Application Report
SLUA408A–December 2006–Revised September 2008

HDQ Communication Basics

PMP Portable Power

ABSTRACT

This report compares the communication timing specifications for the different battery gas gauge and battery monitor products. It discusses some of the interface requirements necessary for robust HDQ communication, such as the need for a break prior to each communication, how to reliably read 16-bit dynamic values over the 8-bit bus, and the need for some noise filtering on the HDQ line. It also discusses the firmware strategy required for implementing the HDQ interface using a discrete I/O port and describes the popular option of using a UART to handle the HDQ bit timing requirement

Basic HDQ Protocol

HDQ communication between a host and slave device uses a single-wire, open-drain interface. The communication protocol is asynchronous return-to-one referenced to Vss. A passive pullup resistance is required to pull the HDQ line to a high state when neither the host nor the slave is pulling the line low during the two-way communication over the single wire interface. The interface uses a command-based protocol, where the host sends a command byte to the HDQ slave device. The command directs the slave either to store the next eight bits of data received to a register specified by the command byte (write command), or to output the eight bits of data from a register specified by the command byte (read command). Command and data bytes consist of a stream of bits that have a maximum transmission rate of 5 Kbits/s. The least-significant bit of a command or data byte is transmitted first. The first 7 bits of the command word are the register address and the last command bit transmitted is the read/write (R/W) bit. Figure 1 illustrates a typical HDQ read cycle.

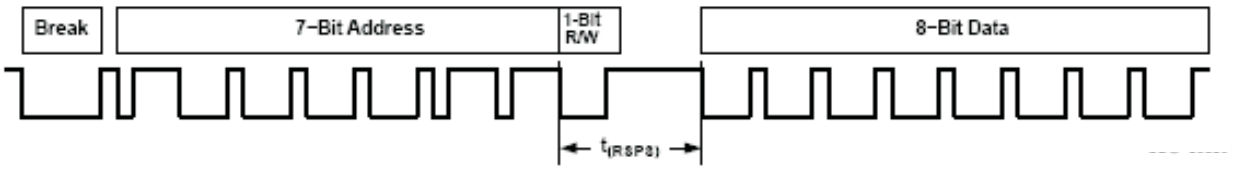

Figure 1. Typical HDQ Read Cycle

The HDQ line may remain high for an indefinite period of time between each bit of address or between each bit of data on a write cycle. After the last bit of address is sent on a read cycle, the HDQ slave starts outputting the data after the specified response time, $t_{(RSPS)}$. Some have interpreted the response time as the time after the last command bit before the first data bit of the response begins. This is incorrect. The response time is measured from the fall time of the command R/W bit to the fall time of the first data bit returned by the slave and therefore includes the entire bit time for the R/W bit. Because the minimum response time is equal to the minimum bit cycle time, this means that the first data bit may begin as soon as the command R/W bit time ends.

Sometimes this communication protocol is referred to as HDQ8 to distinguish it from the HDQ16 protocol used by some devices like the bq2060 and bq2063. The bit timing of HDQ16 is identical to that of HDQ8, except that 16 bits of data are written or read instead of 8. The HDQ16 command word is still a 7-bit address plus a R/W bit.

Break

The HDQ communication engine is reset if the HDQ line is held low for longer than the 190-s minimum break time, $t_{(B)}$. If the host does not get an expected response from the HDQ slave or if the host needs to restart a communication before it is complete, the host can hold the HDQ line low and generate a break to reset the communication engine. The next communication can begin after the 40-µs minimum break recovery time, $t_{(BR)}$. Break timing is illustrated in Figure 2.

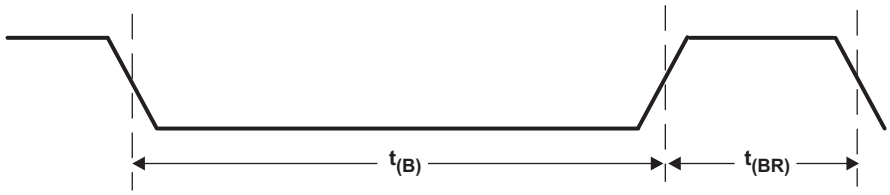

Figure 2. HDQ Break and Break Recovery Timing

If the HDQ line is disconnected and reconnected, unpredictable bit values may be input to the HDQ engine, leaving it in a non-reset state. It may be possible for the HDQ line to be disconnected during the middle of a communication. If HDQ communications are interrupted or unexpected transients occur on the HDQ line and there are no periodic breaks to reset the communication engine, it is possible for the HDQ slave engine to perform an incorrect command, including an unintended write.

The most robust design practice for reliable HDQ communication is to precede each communication with a break. This ensures that the engine is reset before each communication and minimizes the risk of unintended writes and/or reading incorrect data. Figure 1 illustrates a typical HDQ read cycle following a break.

Basic Timing

Figure 3 and Figure 4 show the HDQ bit timing. Figure 3 shows the host-to-slave bit timing and Figure 4 shows the slave-to-host bit timing. The HDQ timing description has been updated on more recent devices, like the bqJunior gas gauges, in an attempt to simplify the understanding of the HDQ timing requirements. The updated HDQ timing is specified using Figure 3a and Figure 4a, where previous HDQ interfaces have used Figure 3b and Figure 4b. The updated HDQ timing specifies the low-to-high transitions with a minimum and maximum value, where the older parts use different names for the minimum and maximum transition times. Other than changing the names and simplifying the way the timing is specified, the HDQ interface remains basically unchanged.

Short noise spikes coupled onto the HDQ line also can be perceived as a logic 1 by the communication engine. Some of the HDQ engines may detect spikes as short as 5 ns as a logic 1; so, some filtering on the HDQ line may be prudent. The bqJunior HDQ gas gauges (bq26500, bq26501, bq27000, and bq27010) have been designed to improve noise immunity on the HDQ line and will ignore such a short spike.

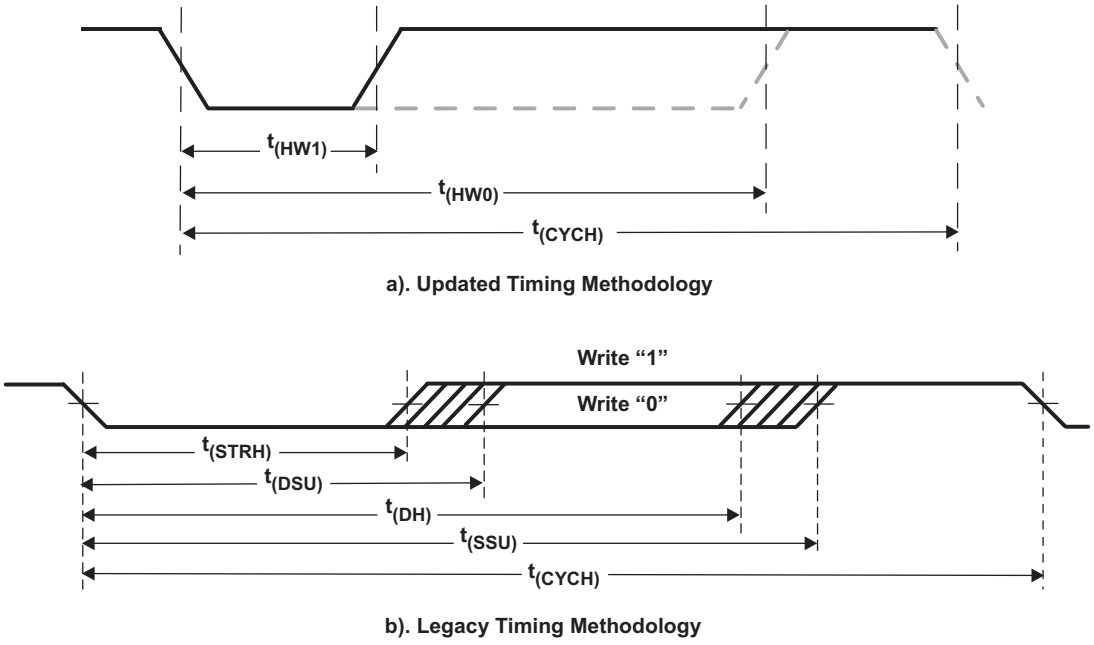

Figure 3. Host Transmitted HDQ Bit

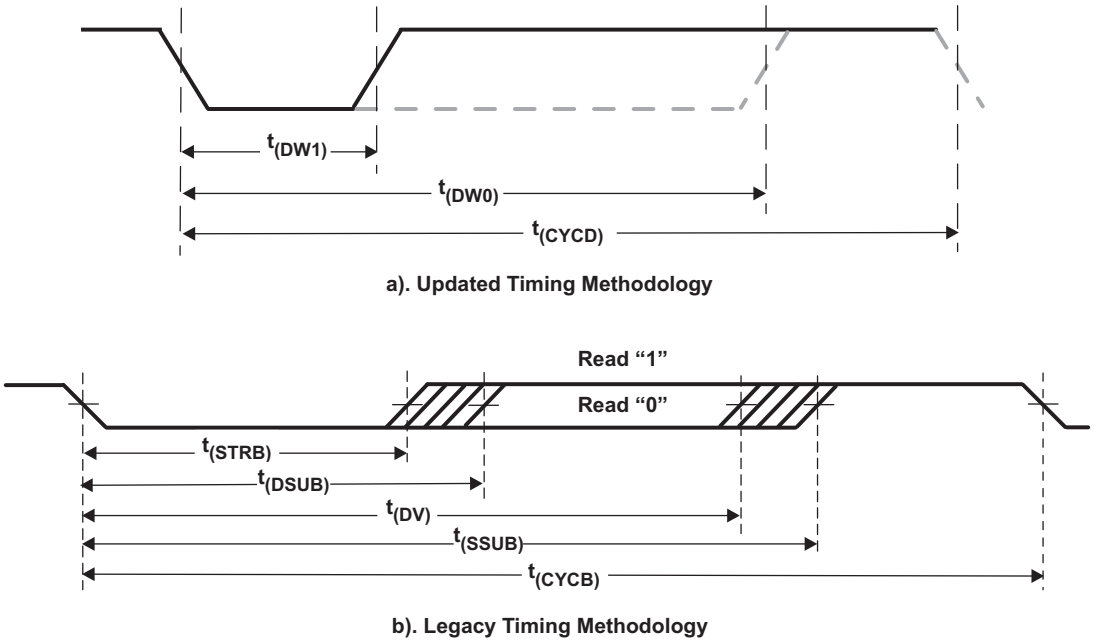

Figure 4. Slave Transmitted HDQ Data Bit

Comparison of Figure 3b with Figure 3a shows that t_{STRH} and t_{DSU} are equivalent to $t_{(HW1)}$ minimum and maximum values, respectively, and that t_{DH} and t_{SSU} are equivalent to $t_{(HW0)}$ minimum and maximum values, respectively. Likewise, comparison of Figure 4b with Figure 4a shows that t_{STRB} and t_{DSUB} are equivalent to $t_{(DW1)}$ minimum and maximum values, respectively, and that t_{DV} and t_{SSUB} are equivalent to $t_{(DW0)}$ minimum and maximum values, respectively.

Table 1. Timing Comparison

bq2018, bq2013H, bq2014H, bq2050H, bq2052 (Legacy Timing Methodology)				bq2019, bq262x0, bq2650x, bq270x0 (Updated Timing Methodology)			
Parameter	**Device**	**Minimum**	**Maximum**	**Parameter**	**Device**	**Minimum**	**Maximum**
t_B	All	190 µs		$t_{(B)}$	All	190 µs	
t_{BR}	All	40 µs		$t_{(BR)}$	All	40 µs	
t_{STRH}	All	5 ns		$t_{start\text{-}detect}$	bq2019, bq262x0	5 ns (1)	
				$t_{(HW1)}$	bq2019, bq262x0	32 µs (1)	
					bq26500	17 µs (2)	
					bq26051, bq270x0	0.5 µs (2)	
t_{DSU}	All		50 µs		All		50 µs
t_{DH}	All	100 µs		$t_{(HW0)}$	bq2019, bq262x0, bq26500	100 µs	
					bq26501, bq270x0	86 µs (3)	
t_{SSU}	All		145 µs		All		145 µs
t_{CYCH}	All	190 µs		$t_{(CYCH)}$	All	190 µs	
t_{STRB}	All	32 µs		$t_{(DW1)}$	All	32 µs	
t_{DSUB}	All		50 µs				50 µs
t_{DV}	All	80 µs		$t_{(DW0)}$	All	80 µs	
t_{SSUB}	All		145 µs				145 µs
t_{CYCB}	All	190 µs	250 µs	$t_{(CYCD)}$	All but bq2650x	190 µs	250 µs
					bq2650x		260 µs(4)
t_{RSPS}	All	190 µs	320 µs	$t_{(RSPS)}$	All	190 µs	320 µs

The few basic differences in HDQ timing among all the parts that support a slave HDQ interface are described in the following notes.

(1) The bq2019, bq26200, and bq26220 battery monitors are specified with a $t_{start\text{-}detect}$ time of 5 ns and a $t_{(HW1)}$ minimum time of 32 µs. These parts have identical characteristics to the bq2018, bq2013H, bq2014H, bq2050H, and bq2052H with a t_{STRH} specification of 5 ns. This specification methodology was an attempt to say that a spike on the line as short as 5 ns may initiate a communication start and if so, would be recognized as a logic 1. The recommended communication would be to use a much wider pulse, like 32 µs. All these parts reliably operate with a t_{STRH} or $t_{(HW1)}$ time down to about 1 µs. Usually, the addition of ESD protection on the HDQ line adds enough filtering to take out extremely short spikes like 5 ns, but does not substantially attenuate a 1-µs pulse.

(2) The bq26500 has internal rejection on the HDQ line that prevents recognizing any communication start if the HDQ line pulses low for much shorter than 17 µs. This part cannot interface with a TI OMAP HDQ interface that uses a 1-µs pulse for a logic-1 communication. The bq26501, bq27000, and bq27010 have a noise rejection interface that recognizes logic-1 pulses of at least 0.5 µs wide and thus are compatible with a TI OMAP HDQ interface.

(3) The bq26501, bq27000, and bq27010 have a $t_{(HW0)}$ minimum time of 86 µs instead of the normal 100-µs specification. This is required to achieve compatibility with a 13-MHz TI OMAP HDQ interface. Because a minimum of 86 µs is less than 100 µs, any host that meets the 100-µs minimum timing is also compatible with these parts.

(4) The bq26500 and bq26501 have a maximum bit cycle time, $t_{(CYCD)}$, on data sent to the host of 260 µs instead of the normal 250-µs specification value. This means that a host using either of these parts needs to wait a little longer for each bit before declaring a communication timeout.

Open HDQ Signal

Many HDQ slave devices have a built-in, pulldown current to force the HDQ line to a logic 0 if the HDQ line is opened. However, some devices like the bq2013H, bq2014H, and bq2050H do not have this feature and the HDQ line floats if disconnected. If the line floats to a midrange voltage level, the CMOS input stage could draw some *shoot-through* current in the HDQ input stage that may cause up to 100 μA of additional current drain by the IC. In these designs, the designer may wish to add a high value resistance, like 1 MΩ, at the slave HDQ device to ensure that the desired logic level at the slave HDQ input is retained when the HDQ host connection is broken.

Reading 16-Bit Words

Most HDQ interfaces only read or write 8 bits of data for each address sent to the device. The exceptions are the bq2060 and bq2063 gas gauges that handle 16-bit words on their HDQ16 interface. Reading 16-bit dynamic values from an 8-bit HDQ device requires special care to ensure that a register update occurring during the process of reading the two 8-bit bytes of the 16-bit word does not cause an inappropriate value read by the host. For example, if the 16-bit word is an incrementing or decrementing counter, it is possible for a carry or borrow to the high byte could occur after one data byte is read but before the other data byte is read. In this case, the word read by the host might be as much as 256 counts in error due to the low-high byte rollover that occurred during the data read process. To prevent any system issues, any 16-bit values read by the host should be read with the following protocol.

1. Read high byte (H0)
2. Read low byte (L0)
3. Read high byte again (H1)
4. If H1 = H0, then the valid 16-bit result is H0:L0
5. Otherwise, read low byte again (L1) and the valid 16-bit result is H1:L1.

This procedure is sufficient to guarantee that the 16-bit word is read correctly if the 3- to 4-byte reads occur in less time than the update rate of the register value.

Host Processor Interrupts Using Discrete I/O Port for HDQ

If the host implements the HDQ communication using a discrete processor I/O port, the timing of the transmitted HDQ data and the sampling of the received HDQ data depends on the host processor timing of the transitions on the HDQ line. If the HDQ communication routine is interrupted during a communication, it may cause the transmitted times to stretch and received data may not be interpreted correctly.

One solution is to disable interrupts during HDQ communication critical times. When the host is sending address or data, there is no restriction on the time between each bit, so host interrupts can be enabled during the high time between bits. Interrupts may need to be disabled during the low bit times to ensure that the bit low times meet the required HDQ timing constraints. After the last address bit (R/W bit) is sent on a read, interrupts also must be disabled to ensure that the received data is sampled correctly. Interrupts must be disabled during the entire receipt of the 8 data bit times.

If disabling interrupts during HDQ communication critical times is not feasible, another approach is to leave interrupts enabled, but to have all enabled interrupts set a flag that can be read to determine whether an interrupt occurred during an HDQ communication. The strategy would be to abort the communication with a break and then retry the communication if an interrupt occurred during an HDQ communication. This method requires that there be a reasonable probability of completing an HDQ communication without an interrupt.

Using UART Interface to HDQ

An implementation option for HDQ is to use an UART. An advantage to using the UART is that if the UART is handling the communication and storing the results in a buffer, host processor interrupts during the communication do not affect the timing of the HDQ communication with the HDQ slave. Use of the UART for HDQ communication requires that each word sent to or received from the UART is only a single bit of the HDQ data or address. The procedure is to set the UART baud rate to 57,600 with no parity and 2 stop bits. This yields a data word with 11 bits total (start bit, 8 data bits, and 2 stop bits). At a baud rate of 57,600 (17.3 μs per bit), this is a total communication time of 190.9 s and meets the required HDQ bit timing of 190 s minimum. If data of 0xFE is sent to the UART, the transmitted data is low for 34.6 s and then high for the remaining bit time and is interpreted by the HDQ slave as a one. If data of 0xC0 is sent

to the UART, the transmitted data is low for 121.5 µs and then high for the remaining bit time and is interpreted by the HDQ slave as a zero. When data is sent to the host from the HDQ slave, the received data could be interpreted as either 0xFE or 0xFC if a logic 1 is sent, or either 0xF0, 0xE0, 0xC0, 0x80, or 0x00 if a logic 0 is sent. A simple test of the received data determines the received data bit. If the received data is greater than 0xF8, the data bit should be interpreted as a logic 1 and if less than or equal to 0xF8, the data bit should be interpreted as a logic 0. This analysis assumes the UART samples the received data approximately half-way through each of the 17.3-µs UART bit times and that capacitive loading on the HDQ line may delay the rise time of the data a few microseconds.

Note that the TX and RX of the UART must be tied together because HDQ is a single-wire interface. In case the TX output is not an open-drain output, it needs to be converted to an open-drain output as shown in Figure 5. Note also that any data sent out to the HDQ slave is also received by the UART. So, if 8 bits of address are sent and then 8 bits of data from the HDQ slave are received from that address, the UART inputs 16 bytes of data into the UART data buffer. The host needs to skip the first 8 bytes which contain the command word sent to the HDQ slave and use the second 8 bytes of data.

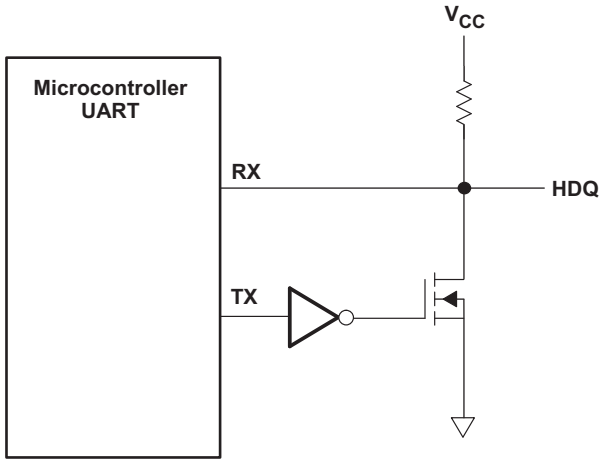

Figure 5. HDQ Communication Using UART Without Open-Drain Output

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265