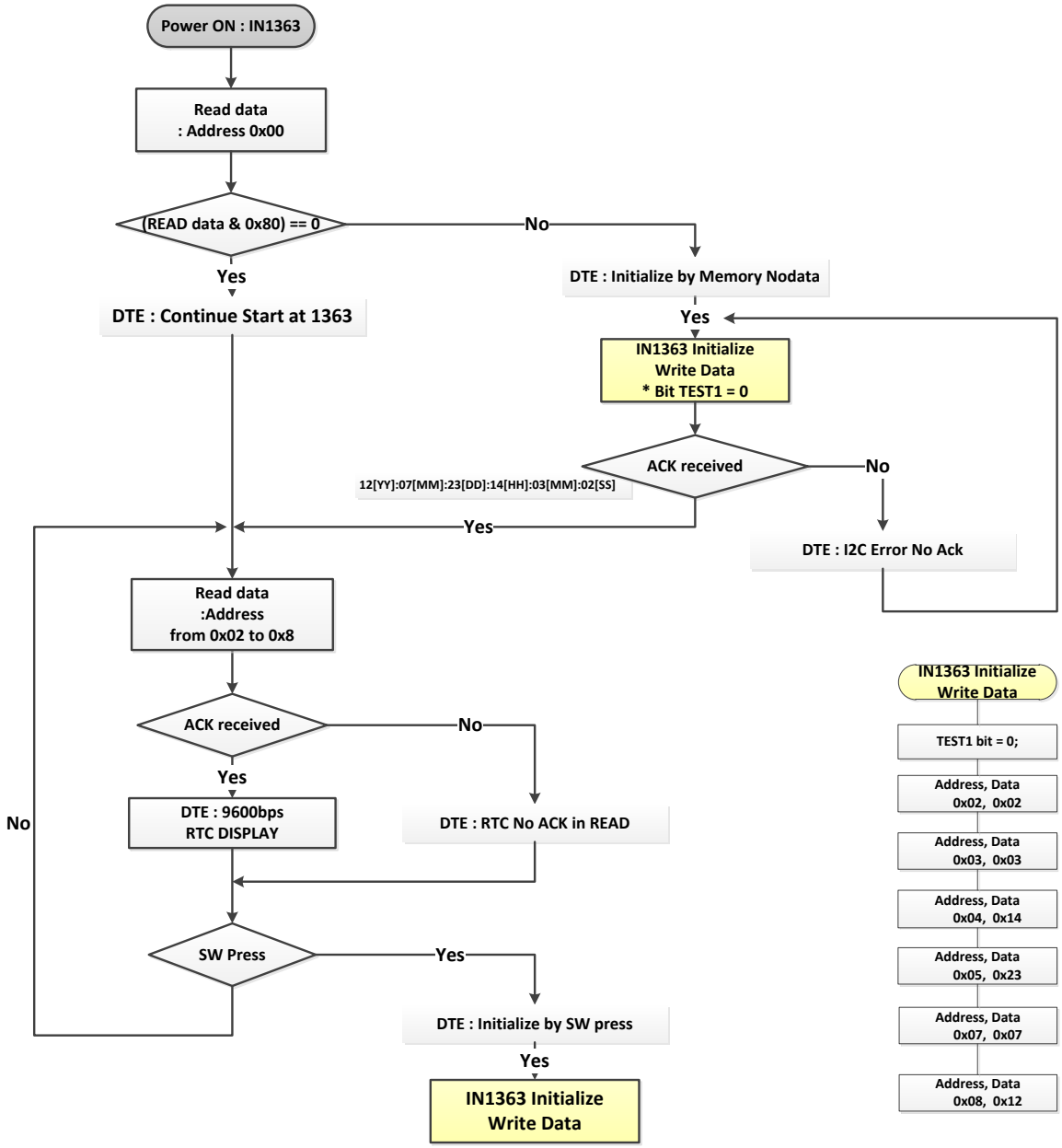


Fig3. Flow Chart for IN1363

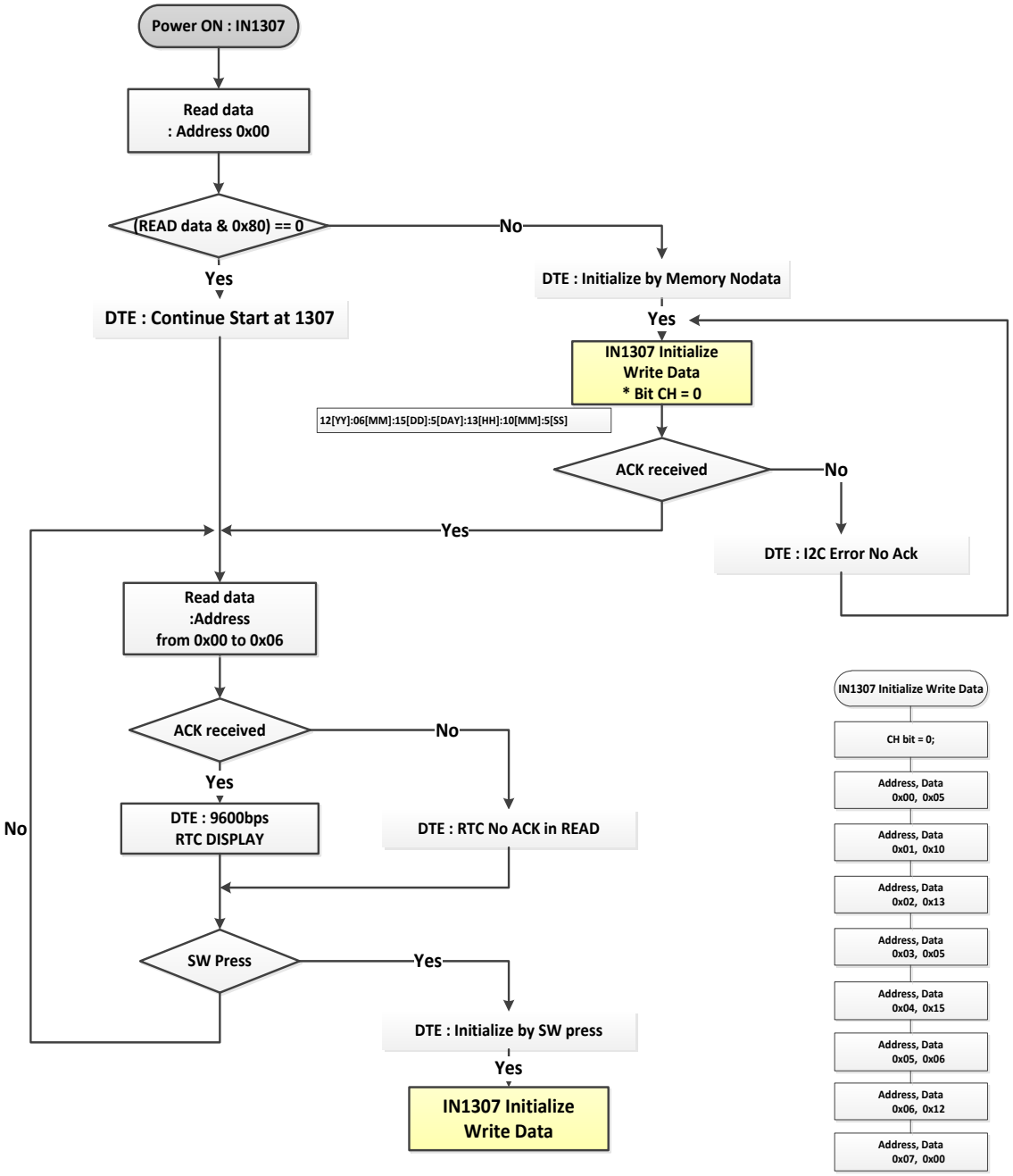


Fig4. Flow Chart for IN1307

2.2. Register Set Up for IN1363

2.2.1. Initialization Register Setting Up

It is necessary to enable the generation (bit TEST1 = 0) when setting the initial configurations..

Register Address	Register Name	Initialize Write Data	Contents
0x00H	Control/Status 1	0x00	TEST1 bit = 0 (normal mode) STOP bit = 0 (RTC source clock) TESTC bit = 0 (normal op)
0x01H	Control/Status 2	0x00	AIE bit = 0 (Alarm int disabled) TIE bit = 0 (Timer int disabled)
0x0DH	CLKPUT control	-	
0x0EH	Timer Control	-	
0x0FH	Timer	-	

2.2.2 Timer Register Setting Up

Register Address	Register Name	Initialize Write Data	Contents
0x02H	Seconds	0x02	
0x03H	Minutes	0x03	
0x04H	Hours	0x14	
0x05H	Dates	0x23	
0x06H	Day of week	-	Sunday=0,Monday=1,Tuesday=2, Wednesday=3,..., Saturday=6
0x07H	Century/month	0x07	Bit7 'C' of the months/century register indicates century for year 19xx (bit7=1), and year 20xx (bit7=0).
0x08H	Year	0x12	
0x09H	Minute alarm	-	
0x0AH	Hour alarm	-	
0x0BH	Date alarm	-	
0x0CH	Weekday alarm		

For example, we want to set the date for 23 July 2012, current time to 12:03:02, and then we need to call.

1. Seconds setup : rtc_write(SECONDS, 0x02);
2. Minutes setup : rtc_write(MINUTES, 0x03);
3. Hours setup : rtc_write(HOURS, 0x14);
4. Dates setup : rtc_write(DATES, 0x23);
5. Month setup : rtc_write(MONTH, 0x07);
6. Year setup : rtc_write(YEAR, 0x12);

2.3 Register Set Up for IN1307

2.3.1. Initialization Register Setting Up

It is necessary to **enable the generation (bit CH = 0)** when setting the initial configurations..

Register Address	Register Name	Initialize Write Data	Contents
0x00H.Bit8	Control/Status 1	0x00	CH bit = 0 (normal mode)

2.3.2 Timer Register Setting Up

Register Address	Register Name	Initialize Write Data	Contents
0x00H	Seconds	0x05	
0x01H	Minutes	0x10	
0x02H	Hours	0x13	
0x03H	Day	0x05	Range 1-7
0x04H	Date	0x15	
0x05H	Month	0x06	
0x06H	Year	0x12	
0x07H	Control		

For example, we want to set the date for 23 June 2012, current time to 13:10:05, and then we need to call.

1. Seconds setup : rtc_write(0x00, 0x05);
2. Minutes setup : rtc_write(0x01, 0x10);
3. Hours setup : rtc_write(0x02, 0x13);
4. Day setup : rtc_write(0x03, 0x05);
5. Dates setup : rtc_write(0x04, 0x15);
6. Month setup : rtc_write(0x05, 0x06);
7. Year setup : rtc_write (0x06 , 0x12);

2.4 Hyper Terminal Configuration

1. Bits per second : 9600 BPS
2. Data bit : 8 bit
3. Parity bit : None
4. Stop bits : 1 bit
5. Flow Control : None

```
case RX_LOOP:
    if( Timer1000ms_f ){
        Timer1000ms_f = 0;

        if( rtc_read_7_byte(READ_ADDRESS)){
            printf("I2c RTC No ACK in READ\n");
            break;
        }

        format[11] = ((Receive_Data[6] & 0xf0) >> 4) + '0'; //year
        format[12] = (Receive_Data[6] & 0x0F) + '0';

        format[0] = ((Receive_Data[5] & 0x10) >> 4) + '0';  //month
        format[1] = (Receive_Data[5] & 0x0F) + '0';

        format[2] = ((Receive_Data[3] & 0x30) >> 4) + '0';  //Date
        format[3] = (Receive_Data[3] & 0x0F) + '0';

        format[5] = ((Receive_Data[2] & 0x30) >> 4) + '0';  //Hour
        format[6] = (Receive_Data[2] & 0x0F) + '0';

        format[7] = ((Receive_Data[1] & 0x70) >> 4) + '0';  //Min
        format[8] = (Receive_Data[1] & 0x0F) + '0';

        format[9] = ((Receive_Data[0] & 0x70) >> 4) + '0';  //Second
        format[10] = (Receive_Data[0] & 0x0f) + '0';


        printf (" %c%c[YY]:%c%c[MM]:%c%c[DD]:%c%c[HH]:%c%c[MM]:%c%c[SS]\n",
            format[11],format[12],format[0], format[1],format[2],
            format[3],format[5],format[6],format[7],
            format[8],format[9],format[10]);
```

Fig5. Example test program for IN1363

Run the Serial terminal and make sure the baud rate is set correctly at 9600bps

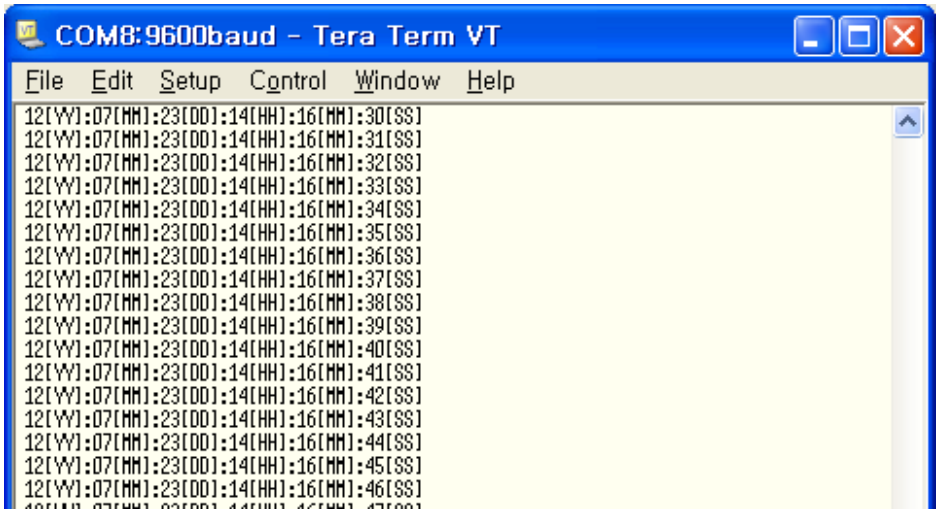


Fig6. IN1363 RTC Data on Serial Terminal

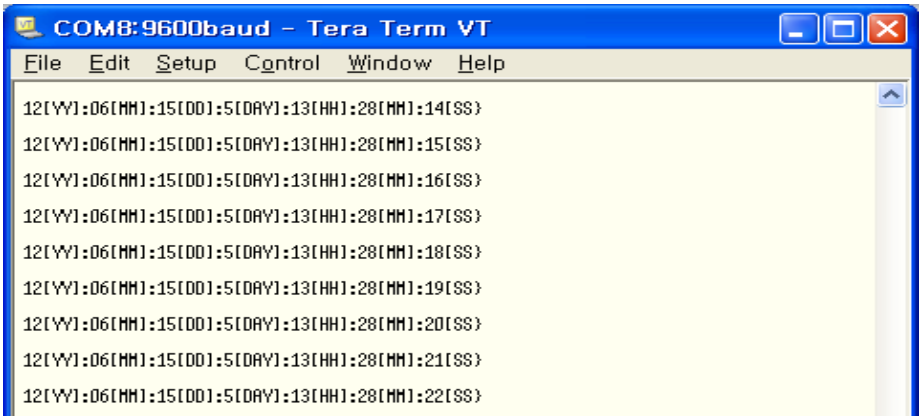
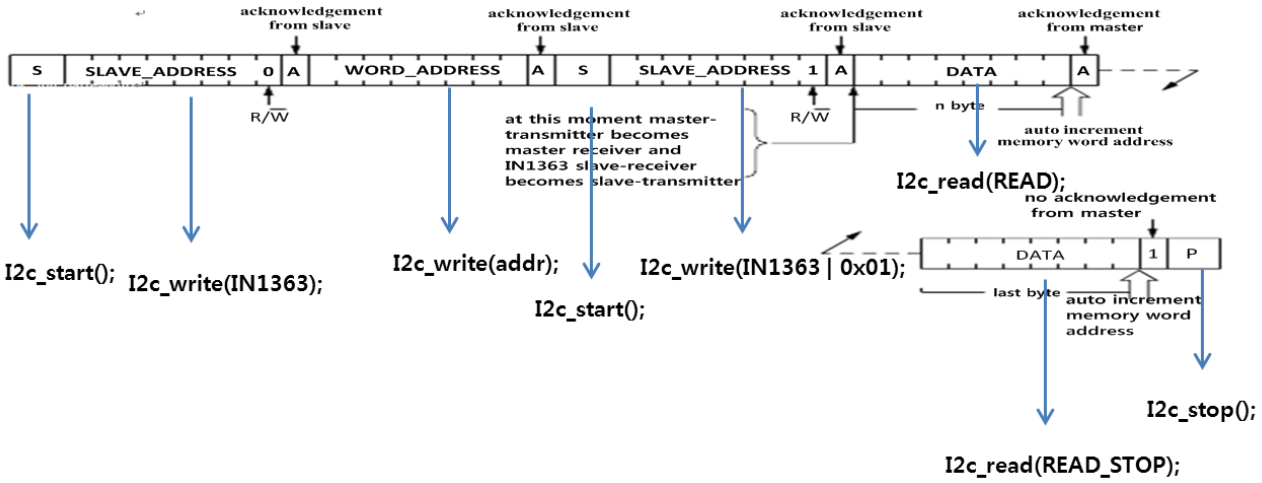


Fig7. IN1307 RTC Data on Serial Terminal

3.Example Source for RTC

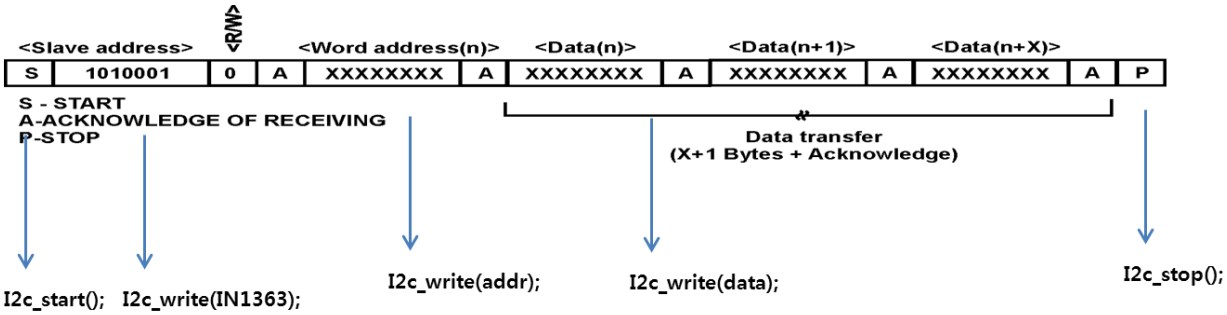
This documentation shows an example of a main function to setup and read/write data.
If you request example source code, we can provide source codes.

3.1. Read Current Time from IN1363 & IN1307



- Slave Address of IN1363 = 0xA2
- Slave Address of IN1307 = 0xD0;

3.2.Write Set Time for IN1363 & IN1307



3.3. I2C Example Source for IN1363 & IN1307

```
#ifdef IN1363
        sbit SCL        = P1^7;
        sbit SDA        = P1^6; //
#endif

#ifdef  IN1307
        sbit SDA        = P0^2;
        sbit SCL        = P0^1;
#endif

void i2c_start(void)              // START condition
{
        SDA = HIGH;
        DelayTimeLoop();
        SCL = HIGH;
        DelayTimeLoop();
        SDA = LOW;
        DelayTimeLoop();
        SCL = LOW;
        DelayTimeLoop();
}
void i2c_stop(void)     // STOP condition
{
        SCL = HIGH;
      DelayTimeLoop();
      SDA = HIGH;
        DelayTimeLoop();
}

/* Clock pulse generation. The function returns data or acknowledgment   bit */
unsigned char i2c_clock(void)//bit i2c_clock(void)
{
        bit level;                      // state of SDA line
        SCL = 1;
        DelayTimeLoop();
        while (!SCL);                   // if a pulse was stretched
        DelayTimeLoop();
        level = SDA;
        DelayTimeLoop();
        SCL = 0;
        return (level);
}

/* Writing a byte to a slave, with most significant bit first. The  function returns acknowledgment
bit.*/
unsigned char i2c_write(unsigned char  byte)
{
        unsigned char mask = 0x80;
        unsigned char aaa;
        while (mask) {
```

```
                if (byte & mask)
                        SDA = 1;
                else
                        SDA = 0;
                i2c_clock();
                mask >>= 1;
        }
        aaa = i2c_clock();
        return (aaa);
}

/* Reading byte from a slave, with most significant bit   first. The parameter indicates, whether to
acknowledge   (1) or not (0) */
unsigned char i2c_read(unsigned char acknowledgment)
{
        uchar mask = 0x80, byte = 0x00;
        while (mask) {
                if (i2c_clock())
                        byte |= mask;
                mask >>= 1;    /* next bit to receive */
        }

        if (acknowledgment) {
                SDA = 0;
                i2c_clock();
                SDA = 1;
        }
        else {
                SDA = 1;
                i2c_clock();
        }
        return (byte);
}

unsigned char rtc_read_7_byte(unsigned char addr)
{
        unsigned char status , i;
        i2c_start();
        if (!i2c_write(0xA2 )){//0xA2 is Slave Address for IN1363, 0xD2 is for IN1307
                DelayTimeLoop();
                if (!i2c_write(addr)){
                        i2c_start();
                        if (!i2c_write(0xA2 | 0x01)){ IN1307's slave address is 0xD2
                                for(i=0;i<6;i++){
                                        Receive_Data[i] = i2c_read(1);
                                }
                                Receive_Data[6] = i2c_read(0);
                        }
                         else {
                                status = 1;
                         }
                }
                else{
                        status = 1;
```

```
                }
        }
        else    status = 1;
        i2c_stop();
        return(status);
}
```

4. Test Procedure for RTC

4.1. Test Procedure for IN1363

1. Make sure that test fixture is power off.
2. Place IN1363 on IN1363_SOCKET of test fixture.
3. Connect the Battery on IN1363_BAT.
4. Run the Serial terminal and make sure the baud rate is set correctly at 9600bps
5. After turning on the power of test fixture, identify whether Green LED of Status Indic ator is flickering. If that's flickering every 1S, it indicates IN1363 working . Method fo r the program is refer to the Chapter 2.
6. Data is displayed in Serial Terminal window like to Fig6. refer to Fig6
7. And then test fixture is power off.
8. Wait for a period of time (two hours, 10 minutes, etc .).
9. Turn the test fixture on again, and check the time. the amount time as power off is p assed .

4.2. Test Procedure for IN1307

1. Make sure that test fixture is power off.
2. Place IN1363 on IN1307_SOCKET of test fixture.
3. Connect the Battery on IN1307_BAT.
4. Run the Serial terminal and make sure the baud rate is set correctly at 9600bps
5. After turning on the power of test fixture, identify whether Green LED of Status Indic ator is flickering. If that's flickering every 1S, it indicates IN1307 working . Method fo r the program is refer to the Chapter 2.
6. Data is displayed in Serial Terminal window like to Fig7. refer to Fig7
7. And then test fixture is power off.
8. Wait for a period of time (two hours, 10 minutes, etc .).
9. Turn the test fixture on again, and check the time. the amount time as power off is p assed .