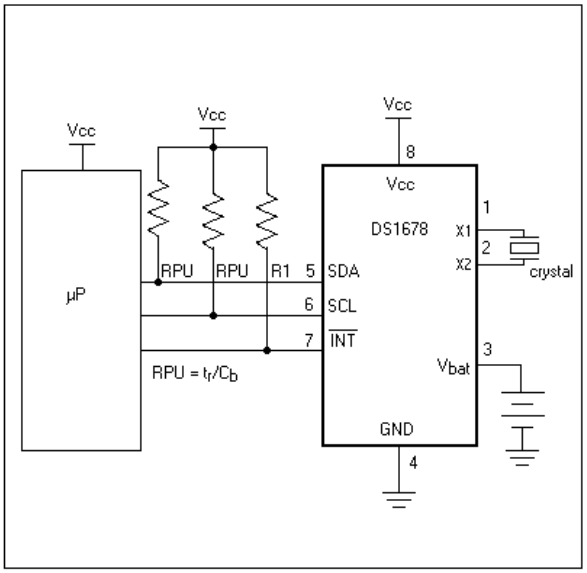
DS1678
Real-Time Event Recorder

www.maxim-ic.com

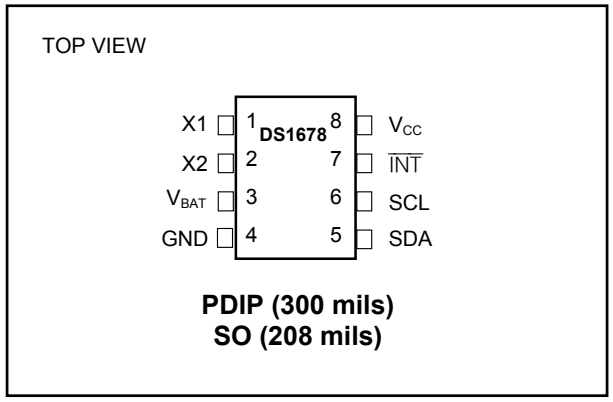
GENERAL DESCRIPTION

The DS1678 real-time clock (RTC) event recorder records the time and date of a nonperiodic, asynchronous event each time the $\overline{INT}$ pin is activated. The device records the seconds, minutes, hours, date, day of the week, month, year, and century when the first event occurs, and starts the 16-bit elapsed time counter (ETC). Subsequent events trigger the recording of the ETC into the event-log memory. This allows for up to 1025 events to be logged. Events can be logged while the device is operating from either V_{CC} or V_{BAT}.

TYPICAL OPERATING CIRCUIT



PIN CONFIGURATION



PDIP (300 mils)
SO (208 mils)

FEATURES

- **Real-Time Clock/Calendar in Binary-Coded Decimal (BCD) Format Counts Seconds, Minutes, Hours, Date, Month, Day of the Week, and Year with Leap Year Compensation and is Year 2000 Compliant**
- **Logs Up to 1025 Consecutive Events in Read-Only Battery-Backed Memory**
- **User-Programmable Event Trigger can be Triggered by the Falling Edge, Rising Edge, or Rising and Falling Edges of the $\overline{INT}$ Pin**
- **Event Counter Register Provides Data on the Number of Events that Have Been Logged in the Current Event-Logging Mission**
- **Programmable RTC Alarm**
- **32-Byte, Battery-Backed, General-Purpose NV RAM**
- **I^2C* Serial Interface**
- **Three Resolution Options for Trade-Off Accuracy vs. Maximum Time Between Events**
- **-40°C to +85°C Industrial Temperature Range**
- **Underwriters Laboratory (UL) Recognized**

ORDERING INFORMATION

PART†	PIN-PACKAGE	TOP MARK††
DS1678	8 Plastic DIP	DS1678
DS1678+	8 Plastic DIP	DS1678
DS1678S	8 SO	DS1678S
DS1678S+	8 SO	DS1678S
DS1678S/T&R	8 SO (Tape and Reel)	DS1678S
DS1678S+T&R	8 SO (Tape and Reel)	DS1678S

† *All devices are specified over the -40°C to +85°C operating range.*

†† *A "+" anywhere on the top mark denotes a lead-free device.*

+ Denotes a lead-free/RoHS-compliant device.

* *I^2C is a trademark of Philips Corp. Purchase of I^2C components from Maxim Integrated Products, Inc., or one of its sublicensed Associated Companies, conveys a license under the Philips I^2C Patent Rights to use these components in an I^2C system, provided that the system conforms to the I^2C Standard Specification as defined by Philips.*

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Pin Relative to Ground……..-0.3V to +6.0V
Operating Temperature Range (noncondensing)……...-40°C to +85°C
Storage Temperature Range……..…-55°C to +125°C
Soldering Temperature…….See IPC/JEDEC J-STD-020 Specification

This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect device reliability.

RECOMMENDED DC OPERATING CONDITIONS

(V_{CC} = $V_{CC(MIN)}$ to $V_{CC(MAX)}$, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at nominal supply voltage and T_A = +25°C, unless otherwise noted.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Power-Supply Voltage	V_{CC}	(Note 2)	4.5	5.0	5.5	V
Input Logic 1	V_{IH}	(Note 2)	2.2		V_{CC} + 0.3	V
Input Logic 0	V_{IL}	(Note 2)	-0.3		+0.8	V
Pullup Resistor Value	V_{PU}	V_{CC} = 0V (Note 2)			5.5	V
Battery Voltage	V_{BAT}	(Note 2)	2.6		3.5	V

DC ELECTRICAL CHARACTERISTICS

(**V_{CC} = $V_{CC(MIN)}$ to $V_{CC(MAX)}$**, T_A = -40°C to +85°C.)

PARAMETER		SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Input Leakage	SDA, SCL	I_{LI}	SDA output off	-1		+1	µA
	$\overline{INT}$		$\overline{INT}$ output off		10		
Logic 0 Output I_{OL} = 4mA (SDA, $\overline{INT}$)		V_{OL}				0.4	V
Active Supply Current		I_{CCA}			1	2	mA
Power-Fail Voltage (Note 2)		V_{PF}	V_{BAT} = 3.0V	1.216 x V_{BAT}	1.25 x V_{BAT}	1.284 x V_{BAT}	V
LOBAT Trip Point		$LOBAT_{TRP}$			1.35		

DC ELECTRICAL CHARACTERISTICS

(**V_{CC} = 0V**, T_A = -40°C to +85°C.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V_{BAT} Current (Oscillator On)	I_{OSC}			300	500	nA
V_{BAT} Current (Oscillator Off)	I_{BAT}			50	150	nA

AC ELECTRICAL CHARACTERISTICS

(V_{CC}= 2.6V to 5.5V or V_{BAT} = 2.6V to 3.5V, T_A = -40°C to +85°C.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Input Capacitance	C_I	$\overline{INT}$		10		pF
Minimum Signal Width	t_{GLITCH}		0.122		0.245	ms
Minimum Event Rate	t_{EVENT}		0.854		1.22	ms

AC ELECTRICAL CHARACTERISTICS

(V_{CC} = 4.5V to 5.5V, T_A = -40°C to +85°C.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f_{SCL}	Fast mode	100		400	kHz
		Standard mode			100	
Bus Free Time Between a STOP and START Condition	t_{BUF}	Fast mode	1.3			µs
		Standard mode	4.7			
Hold Time (Repeated) START Condition (Note 3)	$t_{HD:STA}$	Fast mode	0.6			µs
		Standard mode	4.0			
LOW Period of SCL	t_{LOW}	Fast mode	1.3			µs
		Standard mode	4.7			
HIGH Period of SCL	t_{HIGH}	Fast mode	0.6			µs
		Standard mode	4.0			
Setup Time for a Repeated START	$t_{SU:STA}$	Fast mode	0.6			µs
		Standard mode	4.7			
Data Hold Time (Note 4)	$t_{HD:DAT}$	Fast mode	0		0.9	µs
		Standard mode	0			
Data Setup Time (Note 5)	$t_{SU:DAT}$	Fast mode	100			ns
		Standard mode	250			
Rise Time of Both SDA and SCL Signals (Note 6)	t_R	Fast mode	$20 + 0.1C_B$		300	ns
		Standard mode	$20 + 0.1C_B$		1000	
Fall Time of Both SDA and SCL Signals (Note 6)	t_F	Fast mode	$20 + 0.1C_B$		300	ns
		Standard mode	$20 + 0.1C_B$		300	
Setup Time for STOP	$t_{SU:STO}$	Fast mode	0.6			µs
		Standard mode	4.0			
Capacitive Load for Each Bus Line (Note 6)	C_B				400	pF
Input Capacitance (SCL, SDA)	C_I			5		pF

WARNING: Under no circumstances are negative undershoots of any amplitude allowed when the device is in write protect.

Note 1: Limits at -40°C are guaranteed by design and not production tested.

Note 2: All voltages referenced to ground.

Note 3: After this period, the first clock pulse is generated.

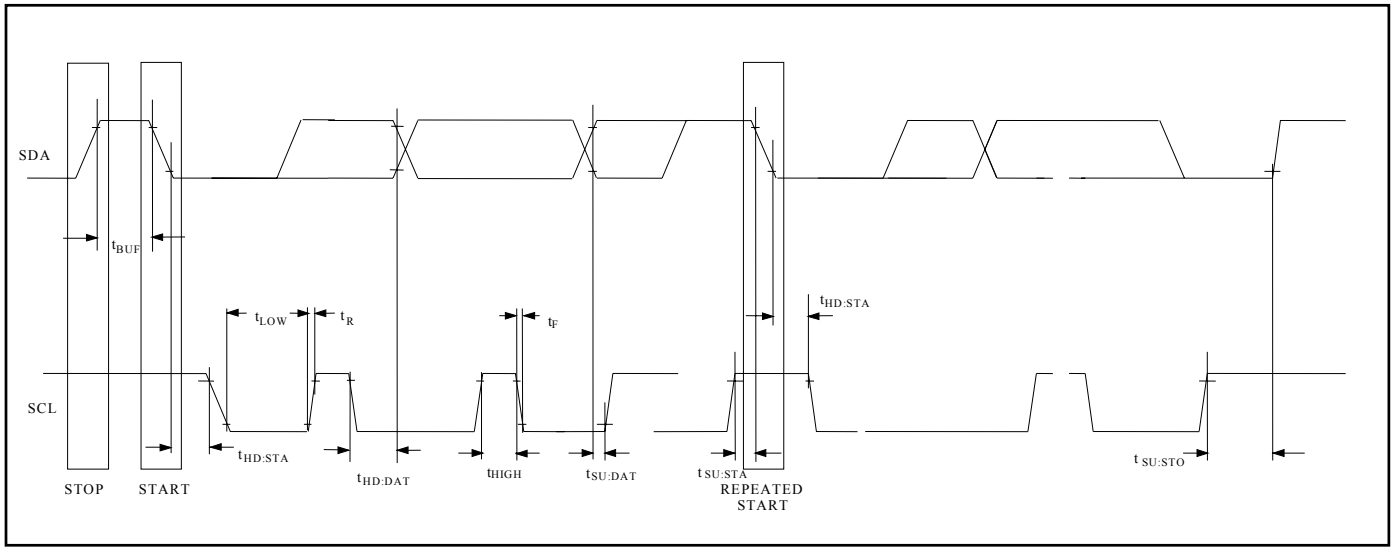
Note 4: A device must initially provide a hold time of at least 300ns for the SDA signal to bridge the undefined region of the falling edge of SCL. The maximum $t_{HD:DAT}$ need only be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.

Note 5: A fast-mode device can be used in a standard-mode system, but the requirement $t_{SU:DAT} > 250ns$ must then be met. This is automatically the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{R(MAX)} + t_{SU:DAT} = 1000 + 250 = 1250ns$ before the SCL line is released.

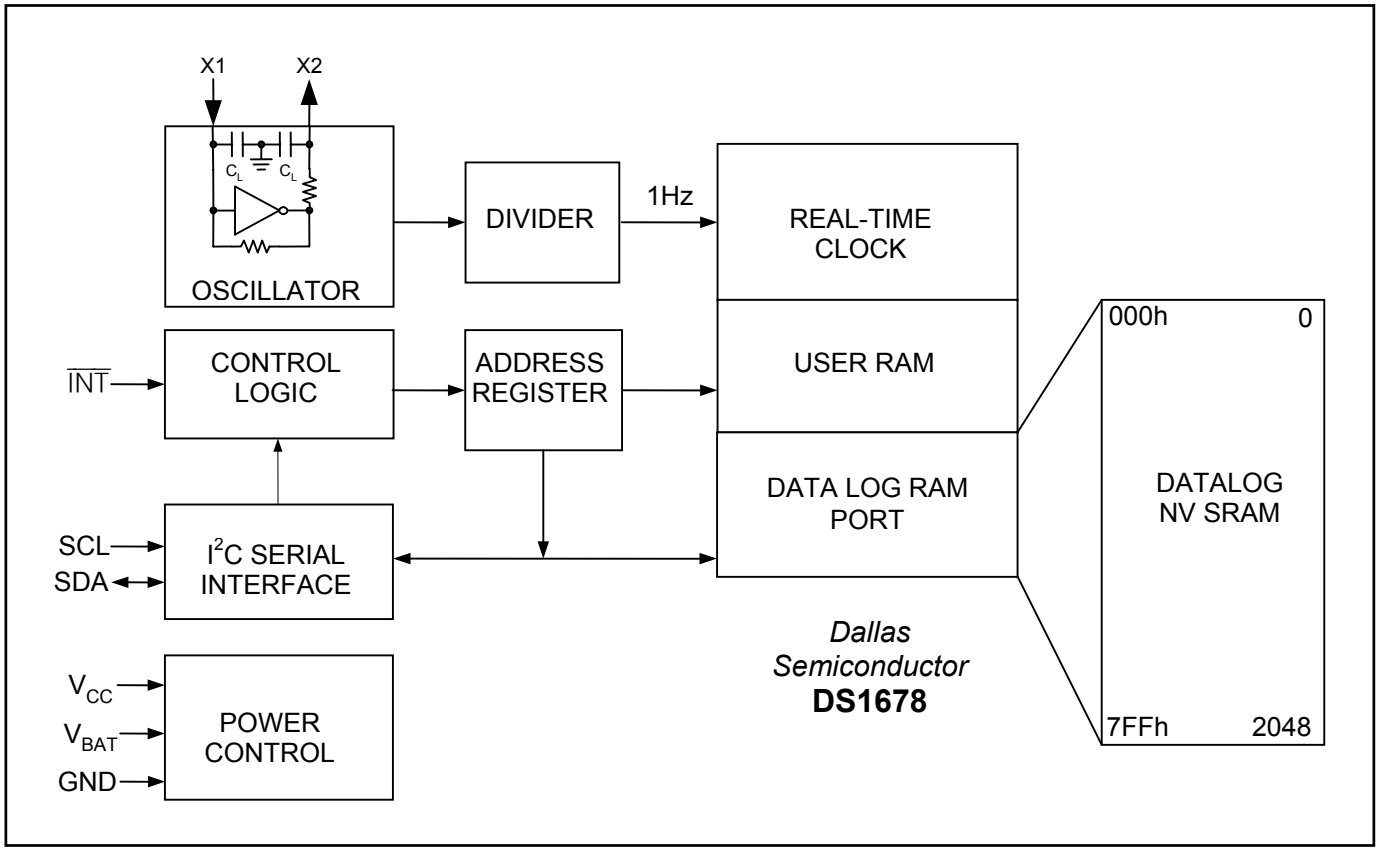
Note 6: C_B—Total capacitance of one bus line in pF.

Note 7: t_R and t_F are measured with a 1.7kΩ pullup resistor, 200pF pullup capacitor, 1.7kΩ pulldown resistor, and 5pF pulldown capacitor.

I²C COMMUNICATION TIMING DIAGRAM



BLOCK DIAGRAM



PIN DESCRIPTION

PIN	NAME	FUNCTION
1, 2	X1, X2	Connections for Standard 32.768kHz Quartz Crystal. For greatest accuracy, the DS1678 must be used with a crystal that has a specified load capacitance of 12.5pF. There is no need for external capacitors or resistors. Note: X1 and X2 are very high-impedance nodes. It is recommended that they and the crystal be isolated from high-frequency signals. For more information on crystal selection and crystal layout considerations, refer to *Application Note 58: Crystal Considerations with Dallas Real-Time Clocks*.
3	V_{BAT}	Battery Input for Standard Lithium Cell or Other Energy Source. All functions of the DS1678 with the exception of the serial interface circuitry are powered by V_{BAT} when $V_{CC} < V_{BAT}$. All functions are powered by V_{CC} when $V_{CC} > V_{BAT}$. The serial interface is enabled when V_{CC} is above V_{PF}. If a battery or other energy source is not used, V_{BAT} should be connected directly to ground. Diodes must not be placed between the battery and the V_{BAT} input or improper operation results. UL recognized to ensure against reverse charging current when used with a lithium battery. See "Conditions of Acceptability" at www.maxim-ic.com/qa/info/ul/.
4	GND	Ground
5	SDA	Serial Data Input/Output. SDA is the data input/output (I/O) signal for the I^2C serial interface. The SDA pin is an open-drain I/O and requires an external pullup resistor.
6	SCL	Serial Clock Input. SCL is used to synchronize data movement on the serial interface. It requires an external pullup resistor.
7	$\overline{INT}$	Active-Low Interrupt Input/Output. The $\overline{INT}$ pin is an I/O that is activated by an external device to signify an event has occurred and should be logged. Once the pin is activated, the event is recorded in the event-log memory and the Event Counter Register is incremented. The TRx bits determine which input edge(s) trigger an event: An event can be triggered by a falling edge on the $\overline{INT}$ pin, a rising edge, or by both the rising and falling edges. The $\overline{INT}$ pin can also be used as an output when the DS1678 is not in an event-logging mission. The $\overline{INT}$ pin becomes an output and generates an alarm interrupt if the DISx bits are set to zero and the RTC reaches the preset value in the alarm register. The $\overline{INT}$ output remains low as long as the status bit causing the interrupt is present and the DISx bits are set to zero. The $\overline{INT}$ pin is an open-drain input/output with a weak internal pulldown resistor to prevent the pin from floating when the pin is tri-stated.
8	V_{CC}	DC Power for Primary Power Supply

DETAILED DESCRIPTION

The Event Counter Register contains the total number of events that have been logged in the current event-logging mission. The Event Counter Register also allows the user to determine if the data in the event-log memory has rolled over.

The 16-bit ETC can be incremented once per second, once per minute, or once per hour. Each event transfers the current ETC value into the event-log memory, then clears and restarts the ETC. The three increment periods allow users to maximize the resolution while providing an adequate maximum time between events. The seconds resolution provides the time of an event down to the second, while allowing up to 65,535 seconds (18.2 hours) between events without using additional event-log memory. The minutes resolution provides the time of an event down to the minute, while allowing up to 45.5 days between events without using extra memory locations. The hours resolution provides the time of an event down to the hour in which it occurred, while allowing up to 7.5 years between events without using additional event-log memory. Based on the expected frequency of events, an increment period can be selected to maximize the resolution while minimizing use of the event-log memory.

The event can be triggered in three different ways depending on how the user programs the trigger select (TRx) bits in the Control Register. The event can be triggered by a falling edge on the $\overline{INT}$ pin only, a rising edge only, or it can be triggered by rising and falling edges. Triggering with both the rising and falling edges allows for monitoring when something is turned on/off and how long it is in either state.

The RTC provides seconds, minutes, hours, day, date, month, and year information with leap-year compensation, and year 2000 compliance. The RTC also provides an alarm interrupt. The I^2C interface allows the RTC to function as a stand-alone RTC in the system.

The programmable alarm trip points in the RTC allow a flag to be set in the Control Register when the specified time in the Alarm Trip Point Register is reached. The flag is readable via the I^2C interface during an event-logging mission or, when the DS1678 is not in a mission, $\overline{INT}$ becomes an output and generates an alarm interrupt if the value in the RTC equals the value in the RTC Alarm Register and the duration interval select (DISx) bits are both set to zeros.

The DS1678 operates as a slave device on the I^2C serial bus. Access is obtained by generating a START condition and providing a device identification code. All data is transferred to and from the DS1678 most significant bit (MSB) first. The address counter automatically increments so that subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below 1.25 x V_{BAT}, the device automatically write protects itself by disabling the I^2C interface, terminates any access in progress, and resets the device address counter. Inputs to the device via the I^2C bus are not recognized at this time in order to prevent erroneous data from being written to the device from an out-of-tolerance system. When V_{CC} falls below V_{BAT}, the device switches into a low-current battery-backup mode. Upon power-up, the device switches from battery power to V_{CC} when V_{CC} is greater than V_{BAT} + 0.2V, and recognizes inputs from the system when V_{CC} is greater than 1.25 x V_{BAT} by releasing control of the write protection on the I^2C bus.

The *Block Diagram* shows the main elements of the RTC event recorder. The device has four major components: a 64-bit RTC and control block, 32-byte user NV RAM, 2048 bytes of event-log memory (1024 events), and an I^2C serial interface.

POWER CONTROL

The device is fully accessible and data can be written and read when V_{CC} is greater than V_{PF}. However, when V_{CC} falls below V_{PF}, the internal registers are blocked from any access. The device power is switched from V_{CC} to V_{BAT} when V_{CC} drops below V_{BAT}. Operation, except for the I^2C interface, is maintained from the V_{BAT} source until V_{CC} is returned to nominal levels (Table 1). After V_{CC} returns above V_{PF}, read and write access is allowed.

Table 1. Power Control

SUPPLY CONDITION	READ/WRITE ACCESS	POWERED BY
$V_{CC} < V_{PF}$, $V_{CC} < V_{BAT}$	No	V_{BAT}
$V_{CC} < V_{PF}$, $V_{CC} > V_{BAT}$	No	V_{CC}
$V_{CC} > V_{PF}$, $V_{CC} > V_{BAT}$	Yes	V_{CC}

OSCILLATOR CIRCUIT

The DS1678 uses an external 32.768kHz crystal. The oscillator circuit does not require any external resistors or capacitors (C_L) to operate. Table 2 specifies several crystal parameters for the external crystal, and the oscillator block in the *Block Diagram* shows a functional schematic of the oscillator circuit. Using a crystal with the specified characteristics, the startup time is usually less than one second.

Table 2. Crystal Specifications*

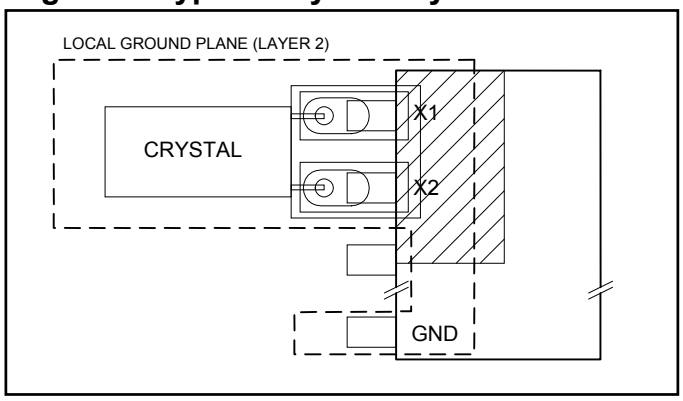
PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS
Nominal Frequency	f_O		32.768		kHz
Series Resistance	ESR			45	kΩ
Load Capacitance	C_L		12.5		pF

**The crystal, traces, and crystal input pins should be isolated from RF generating signals. Refer to* Application Note 58: Crystal Considerations for Dallas Real-Time Clocks *for additional specifications.*

CLOCK ACCURACY

The accuracy of the clock is dependent upon the accuracy of the crystal and the accuracy of the match between the capacitive load of the oscillator circuit and the capacitive load for which the crystal was trimmed. Additional error is added by crystal frequency drift caused by temperature shifts. External circuit noise coupled into the oscillator circuit can result in the clock running fast. Figure 1 shows a typical PC board layout for crystal and oscillator isolation from noise. Refer to *Application Note 58: Crystal Considerations with Dallas Real-Time Clocks* for detailed information.

Figure 1. Typical Crystal Layout



MEMORY

The memory map in Figure 2 shows the general organization of the DS1678. As can be seen in the figure, the device memory is in one contiguous segment with a data port to access the event-log memory. Because the I^2C bus is limited to a maximum of 256 addresses (one byte), the DS1678 uses the data port to access the 2048 bytes of event-log memory. The address that the next data would have been written to before logging was stopped is stored in the Address Pointer Register LSB (3Fh) and MSB (40h). These data bytes are used to recover all the data after a rollover occurs. The data log address pointer points to the oldest event in the memory after a rollover. This is the memory location in event-log memory that would be overwritten by the next event. Read the data from this point to the end of the memory and the start time stamp, including the two-byte ETC from the last event. Working backward from the value in the start time stamp, subtract the value in the ETC from the last event to get the time the last event in the memory occurred. Then subtract the values in each of the two-byte memory locations for elapsed time between events to recover the time the previous event occurred.

The value in the ETC register LSB (3Dh) and MSB (3Eh) is the value in the actual ETC. This is the time from the last event recorded until logging was stopped. Since a new event has not occurred, this data has not been stored in the event memory yet.

The data port is made up of three bytes. The first byte (41h) is the event-log memory address LSB, the second byte (42h) is the event-log memory address MSB, and the third byte (43h) is the event-log memory data byte. To access data via the data port, an I^2C write to the LSB of the event log LSB (41h) is performed, writing the appropriate LSB address information. The I^2C register pointer automatically increments to the event-log memory address MSB (42h), where a second I^2C write is performed, writing the MSB address information. The I^2C register pointer automatically increments to the event-log data byte address (43h). A repeated start, followed by the I^2C slave address with a read command (1) in the $R/\overline{W}$ bit of the I^2C address byte is performed. Subsequent read cycles reads the event-log information in the event-log memory.

For each read, the event-log memory address pointer in main memory locations 41h and 42h is autoincremented to the next higher event-log memory address, while the pointer for the main memory remains at location 43h. This allows the event-log memory to be read continuously without having to write the next desired event-log memory location prior to each data read. The even address locations in the event-log memory correspond to the LSB of the elapsed time between events, and the odd memory locations correspond to the MSB of the elapsed time between events. See Table 3 for more information about how the data is stored in the event-log memory.

When the event-log memory address pointer gets to the last address location (07FFh), the automatic incrementing stops. A new starting address must then be written into the event-log memory pointer bytes (41h and 42h) to begin reading additional data. The event-log memory addresses that can be put into the pointer (41h and 42h) are 0000h to 07FFh. The five MSBs of the address are ignored. Entering a value greater than 07FFh results in the address location associated with the value of the lowest 11 bits of the address.

The RTC and control registers (see Figure 2 for more details) are located in the main memory between addresses 00h and 0Fh. The user NV RAM resides in locations 10h through 2Fh. The event-logging memory data port is located at locations 41h, 42h, and 43h. Memory locations 44h and up are reserved for future extensions and read 00h.

The user can write only to the RTC, control registers, and user NV RAM. The rest of the memory map is read-only from the user's perspective. During an event-log mission, all the memory is read-only. A write terminates the mission. If there is an event being recorded when the mission is terminated, the event finishes being recorded before the mission is stopped, and the values in the MIP and ME bits do not change to zeros until the mission is complete.

During an event-log mission, memory locations 30h and above are not accessible to the user to avoid data collisions from a user read and an event being logged at the same time. If the user tries to read a location with an address greater than 2Fh during a mission, the value returned is 00h.

Table 3. DS1678 Event Elapsed Time Duration

ADDRESS	REGISTER
0000	Event 1 Elapsed Time from Last Event Counter LSB
0001	Event 1 Elapsed Time from Last Event Counter MSB
0002	Event 2 Elapsed Time from Last Event Counter LSB
0003	Event 2 Elapsed Time from Last Event Counter MSB
0004 ↓ 07FB	↓
07FC	Event 1023 Elapsed Time from Last Event Counter LSB
07FD	Event 1023 Elapsed Time from Last Event Counter MSB
07FE	Event 1024 Elapsed Time from Last Event Counter LSB
07FF	Event 1024 Elapsed Time from Last Event Counter MSB

Figure 2. DS1678 RTC and Control Page

	MSB							LSB	
ADDRESS	**BIT 7**	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**	**FUNCTION**
00	0	10 Seconds			Seconds				RTC
01	0	10 Minutes			Minutes				
02	0	12/$\overline{24}$	$\overline{AM}$/PM 10 Hr	10 Hr	Hour				
03	0	0	0	0	0	Day of Week			
04	0	0	10 Date		Date				
05	0	0	0	10 Mo	Month				
06	10 Year				Year				
07	10 Century				Century				
08	MS	10 Seconds Alarm			Seconds Alarm				RTC Alarm
09	MM	10 Minutes Alarm			Minutes Alarm				
0A	MH	12/$\overline{24}$	$\overline{AM}$/PM 10 Hr	10 Hr	Hour Alarm				
0B	MD	0	0	0	0	Day-of-Week Alarm			
0C 0D	(Reads 00h)								Reserved
0E	ME	CLR	DIS1	DIS0	RO	TR1	TR0	EOSC	Control
0F	0	MEM CLR	MIP	CM	LOBAT	ROF	0	ALMF	Status
10 11 12 ↓ 2F	Byte 1 Byte 2 Byte 3 ↓ Byte 32								User-Programmable NV Memory
Higher addresses read back as 00h while a mission is in progress.									
30	0	10 Seconds			Seconds				Time Stamp
31	0	10 Minutes			Minutes				
32	0	12/$\overline{24}$	$\overline{AM}$/PM 10 Hr	10 Hr	Hours				
33	0	0	0	0	0	Day-of-Week			
34	0	0	10 Date		Date				
35	0	0	0	10 Mo	Month				
36	10 Year				Year				
37	10 Century				Century				
38	Event 0 Elapsed Time from Last Event Counter LSB								Event 0 Rollover Stamp
39	Event 0 Elapsed Time from Last Event Counter MSB								
3A	Low Byte								Event Counter
3B	Medium Byte								
3C	High Byte								
3D	Low Byte								Elapsed Time Counter (ETC)
3E	High Byte								
3F	Low Address Byte								Address Pointer
40	High Address Byte								
41	Low Address Byte								Data Log RAM Port
42	High Address Byte								
43	Data Byte								
44 ↓ FF	(Reads 00h)								Reserved

EVENT LOGGING

When the DS1678 event-logging function is enabled, the device is said to be on an "event-log mission" until the event logging is stopped.

An event can be triggered one of three ways depending on the settings of the TRx bits in the Control Register. With the TR0 bit set to one and the TR1 bit set to zero, $\overline{INT}$ is activated on the falling edge of the input signal. With the TR0 bit set to zero and the TR1 bit set to one, $\overline{INT}$ is activated on the rising edge of the input signal. With both TR0 and TR1 bits set to one, $\overline{INT}$ is activated on both the falling and rising edges to allow for the measurement and duration of on/off type events. If TR0 and TR1 are both set to zero, nothing happens when $\overline{INT}$ is toggled, and a mission does not start. This is an illegal state and the mission does not start without a valid value in the TRx bits prior to attempting to start the mission.

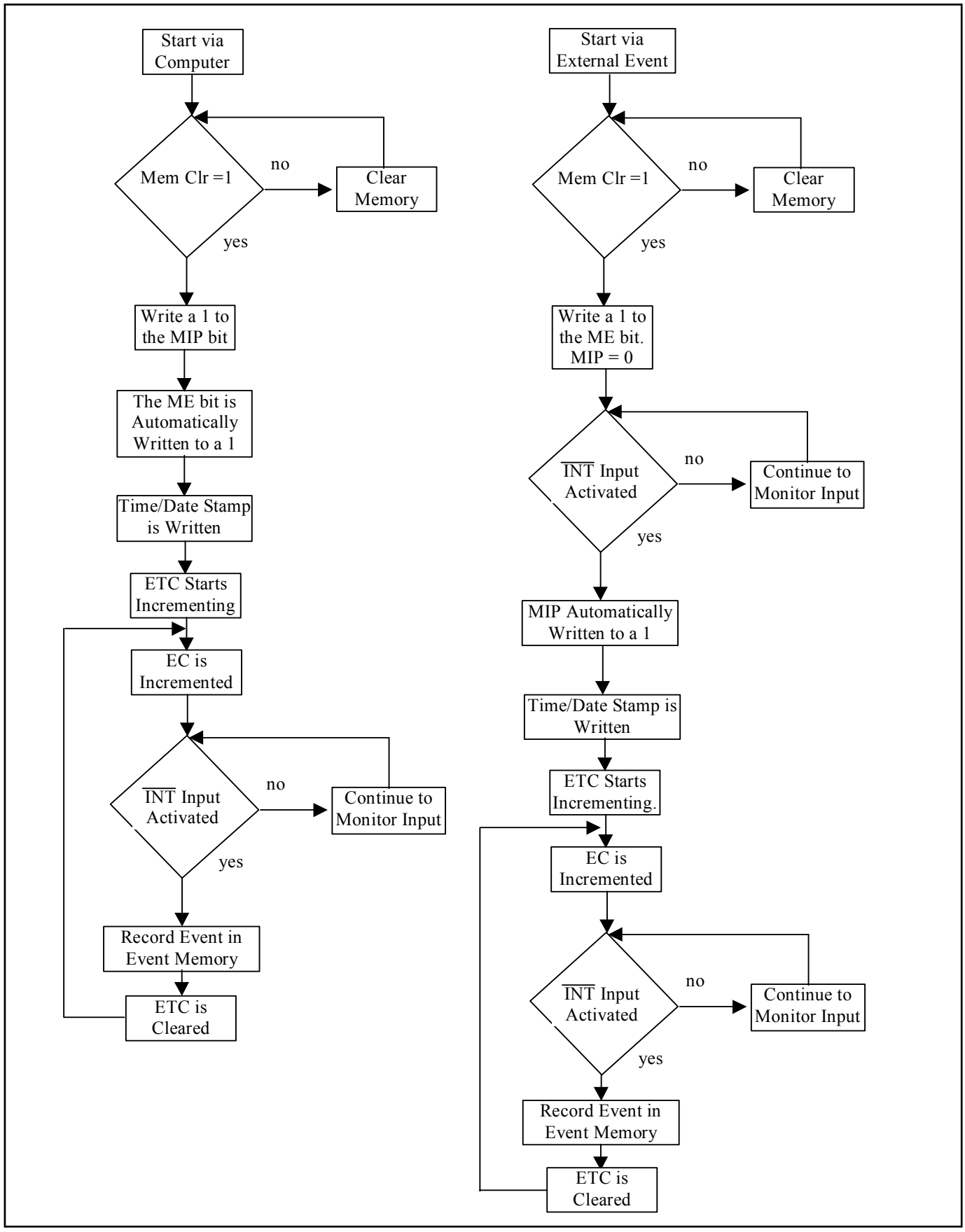
During an event-log mission, every time $\overline{INT}$ is activated, the elapsed time from the last event is written to the event-log memory pages. These memory pages are accessible through the data port in the main memory. To access data via the data port, the LSB of the address location in the event-log memory is written into 41h, the main memory address pointer automatically increments to 42h where the event-log memory address MSB data is written. The data from the event-log memory location corresponding to the address written into main memory locations 41h and 42h is available in location 43h to be read. The event-log data is located at addresses 0000h–07FFh in the event-log memory. The LSB of the first event duration is written to address location 0000h. The MSB of the first event duration is written to address location 0001h. The LSB of the second event is written to address location 0002h. The MSB of the second event duration is written to address location 0003h (see Table 3 for more details). Likewise, the address is incremented with each additional event duration. A total of 2048 registers have been reserved for event-log data, which allow 1024 events to be logged.

An event-log mission can be initiated by two methods (Figure 3). The first method to start a mission is with a delayed start. This is accomplished by writing a one to the ME bit. The mission starts when the first event occurs by activating $\overline{INT}$. When $\overline{INT}$ is activated, the MIP bit in the Status Register is set to one, the current time/date is written to the Start Time Stamp Register, and the Event 0 Rollover Stamp is written to zero. The Event Counter Register is incremented and the ETC starts. Subsequent events are logged as the duration of time from the previous event by writing the contents of the ETC into the event-log memory when that subsequent event is triggered by the activated $\overline{INT}$ pin. Note: The ME bit can only be written to one and a mission started if the MEM CLR bit is set to one.

The second way to start a mission is write a one to the MIP bit of the Status Register over the I^2C interface. When MIP is written to one, the ME bit in the Control Register is automatically set to one. When the MIP bit is written to one, the mission is started by loading the current time/date into the start time stamp, and the Event 0 Rollover Stamp is written to zero. The Event Counter Register is incremented and the ETC starts incrementing. The first event is then logged as the duration of time since the start time. All subsequent events are then logged as the duration of time since the previous event. Note: The MIP bit can only be written to one and a mission started if the MEM CLR bit is set to one.

The MEM CLR bit of the Status Register must be a one to start an event-log mission. This means that the Event-Log Memory, Event Count, ETC, Address Pointer, and Start Time Stamp registers are cleared of data (all zeros) so that an end user cannot turn the logger on and off to avoid recording events. Once the mission is stopped, the memory must be cleared to start a new mission.

Figure 3. Start Mission Flow Chart



ROLLOVER HANDLING

There are two options for dealing with the potential occurrence of a data overrun (i.e., more than 1024 total event logs in the event-log memory) (Figure 4). The first option is to enable rollovers. This is accomplished by setting the rollover bit (bit 3 of the Control Register) to one. When rollover is enabled, new data is written over previous data, starting with the Start Time Stamp Register, as if a new mission is starting.

When a rollover occurs, the Event 0 Rollover Stamp has the elapsed time since event 1024 of the event-log memory. This is to allow the user to recover the information recorded prior to the rollover. At the start of a mission, the Event 0 Rollover Stamp data is zero, as there was no previous event from which to have an elapsed time.

When 1024 events are recorded in the event-log memory, the next event causes a new time/date stamp to be written to the Start Time Stamp Register and the elapsed time since event 1024 written to the event 0 rollover stamp. The new event is written to the first location in the event log, overwriting the old data, and the address pointer is incremented. When the rollover occurs, the rollover flag (ROF) in the Control Register is set to one to indicate that the memory has rolled over at least one time.

The second option for dealing with data overrun is to disable rollovers by setting the rollover bit to 0. The DS1678 stops recording after event 1025, and the address pointer is incremented from 07FFh to 0000h. The device continues monitoring $\overline{INT}$ and the event counter continues to increment when $\overline{INT}$ is activated, even though the event-log memory has been filled.

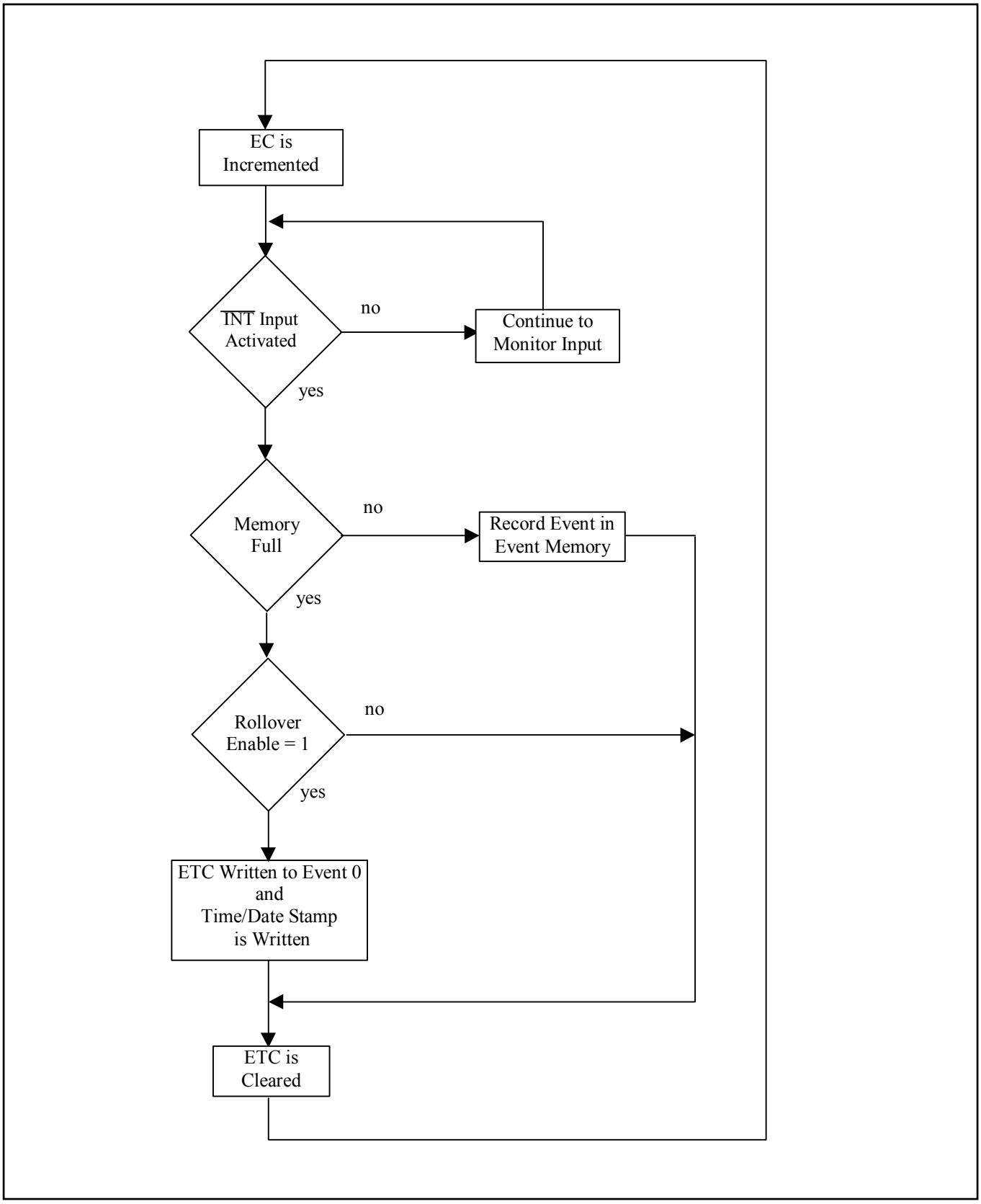
A time stamp for the first event is recorded after a mission begins. The time of acquisition for subsequent events is determined by considering the start time recorded by the time stamp; the value in the Event Counter Register, ROF; and the address of the particular data sample in the event-log memory.

If no rollover has occurred in the event-log memory (ROF = 0), the sample time associated with any particular data point can be calculated by multiplying the sum of the elapsed time between the events up to that event by one second, minute, or hour depending on which resolution is selected in the DISx bits of the Control Register, and adding this elapsed time to the value in the Start Time Stamp Register.

If rollover has been enabled, the user can determine if rollover has occurred by reading the value in the Event Counter Register or the ROF. The Event Counter Register counts the total number of events that have been acquired. If this value is greater than 0400h (decimal 1025), then the user knows that rollover has occurred. If rollover has occurred, the user needs to determine how many times rollover occurred in determining the sample time for any particular data sample. The address pointer points to the oldest data in the event-log memory, and, if the memory has rolled over at least one time, the rollover flag is set to one.

The DS1678 has been designed so the user cannot directly write to the event-log memory. This prevents writing invalid data to the event-log registers.

Figure 4. Rollover Flow Chart



CLOCK, CALENDAR, AND ALARM

The time and calendar information is accessed by reading/writing the appropriate register bytes. Note that some bits are set to zero. These bits always read zero regardless of how they are written. The contents of the time, calendar, and alarm registers are in the BCD format and are year 2000 compliant.

The DS1678 can run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the $\overline{AM}$/PM bit with logic one being PM. In the 24-hour mode, bit 5 is the second 10-hour bit (20–23 hours). The day register increments at midnight. Values that correspond to the day of week are user defined, but must be sequential, e.g., if 1 equals Sunday, then 2 equals Monday, and so on. Illogical time and date entries result in undefined operation.

When reading or writing the time and date registers, secondary (user) buffers are used to prevent errors when the internal registers update. When reading the time and date registers, the user buffers are synchronized to the internal registers on an I^2C start. The time and calendar information is read from these secondary registers, while the clock continues to run. The countdown chain is reset whenever the seconds register is written. Write transfers occur on the acknowledge from the device. Once the countdown chain is reset, to avoid rollover issues the remaining time and date registers must be written within one second.

The DS1678 also contains a time-of-day alarm. The alarm registers are located in registers 08h–0Bh. Bit 7 of each of the alarm registers is a mask bit (Table 4). When all the mask bits are logic zero, an alarm occurs once per week when the values stored in timekeeping registers 00h–03h match the values stored in the time-of-day alarm registers. An alarm is generated every day when the mask bit of the day alarm register is set to one. An alarm is generated every hour when the day and hour alarm mask bits are set to one. Similarly, an alarm is generated every minute when the day, hour, and minute alarm mask bits are set to one. An alarm occurs every second when the day, hour, minute, and seconds alarm mask bits are set to one. As a security measure to prevent unauthorized tampering, writing to any memory location or changing any value in the RTC and control registers stop an event-log mission and clear the MIP bit to zero.

Table 4. Time-of-Day Alarm Bits

ALARM REGISTER MASK BITS (BIT 7)				DESCRIPTION
SECONDS (MS)	MINUTES (MM)	HOURS (MH)	DAYS (MD)	
1	1	1	1	Alarm once per second
0	1	1	1	Alarm when seconds match
0	0	1	1	Alarm when minutes and seconds match
0	0	0	1	Alarm when hours, minutes, and seconds match
0	0	0	0	Alarm when day, hours, minutes, and seconds match

SPECIAL-PURPOSE REGISTERS

The following descriptions define the operation of the DS1678 special-purpose registers.

CONTROL REGISTER (0Eh)

MSB LSB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ME	CLR	DIS1	DIS0	RO	TR1	TR0	EOSC

Bit 7: Mission Enable (ME). This bit enables the device to begin a mission. The ME bit cannot be written to one unless the MEM CLR bit in the Status Register is one, signifying that the memory and registers have been cleared. With the ME bit set to one, the device waits for the first event to occur ($\overline{INT}$ is activated). Once that first event occurs, the MIP bit is set, the time/date stamp is recorded in the Start Time Stamp Register, the Event 0 Rollover Stamp is written to zero, the Event Counter Register is incremented, and the ETC begins incrementing.

When the ME bit is set to logic zero, the DS1678 waits until a one is written to the MIP bit via the I^2C interface to start the mission. When the MIP bit is written to one, the ME bit is set to one, the current time/date is recorded in the Start Time Stamp register, the Event 0 Rollover Stamp is written to zero, the Event Counter Register is incremented, and the ETC begins incrementing.

The ME bit is automatically written to zero whenever a mission is stopped.

Bit 6: Clear Enable (CLR). This bit enables the memory to be cleared. When this bit is set to logic one and the clear memory (CM) bit is subsequently set to one, the Event-Log Memory, Event Counter, and Start Time Stamp registers are all cleared to zeros. Following the writing of a one to the ME bit, the CLR bit is also set to logic zero, and the MEM CLR bit is set to logic one. If the clear enable bit is set, but a command other than writing a one to the clear memory bit is issued next, the CLR bit is cleared to zero and the contents of the Event-Log, Start Time Stamp, and Event Counter registers are unchanged.

Bits 5 and 4: Duration Interval Select 1 and 0 (DIS1 and DIS0). These bits select the amount of time between increments of the ETC that is used to determine the amount of time between events. After the first event is recorded, all subsequent events are recorded as the elapsed time since the previous event. When a subsequent event occurs, the ETC value is stored in the event-log memory.

To obtain the maximum accuracy of the event logger, the smallest possible resolution of the ETC should be selected. The expected maximum time between events must also be taken into account to get the full 1025 events logged, because when the ETC count reaches 65,535 increments, if the next event has not occurred by that point, FFFFh is written into the memory, the ETC rolls over to 0000h and continues to count until the next event occurs or FFFFh is reached again. Whenever FFFFh is reached by the ETC, it stores that value in event-log memory, resets to 0000h, continues counting, and the memory address pointer increments to the next memory location. Whenever an FFFFh appears in the memory, the next two bytes of data are part of the elapsed time for the same event, even if the value in the next two bytes of memory are 0000h, which means that an event occurred at time increment FFFFh. To recover the total elapsed time between events when FFFFh is in the event-log memory; add the contents of the next two bytes to the FFFFh. If the next two bytes are 0000h, indicating that the event occurred at FFFFh, add the 0000h to FFFFh to get FFFFh. If the next two bytes are 0100h, add 0100h to FFFFh to get 100FFh. This

represents the total elapsed time since the previous event. The event counter is not incremented when the ETC rolls over because a new event has not occurred.

The ETC is incremented as the selected seconds, minutes, or hours register of the RTC increments. Because the RTC continues to run, even when the ETC is cleared to prepare for a new event, and the ETC is incremented every time the selected byte in the RTC increments, the actual time resolution is not lost even when events occur more frequently than the minimum time resolution selected. If an event occurs half way between increments of the ETC, the first increment of the next event occurs when the seconds, minutes, or hours register increments next, thus preserving the correct time to the resolution selected in the DISx bits.

When the alarm interrupt output is used, the DISx bits should be set to zeros. An event-logging mission cannot be started if the DISx bits are set to zero. This enables the alarm flag to generate an alarm interrupt via the $\overline{INT}$ output pin.

With the DIS0 bit set to one and the DIS1 bit set to zero, the ETC increments every time the seconds register in the RTC is incremented. This gives the maximum resolution between events, but the counter rolls over to the next two memory bytes when it reaches the maximum value. The largest interval between events that can be accurately measured without using additional memory space and reducing the total number of events able to be logged is 65,535 seconds, or about 18.2 hours. If the maximum time between events could be greater than 18.2 hours, consider using one of the courser resolutions to conserve memory.

With the DIS0 bit set to zero and the DIS1 bit set to one, the ETC increments every time the minutes register in the RTC is incremented. This gives a medium resolution between events, but increases the largest possible interval between events that can be accurately measured without using additional memory space and reduces the total number of events able to be logged to 65,535 minutes, or about 45.5 days.

With both DISx bits set to one, the ETC increments every time the single hours byte in the RTC is incremented. This gives the lowest resolution between events, but increases the largest possible interval between events that can be accurately measured without using additional memory space and reduces the total number of events able to be logged to 65,535 hours, or about 7.5 years.

If a second event occurs before the ETC is able to increment for the first time, all zeros are logged in the event-log memory and the ETC resets. If this occurs, the time base remains correct as it is based on the separate RTC incrementing, but the exact time of the event is no more accurate than the size of the time increment that is chosen. For this reason, it is recommended to use the finest resolution possible for your logging to minimize the errors. If the normal duration between events is several days or months, then a few minutes or an hour may not be significant to your data accuracy.

Table 5. Duration Interval Select Bits

DIS1	DIS0	ELAPSED TIME COUNT RESOLUTION	MAX TIME BETWEEN EVENTS
0	0	Alarm Interrupt Output Enabled	—
0	1	Counter Increments Every Second	18.2 Hours
1	0	Counter Increments Every Minute	45.5 Days
1	1	Counter Increments Every Hour	7.5 Years

Bit 3: Rollover (RO). This bit determines whether the data log function of the DS1678 rolls over or stops writing data to the event-log memory if the event-log memory is completely filled. If RO is set to one, the event-log memory rolls over after all 2048 bytes in the event-log memory have been used. After the 1024th event recorded in the event-log memory, the following sample has the full time/date stamp information written to the Start Time Stamp register and the contents of the ETC written to the two bytes following the start time stamp (Event 0 Rollover Stamp). The next sample has the duration of time from the new start time stamp value written to event-log memory address locations 0000h and 0001h, overwriting the original data. Likewise, subsequent samples increment through the event-log registers, overwriting their data.

The Event 0 Rollover Stamp has the elapsed time since the last event in the event-log memory. This is to allow the user to recover the information prior to the rollover. At the start of a mission, the value in these two bytes is zero since there was no previous event from which to have an elapsed time.

If RO is set to zero, no further event logs are written to the event-log memory after all event-log memory registers have been filled. However, events continue to be recognized, and the Event Counter Register is incremented for each event.

Bits 2 and 1: Trigger Select 0 and 1 (TR1 and TR0). These bits select the edge(s) that activate $\overline{INT}$ to cause an event to be logged. An event can be triggered one of three ways, depending on the settings of the TRx bits in the Control Register. With TR0 set to one and TR1 set to zero, $\overline{INT}$ is activated on the falling edge of the input signal. With TR0 set to zero and TR1 set to one, the $\overline{INT}$ is activated on the rising edge of the input signal.

With TR0 and TR1 set to zero, $\overline{INT}$ is activated by the rising and falling edges to allow for the measurement and duration of on/off type events. If TR0 and TR1 are set to zero, nothing happens when $\overline{INT}$ is toggled, and a mission does not start. This is an illegal state, and the mission does not start without a valid value in the TRx bits prior to attempting to start the mission.

Table 6. Trigger Select Bits

TR1	TR0	EDGE(S) USED TO TRIGGER AN EVENT
0	0	Nothing, Illegal State
0	1	Falling Edge
1	0	Rising Edge
1	1	Both Rising and Falling Edges

Bit 0: Enable Oscillator (EOSC). This bit allows the clock oscillator to shut off to save power. The RTC no longer keeps time when the oscillator is shut off, but the information stored in the device memory is maintained. An event-log mission cannot start with EOSC set to zero, and the RTC must be reset to the correct value after the oscillator is restarted and prior to starting a mission to obtain good data. A clear memory also cannot be executed without the oscillator running. When $V_{CC} > V_{BAT}$, the oscillator automatically starts, no matter what the value in the EOSC bit, to allow proper communications. Disabling the oscillator with EOSC can be used to extend the battery life whenever time and date operation on battery backup is not required.

STATUS REGISTER (0Fh)

MSB LSB

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	MEM CLR	MIP	CM	LOBAT	ROF	0	ALMF

Bit 6: Memory Cleared (MEM CLR). This bit indicates that the Event-Log Memory, Event Counter, and Start Time Stamp registers are all cleared to zero. MEM CLR is cleared to zero when an event-log mission is started (i.e., MIP = 1).

Bit 5: Mission in Progress (MIP). This bit indicates the sampling status of the DS1678. If MIP is logic one, the device is currently on a "mission" in which it is operating in the event-logging mode. The MIP bit is changed to logic one immediately following the activation of $\overline{INT}$ if the ME bit of the Control Register contains a one. To immediately start an event-logging mission via the I^2C bus, a one can be written into the MIP bit and a one is automatically written into the ME bit of the Control Register.

If MIP is logic zero, the DS1678 is not currently in event-logging mode. The MIP bit transitions from logic one to logic zero whenever event logging is stopped. Event logging is stopped when the DS1678 is cleared by writing to the clear enable and memory clear bits, or when any memory location including the RTC or control registers is written to during a mission. The MIP bit can also be written to logic zero by the end user to stop event logging via the I^2C bus. By writing a zero to the MIP bit and stopping the mission, a zero is automatically written to the ME bit of the Control Register. It cannot, however, be written to logic one to start a mission unless the MEM CLR bit is a one to signify that the memory has been cleared.

Bit 4: Clear Memory (CM). This bit triggers the memory to be cleared if the CLR clear enable and EOSC oscillator enable bits in the Control Register are set to one. This causes the Event-Log Memory, Event Counter, and Start Time Stamp registers to all be cleared to zeros. Once the memory has been cleared, the CLR enable bit and the CM bits are set to zeros, and the MEM CLR bit is set to one to allow a new mission to begin. Clearing the memory is a two-write process to reduce the risk of accidentally erasing the memory. The CLR bit must be set to one before the CM bit can be written to one. During the clear memory operation, the DS1678 should not be accessed for 500µs while the memory is erased. The MEM CLR bit should read a one before trying to access the cleared memory or registers.

Bit 3: Low-Battery Flag (LOBAT). This bit reflects the status of the backup power source connected to the V_{BAT} pin. A logic one for this bit indicates an exhausted lithium energy source.

Bit 2: Rollover Flag (ROF). This bit is set to one if the RO bit of the Control Register is set to one, the last data log memory location has been filled, and a new event has occurred, which causes the time/date stamp to be overwritten. If RO is set to zero (rollover is disabled), the last data log memory location has been filled, and a new event has occurred, ROF is set to one to indicate that more events have occurred than the number of available memory locations. The event counter continues to record events, even after the event-log memory is full. The ROF is cleared by the clear memory command.

Bit 0: Alarm Flag (ALMF). A logic one in the alarm flag bit indicates that the current time has matched the time-of-day alarm registers. If, at the same time, the DISx bits are both logic zero, $\overline{INT}$ goes low to issue an alarm interrupt. ALMF is a read-only bit and is cleared by accessing any of the Alarm Register bytes either with a read or a write. Writing any memory location during a mission stops the mission. A mission cannot be started when the DISx bits are set to zero.

EVENT COUNTER REGISTER (3Ah–3Ch)

This three-byte register set provides the number of events that have been logged during the current data-logging operation (also known as a "mission"). The contents of this register can be used by software to point to the most recent data sample in the event-log memory. The data in these registers are cleared when the event-log memory is cleared. The event counter is not incremented when the ETC reaches FFFFh and rolls over to the next 16 bits of memory.

ADDRESS POINTER REGISTER (3Fh–40h)

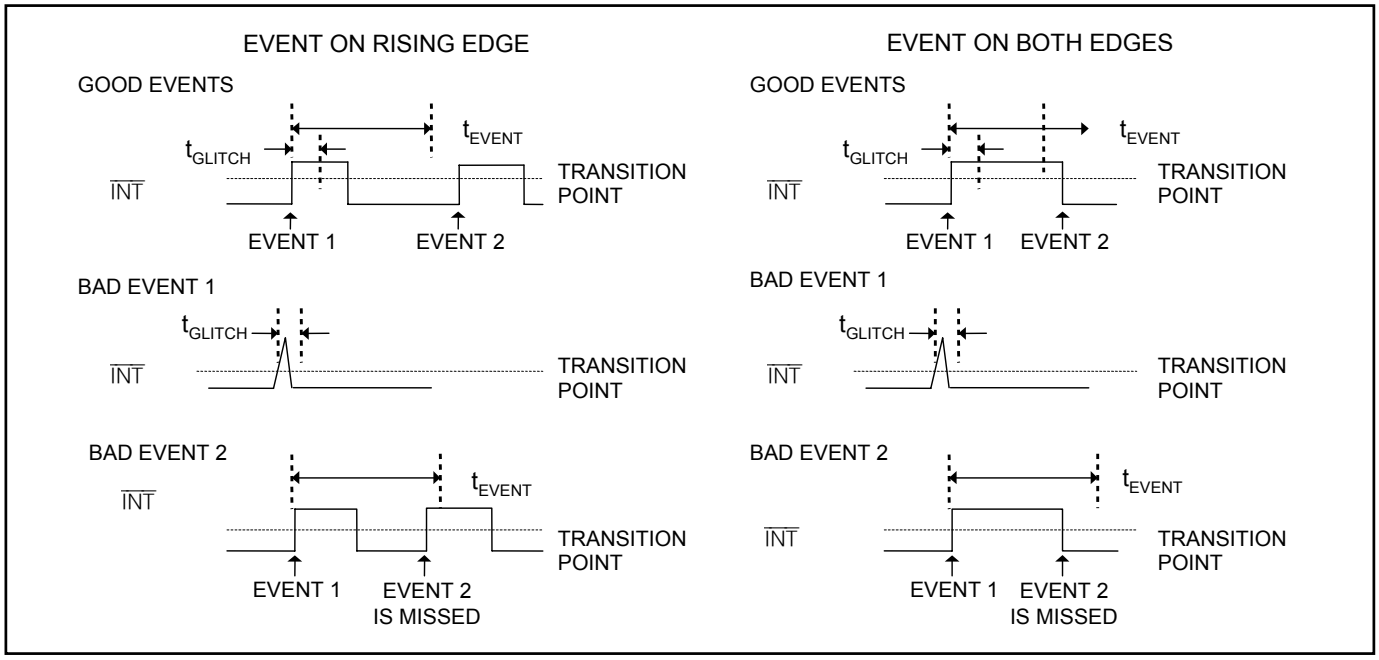
The address pointer register always contains the address that the next data LSB is written to in the event-log memory. The address pointer registers are located in the main memory map at LSB (3Fh) and MSB (40h). These are helpful in recovering all the data if a rollover occurs. The address pointer points to the oldest event in the memory after a rollover. This is the memory location that would be overwritten by the next event. Read the data from this point to the end of the memory and the start time stamp, including the two-byte ETC from the last event to recover all the data in the correct order.

GLITCH-CONTROL CIRCUIT

The DS1678 has a built-in glitch-control circuit to filter noise on $\overline{INT}$ from triggering false events. A minimum of one internal clock cycle (0.122ms) up to a maximum of two internal clock cycles (0.245ms) are required to recognize a transition on the input as an event. An event then requires an additional six to eight internal clock cycles (0.752ms to 0.977ms) to be processed and recorded into memory. This means that the minimum event occurrence that can be recognized by the DS1678 requires seven to 10 internal clock cycles (0.854ms to 1.22ms). Failure to ensure this timing causes the event to be ignored. Thus, it is recommended that you design with the maximum timing specs. See Figure 5.

$\overline{INT}$ has a weak internal pulldown resistor to prevent the pin from floating if the signal connected to the pin is tri-stated. Without the resistor, the input would float and potentially log phantom events. With the pulldown resistor, the pin can be transitioned to a low state, causing an event to be recorded if $\overline{INT}$ is held high by an outside signal that becomes tri-stated.

Figure 5. Event Recognition Timing



SECURITY

The DS1678 provides several measures to ensure data integrity for the end user. These security measures are intended to prevent third-party intermediaries from tampering with the data that has been stored in the event-log memory.

As a first security measure, the event-log memory is read-only from the perspective of the end user. The DS1678 can write the data into these memory banks, but the end user cannot write data to individual registers. This prevents an unscrupulous intermediary from writing false data to the DS1678. The end user, however, can clear the contents of the event-log memory. A new mission cannot be started unless the MEM CLR bit has been set to one to indicate that the memory and registers are cleared.

As a second security measure, changing any value in the memory including the RTC and control registers stops event logging and clears the MIP and ME bits. The MEM CLR bit must be set to one so the memory and registers are cleared before a new event-log mission can begin.

I^2C SERIAL DATA BUS

The DS1678 supports a bidirectional I^2C bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter, and a device receiving data as a receiver. The device that controls the message is called a "master." The devices that are controlled by the master are "slaves." The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1678 operates as a slave on the I^2C bus. Connections to the bus are made via the open-drain I/O lines SDA and SCL.

The following bus protocol has been defined (Figure 6):

- Data transfer can be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is HIGH are interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain HIGH.

Start data transfer: A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

Stop data transfer: A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the bus specifications a standard mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1678 works in both modes.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge-related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

Figure 6 details how data transfer is accomplished on the I^2C bus. Depending upon the state of the R/$\overline{W}$ bit, two types of data transfer are possible:

1) **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte.

2) **Data transfer from a slave transmitter to a master receiver**. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. Next follows a number of data bytes transmitted by the slave to the master. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Because a repeated START condition is also the beginning of the next serial transfer, the bus is not released.

The DS1678 can operate in the following two modes:

1) **Slave receiver mode (DS1678 write mode):** Serial data and clock are received through SDA and SCL. After each byte is received, the receiver transmits an acknowledge bit. START and STOP conditions are recognized as the beginning and end of a serial transfer. The slave address byte is the first byte received after master generates the START condition. The address byte contains the 7-bit DS1678 address, which is 1001010, followed by the direction bit (R/$\overline{W}$), which is 0. The second byte from the master is the register address. This sets the register pointer. If the write is being done to set the register pointer, a STOP or repeated START may then be sent by the master. Otherwise, the master then transmits each byte of data, with the DS1678 acknowledging each byte received. The master generates a STOP condition to terminate the data write (Figure 7).

2) **Slave transmitter mode (DS1678 read mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the direction bit indicates that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1678 while the serial clock is input on SCL. The slave address byte is the first byte received after the master generates a START condition. The address byte contains the 7-bit DS1678 address, which is 1001010, followed by the direction bit (R/$\overline{W}$), which is 1. After receiving a valid slave address byte and direction bit, the DS1678 generates an acknowledge on the SDA line. The DS1678 begins to transmit data on each SCL pulse starting with the register address pointed to by the register pointer. As the master reads each byte, it must generate an acknowledge. The DS1678 must receive a "not acknowledge" on the last byte to end a read (Figure 7).

Figure 6. Data Transfer on I²C Serial Bus

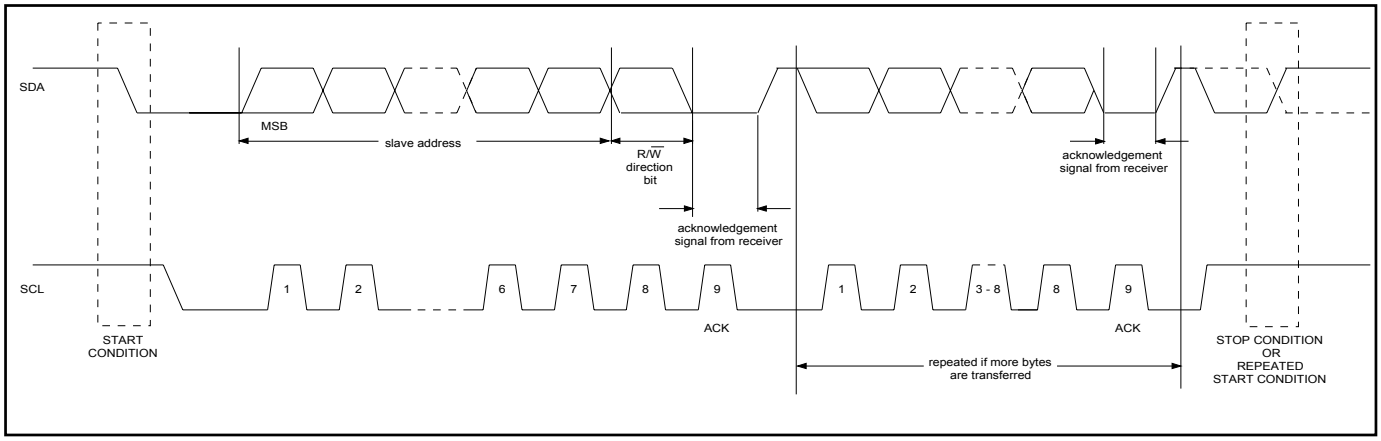
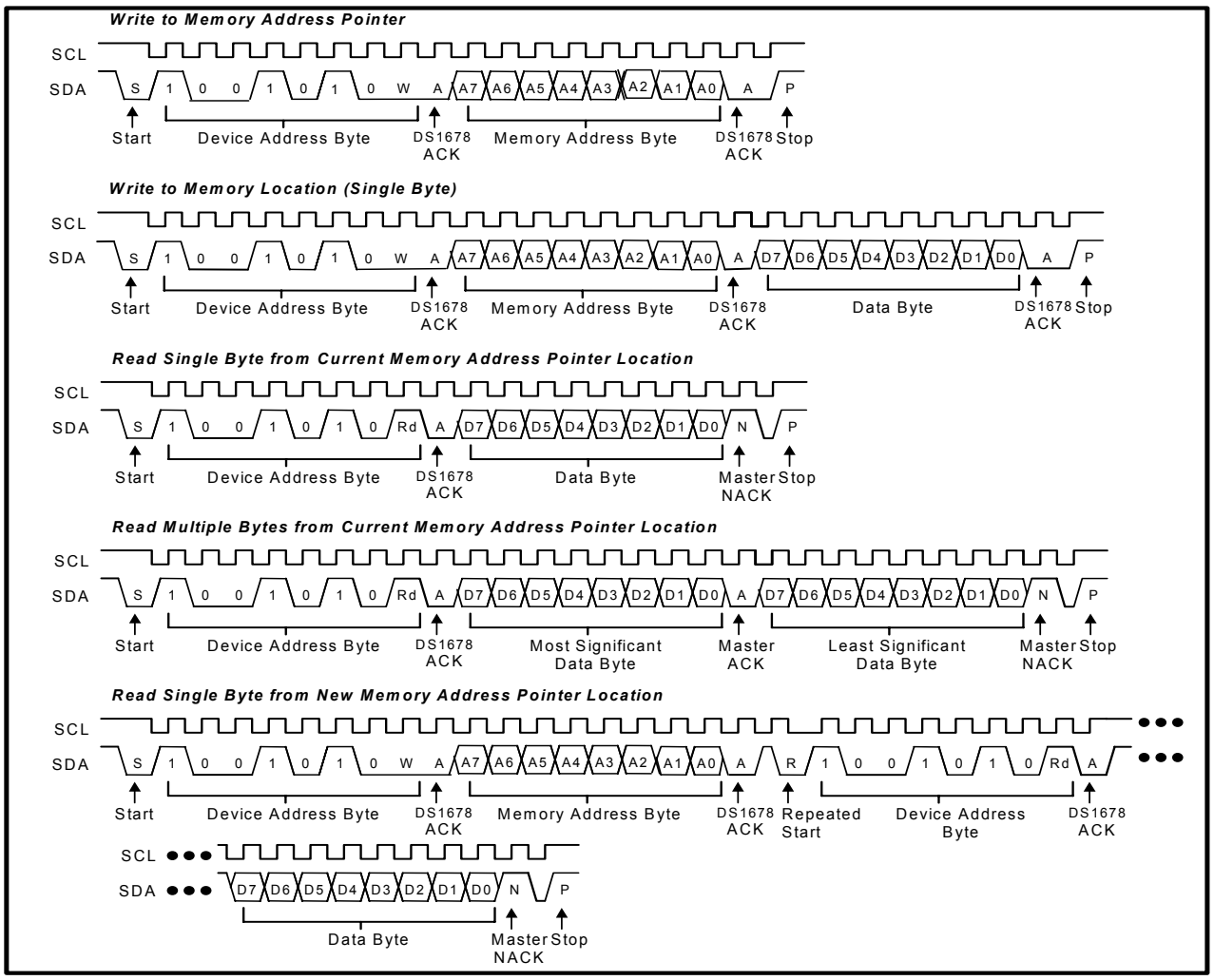


Figure 7. I²C Serial Communication with DS1678



CHIP INFORMATION

TRANSISTOR COUNT: 110,836
PROCESS: CMOS
SUBSTRATE CONNECTED TO GROUND

THERMAL INFORMATION

PART	THETA-J_A (°C/W)	THETA-J_C (°C/W)
8 PDIP	110	40
8 SO	113	31

PACKAGE INFORMATION

(For the latest package outline information, go to www.maxim-ic.com/DallasPackInfo.)

PACKAGE	DOCUMENT NUMBER
8-Pin Plastic DIP (300 mils)	56-G5005-000.pdf
8-Pin SO (208 mils)	56-G4010-001.pdf

Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408-737-7600