



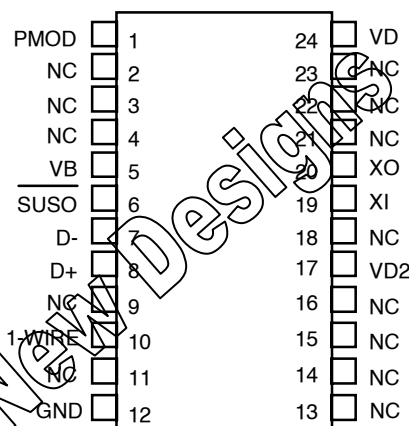
DS2490 USB to 1-Wire Bridge Chip

www.maxim-ic.com

FEATURES

- Communicates at regular and overdrive 1-Wire[®] speeds
- Supports stiff 5V pullup for EEPROM, sensors, and crypto iButton[®]
- Slew rate controlled 1-Wire timing and active pullup to accommodate long 1-Wire network lines and reduce radiation
- Programmable 1-Wire timing and driver characteristics accommodate a wide range of 1-Wire network configurations
- Low- to high-level command types, including macros, for generating 1-Wire communication
- Crystal oscillator timebase provides precision timed 1-Wire waveforms
- High-speed 12Mbps Universal Serial Bus (USB) interface
- Integrated USB-compliant transceiver
- Supports USB remote wake-up from a 1-Wire device event to resume a suspended host system
- 0°C to +70°C operating temperature range

PIN ASSIGNMENT



24-Pin SO
Top View
(300-mil)

ORDERING INFORMATION

| PART NUMBER | TEMP RANGE | PIN-PACKAGE |
|-------------|--------------|-----------------|
| DS2490S+ | 0°C to +70°C | 24 SO (300 mil) |
| DS2490S+T&R | 0°C to +70°C | 24 SO (300mil) |

+ Denotes a lead(Pb)-free/RoHS-compliant package.

T&R = Tape and reel.

DESCRIPTION

The DS2490 is a bridge chip that enables communication between a USB host system and a 1-Wire bus. It provides regular, overdrive, and flexible 1-Wire communication speeds and a full-speed 12Mbps connection to USB. USB vendor-specific commands defined in this specification are used to control the DS2490 and communicate with attached 1-Wire devices. A functional block diagram of the DS2490 is shown in Figure 1.

1-Wire and iButton are registered trademarks of Maxim Integrated Products, Inc.

SIGNAL SUMMARY Table 1

| SIGNAL NAME | TYPE | FUNCTION |
|--------------------------|------|---|
| VD | PWR | Power supply input for digital and 1-Wire functions. Range: $5.0 \pm 10\%$ |
| VD2 | PWR | Second VD supply, must be tied to VD externally to the IC (the two pins are not tied together inside the package). |
| VB | PWR | Power supply input for USB functions. Range: $3.3V \pm 10\%$ supply regulated from USB supplied VBUS. |
| D+ | I/O | USB data—non-inverted of differential data pair. |
| D- | I/O | USB data—inverted signal of differential data pair. |
| 1-Wire | I/O | 1-Wire input/output. |
| PMOD | I | Reserved for future use. Must be tied to GND. |
| $\overline{\text{SUSO}}$ | O | Suspend Output—buffered USB suspend-state output from USB device controller. When HIGH the USB is in an active non-suspended state, when LOW the USB has entered a suspended state. This is an open drain output and requires an external pullup. |
| XI | I | Crystal input. Use a 12.0 MHz, fundamental mode, parallel-resonant crystal. A 12.0MHz CMOS clock source may also be used. |
| XO | O | Crystal output. Connect to other side of crystal 1 if used. |
| GND | PWR | Ground reference and ground return for 1-Wire bus. |
| NC | | No connect. For factory use or reserved, do not connect to these pins. |

RELATED DOCUMENTS

This specification uses terms from and references or complies with the *Universal Serial Bus Specification v1.1*, which may be obtained from the USB Implementers Forum website: www.usb.org. The USB specification is considered to be part of the DS2490 specification.

DOCUMENT ORGANIZATION

The remainder of this document is organized into the following major sections:

| SECTION | SUMMARY |
|------------------------|---|
| OVERVIEW | Device functional summary and application examples |
| 1-WIRE I/E CONTROLLER | Edge control and timing diagrams of 1-Wire signals |
| USB COMMUNICATION | Configuration model, core and vendor-specific command summaries |
| MODE COMMANDS | Commands used to configure 1-Wire interface operational characteristics |
| CONTROL COMMANDS | Commands used to control 1-Wire communication command processing |
| COMMUNICATION COMMANDS | Commands used to communicate with an attached 1-Wire device |
| DEVICE FEEDBACK | Technique to obtain device status information |
| USB TRANSCEIVER | Transceiver connection requirements |
| OSCILLATOR | Oscillator connection requirements |
| SUSPEND OUTPUT | $\overline{\text{SUSO}}$ signal operation and purpose |

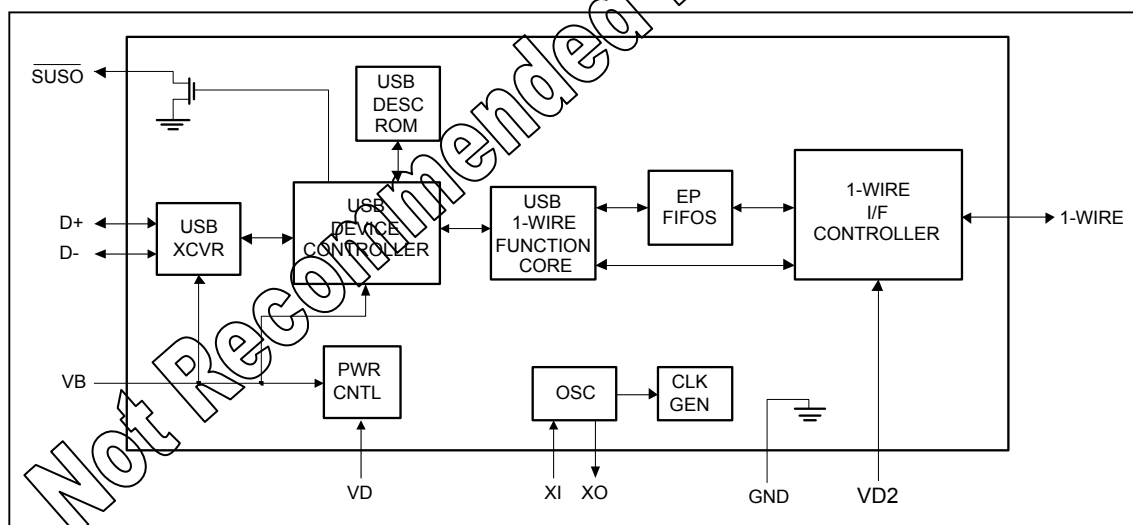
| SECTION | SUMMARY |
|----------------------------|--|
| ELECTRICAL CHARACTERISTICS | DC and AC specifications |
| APPLICATION INFORMATION | HW application example |
| APPENDIX 1 | CONTROL COMMANDS—USB setup packet encoding |
| APPENDIX 2 | COMMUNICATION COMMANDS—USB setup packet encoding |
| APPENDIX 3 | MODE COMMANDS—USB setup packet encoding |
| APPENDIX 4 | USB command and command type constant codes |

OVERVIEW

The DS2490 directly interfaces a USB port to a 1-Wire bus. As shown in Figure 1, the DS2490 incorporates a USB physical interface, a USB device controller coupled with a 1-Wire-specific USB function core, and a 1-Wire bus interface controller. The 1-Wire interface controller shapes the slopes of the 1-Wire waveforms, applies strong pullup to 5V, and reads the 1-Wire bus using a non-TL threshold to maximize the noise margin for best performance on large 1-Wire Networks. 1-Wire waveform timing is accurately controlled with a crystal-based oscillator.

The DS2490 also supports USB remote wake-up which enables the DS2490-based USB peripheral to send resume signaling to a suspended host system. If the remote wake-up function is enabled and the host system is in a suspended state, a 1-Wire device attachment will cause the DS2490 to perform a host system wake-up and allow the 1-Wire device to be serviced.

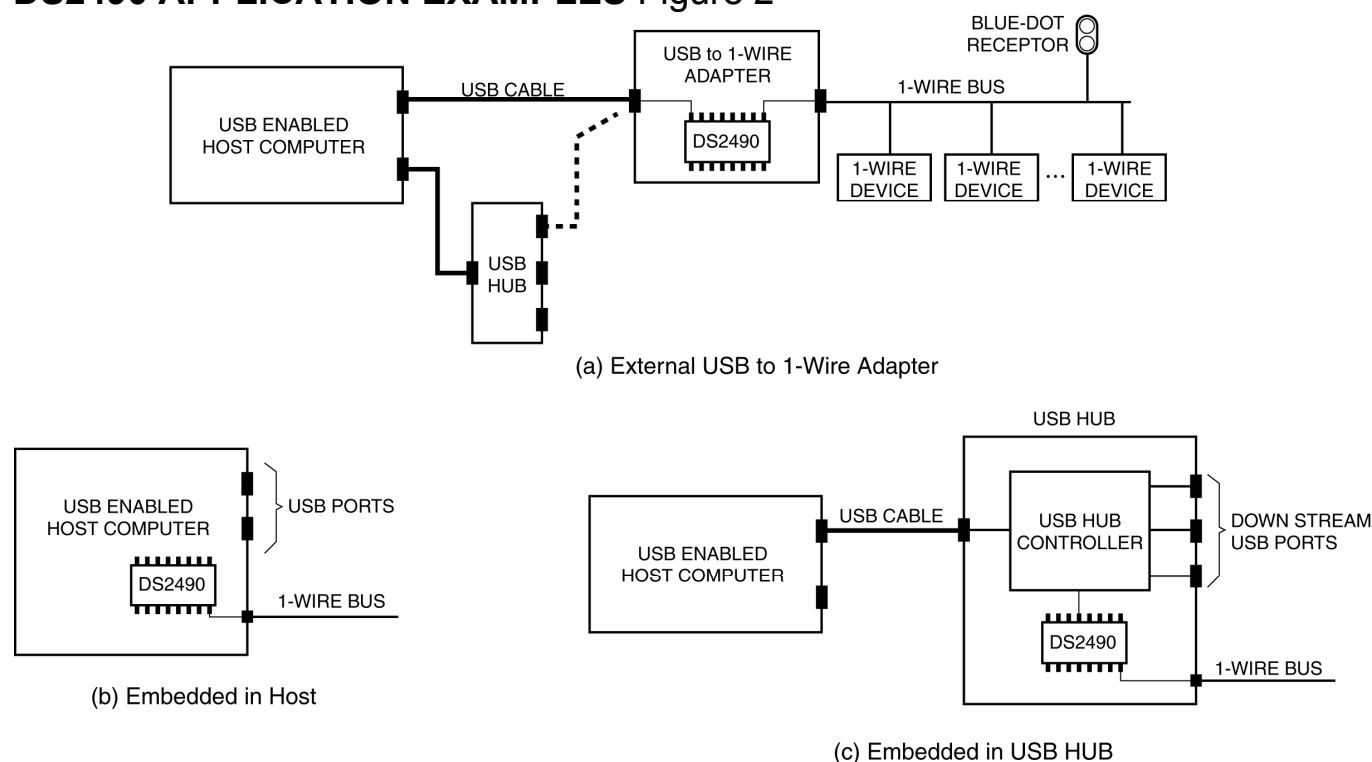
DS2490 FUNCTIONAL BLOCK DIAGRAM Figure 1



Typical application examples of the DS2490 are shown in Figure 2 (a to c). As shown in all the examples, all host control and communication with the device is accomplished over a USB communication link. A USB vendor-specific command set, as defined in this document, is used to select operational modes (MODE COMMANDS), control command processing (CONTROL COMMANDS), and communicate over the 1-Wire interface (COMMUNICATION COMMANDS). Shown in Figure 2, example (a) is a DS2490-based USB peripheral application. The peripheral function is a USB to 1-Wire adapter and provides both USB and 1-Wire I/O connections. In this example, the peripheral is attached to the USB enabled host computer either directly at a root port or through a USB hub. The 1-Wire bus interface provided by the DS2490 supports all 1-Wire devices manufactured by Dallas Semiconductor as well as

the various 1-Wire bus topologies simple multi-drop to complex 1-Wire Network. Example (b) and (c) in Figure 2 are variations of example (a) in which the DS2490 is embedded in the host computer or a USB HUB.

DS2490 APPLICATION EXAMPLES Figure 2



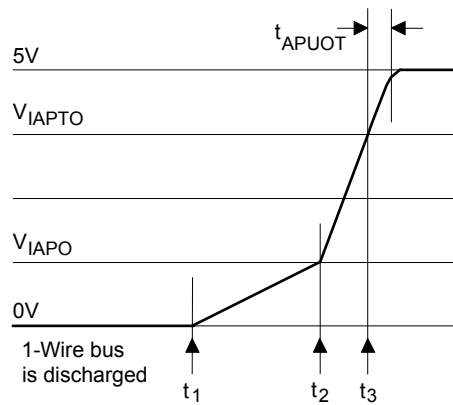
1-WIRE INTERFACE CONTROLLER

1-Wire communication commands sent to the DS2490 are ultimately processed by the 1-Wire interface controller. One of the tasks of the interface controller is to actively shape the edges of the 1-Wire communication waveforms. This speeds up the recharging of the 1-Wire bus (rising edges) and reduces ringing of long lines (falling edges). The circuitry for shaping rising edges is always active. The slew rate of falling edges is actively controlled only at flexible speed and requires the parameter for slew rate control being different from its power-on default value. See the MODE COMMANDS section for parameter control and power-on defaults.

All Rising Edges

The active pullup of the rising edges reduces the rise time on the 1-Wire bus significantly compared to a simple resistive pullup. Figure 3 shows how the DS2490 is involved in shaping a rising edge.

ACTIVE PULLUP Figure 3

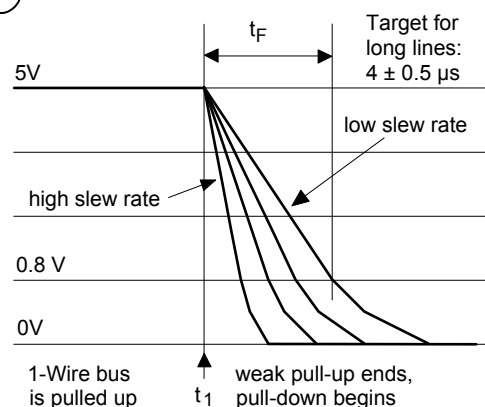


The circuit operates as follows: At t_1 , the pulldown (induced by the DS2490 or a device on the bus) ends. From this point on the 1-Wire bus is pulled high by the weak pullup current I_{WEAKPU} provided by the DS2490. The slope is determined by the load on the bus and the value of the pullup current. At t_2 , the voltage crosses the threshold voltage V_{IAPO} . Now, the DS2490 switches over from the weak pullup current I_{WEAKPU} to the higher current I_{ACTPU} . As a consequence, the voltage on the bus now rises faster. As the voltage on the bus crosses the threshold V_{IAPTO} at t_3 , a timer is started. As long as this timer is on (t_{APUOT}), the I_{ACTPU} current will continue to flow. After the timer is expired, the DS2490 will switch back to the weak pullup current.

Falling Edges (DS2490-initiated)

Whenever the DS2490 begins pulling the 1-Wire bus low to initiate a time slot, for example, it first turns off the weak pullup current I_{WEAKPU} . Then, at regular and overdrive speed it will generate a falling edge at a slew rate of typically 15 V/ μ s. This value is acceptable for short 1-Wire busses and adequate for communication at overdrive speed. For 1-Wire networks of more than roughly 30m length, flexible speed should always be used. One of the parameters that is adjustable at flexible speed is the slew rate of DS2490-initiated falling edges. The effect of the slew rate control is shown in Figure 4.

SLEW RATE CONTROL Figure 4



Extensive tests have shown that 1-Wire networks with lengths of up to 300m will perform best if the fall time t_F is in the range of $4 \pm 0.5 \mu$ s. This translates into a slew rate of approximately 1 V/ μ s. This slew rate is typically achieved by selecting a PULLDOWN SLEW RATE parameter code of 0 x 4 (see MODE COMMANDS). If the actual measured fall time is longer than the target value, a parameter code of 0 x 3

or lower should be used. If the fall time is shorter, a parameter code of 0x5 or higher should be used. Once determined, the value code for the PULLDOWN SLEW RATE control parameter should be stored in the host and always be loaded into the DS2490 after a power-on or master reset cycle.

1-WIRE TIMING DIAGRAMS

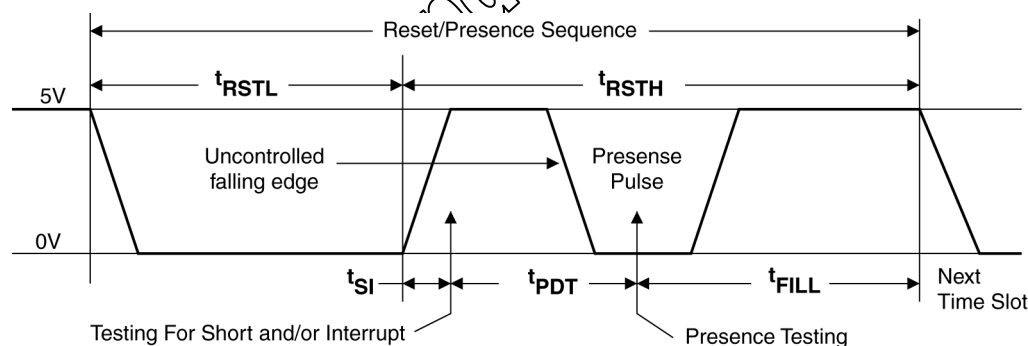
This section explains the 1-Wire bus waveforms generated by the DS2490. First, the communication waveforms such as the reset/presence detect sequence and read/write data time slots are discussed followed by a detailed description of the Pulse function under various conditions.

1-Wire Communication Wave Forms

One of the major features of the DS2490 is that it relieves the host from generating the timing of the 1-Wire signals and sampling the 1-Wire bus at the appropriate times. The reset/presence detect sequence is shown in Figure 5. This sequence is composed of four timing segments: the reset low time t_{RSTL} , the short/interrupt sampling offset t_{SI} , the presence detect sampling offset t_{PDT} and a delay time t_{FILL} . The timing segments t_{SI} , t_{PDT} and t_{FILL} comprise the reset high time t_{RSTH} where 1-Wire slave devices assert their presence or interrupt pulse. During this time, the DS2490 pulls the 1-Wire bus high with a weak pullup current.

Reset/presence timing values are shown in Figure 5. The values of all timing segments for all 1-Wire speed options are shown in the table. Since the reset/presence sequence is slow compared to the time slots, the values for regular and flexible speed are the same. Except for the falling edge of the presence pulse, all edges are controlled by the DS2490. The shape of the uncontrolled falling edge is determined by the capacitance of the 1-Wire bus and the number, speed, and sink capability of the slave devices connected.

RESET/PRESENCE DETECT Figure 5



NOMINAL TIMING VALUES

| SPEED | t_{RSTL} | t_{SI} | t_{PDT} | t_{FILL} | t_{RSTH} |
|-----------|-------------|-----------|------------|-------------|-------------|
| REGULAR | 512 μ s | 8 μ s | 64 μ s | 512 μ s | 584 μ s |
| OVERDRIVE | 64 μ s | 2 μ s | 8 μ s | 64 μ s | 74 μ s |
| FLEXIBLE | 512 μ s | 8 μ s | 64 μ s | 512 μ s | 584 μ s |

Upon executing a 1-WIRE RESET command (see COMMUNICATION COMMANDS), the DS2490 pulls the 1-Wire bus low for t_{RSTL} and then lets it go back to 5V. The DS2490 will now wait for the short/interrupt sampling offset t_{SI} to expire and then test the voltage on the 1-Wire bus to determine if there is a short or an interrupt signal. If there is no short or interrupt the DS2490 will wait for t_{PDT} and test the voltage on the 1-Wire bus for a presence pulse. Regardless of the result of the presence test, the

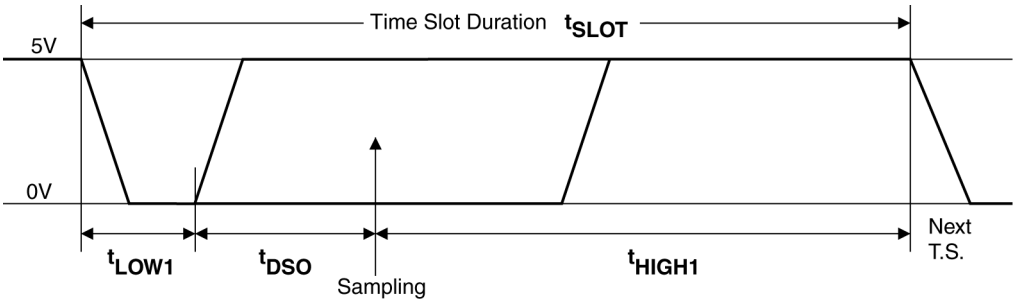
DS2490 will then wait for t_{FILL} to expire and then, depending on the value of embedded 1-WIRE RESET command bits PST, NTF, and ICP, generate a command response byte that is available to the host.

If the test for interrupt or short reveals a logic 0, the DS2490 will wait for $4096\mu\text{s}$ and then test the 1-Wire bus again. If a logic 0 is detected, the 1-Wire bus is shorted and the DS2490 feedback response for the 1-WIRE RESET communication command will indicate a short detection. If a logic 1 is detected, the device will wait for t_{FILL} to expire, after which it will load the feedback response value for the 1-WIRE RESET command with an alarming presence pulse detect value. See the DEVICE FEEDBACK section for additional details. No additional testing for a presence pulse will be done. The DS2490 will perform the short/interrupt testing as described also at overdrive speed, although interrupt signaling is only defined for regular speed.

As shown in Figure 6, a Write-1 and Read Data time slot is comprised of the segments t_{LOW1} , t_{DSO} , and t_{HIGH} . During Write-1 time slots, after the Write-1 low time (t_{LOW1}) expires, the DS2490 waits for the duration of the data sample offset and then samples the 1-Wire voltage to read the response. After this, the waiting time t_{HIGH1} must expire before the time slot is complete. As shown in Figure 7, a Write-0 time slot consists of the two segments t_{LOW0} and t_{REC0} .

Since the defaults for regular speed exceed the 1-Wire sampling maximum for most 1-Wire devices ($t_{\text{LOW1}} + t_{\text{DSO}} = 20\mu\text{s}$) it is recommended to always use flexible speed. The minimum flexible speed settings $t_{\text{LOW1}} = 4\mu\text{s}$ and $t_{\text{DSO}} = 10\mu\text{s}$ suffice for most applications. If the network is large or heavily loaded, Write-1 low time (t_{LOW1}) should be extended to more than $8\mu\text{s}$ to allow the 1-Wire bus to completely discharge. Since a large or heavily loaded network needs more time to recharge, it is also recommended to delay sampling the bus for reading. A higher Data Sample Offset value (t_{DSO}) will increase the voltage margin and also provide extra energy to the slave devices when generating a long series of Write-0 time slots. However, the total of $t_{\text{LOW1}} + t_{\text{DSO}}$ should not exceed $15\mu\text{s}$. Otherwise, the slave device responding may have stopped pulling the bus low when transmitting a logic 0. Note that some long line loading conditions can extend the line recovery so the total $t_{\text{LOW1}} + t_{\text{DSO}}$ can be extended. Care must be taken to not violate the t_{MSR} of the attached 1-Wire devices.

WRITE 1 AND READ DATA TIME SLOT Figure 6

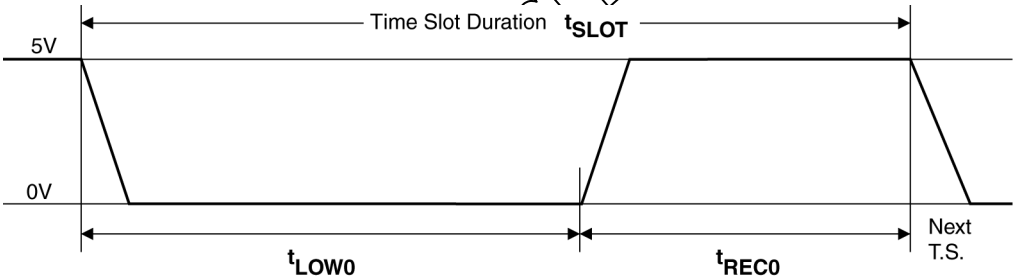


NOMINAL TIMING VALUES

| SPEED | t_{LOW1} | t_{DSO} | t_{HIGH1} | t_{SLOT} |
|-----------|-------------------------|--------------------------|-------------|--------------------------|
| REGULAR | 4 μ s | 16 μ s | 46 μ s | 66 μ s |
| OVERDRIVE | 1.4 μ s | 1 μ s | 8 μ s | 10.4 μ s |
| FLEXIBLE* | 4 μ s to 11 μ s | 10 μ s to 24 μ s | 46 μ s | 60 μ s to 81 μ s |

*Powerup defaults for Flexible speed: $t_{LOW1} = 8\mu$ s, $t_{DSO} = 18\mu$ s

WRITE 0 TIME SLOT Figure 7



NOMINAL TIMING VALUES

| SPEED | t_{LOW0} | t_{REC0} | t_{SLOT} |
|-----------|-------------|--------------------------|--------------------------|
| REGULAR | 56 μ s | 16 μ s | 72 μ s |
| OVERDRIVE | 7.4 μ s | 3 μ s | 10.4 μ s |
| FLEXIBLE* | 56 μ s | 10 μ s to 24 μ s | 66 μ s to 80 μ s |

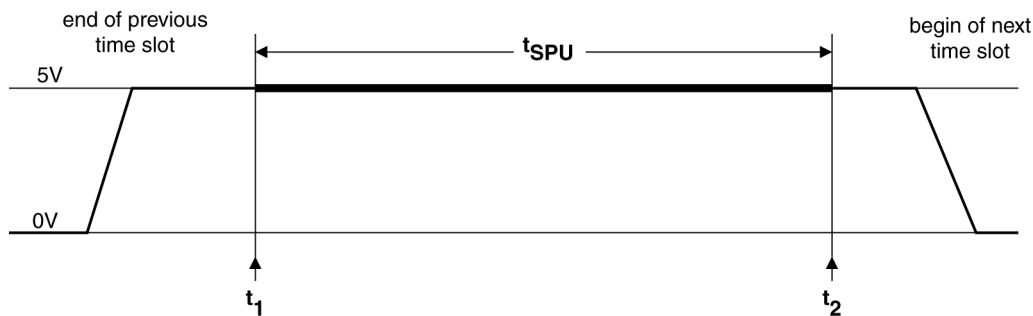
*Powerup defaults for Flexible speed: $t_{REC0} = 18\mu$ s

Pulse Wave Forms

The PULSE COMMUNICATION COMMAND can be used to generate a strong pullup to 5V. The duration of the pulses is specified with the STRONG PULLUP DURATION mode register. Figure 8 shows timing of the pulse. For predefined pulse durations, t_{SPU} is a known value, for semi-infinite and infinite durations these pulse times vary and depend on the characteristics of attached 1-Wire devices and/or host processor intervention. See the MODE COMMAND section for details on duration times. As shown in the figure, at t_1 processing of the PULSE command begins, at t_2 the pulse ends.

Certain applications may require a duration for a strong pullup that cannot be realized using one of the predefined values (see MODE COMMANDS section for values). Selecting infinite duration allows the host to generate pulses of any length. As a consequence, however, the host becomes responsible to actively control the duration of the pulse. Failing to do so may require a power-on reset or master-reset cycle of the DS2490. For this reason, infinite duration should only be used if absolutely necessary. The HALT EXECUTION WHEN DONE or HALT EXECUTION WHEN IDLE control commands are used to terminate an infinite duration pulse.

STRONG PULLUP TO 5V, PRE-DEFINED DURATION Figure 8

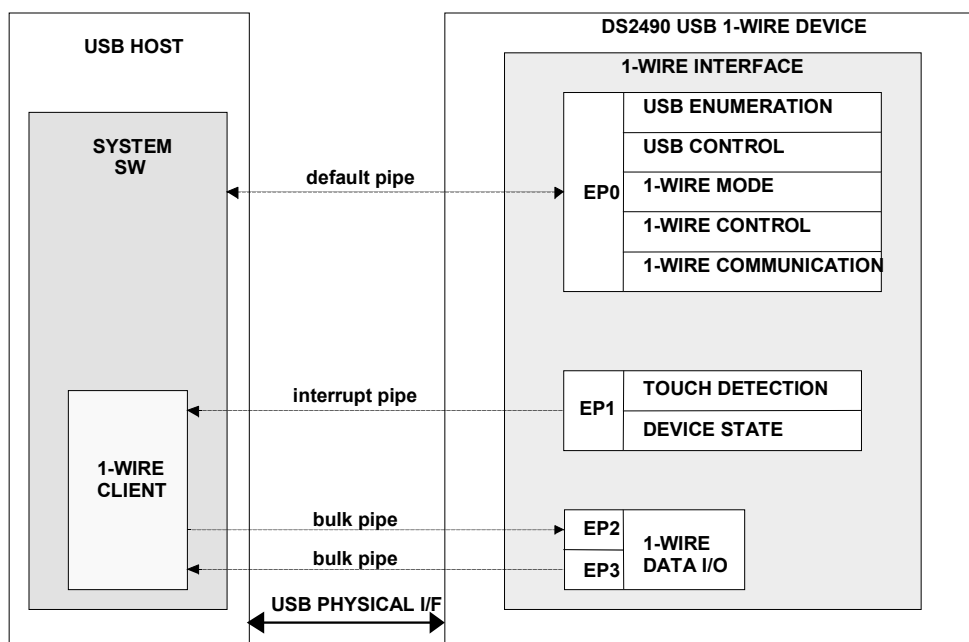


USB COMMUNICATION

Communication with the DS2490 is performed using USB standard requests, also referred to as core commands or requests in this document, and DS2490 vendor specific USB commands. All command communication with the DS2490 is performed over the default control pipe. Non-command communication, including 1-Wire device data I/O and DS2490 status information, occurs in a vendor-specific fashion over bulk and interrupt pipes. Configuration of the DS2490 USB controller is shown in Figure 9. As shown, a Dallas Semiconductor vendor-specific 1-Wire device exists with a single 1-Wire interface. The USB vendor-specific 1-Wire function of the DS2490 will be implemented and controlled by the device driver at the device level of the device class hierarchy, i.e., vendor-specific commands, will be directed to the device level. Within the 1-Wire interface, four endpoints exist to control and communicate with the device. Four alternate settings of the 1-Wire interface exist that correspond to different operating modes for the endpoints. Endpoint and interface descriptions are summarized in the following paragraphs.

The DS2490 provides USB remote wake-up capability. Per the USB specification, if a device supports remote wake-up, it must also provide the ability to enable or disable the remote wake-up function. In addition, the remote wake-up feature must be disabled as the power up default or device reset state. When the remote wake-up feature is enabled, the DS2490 will send USB resume signaling to a suspended host system upon detecting a 1-Wire device attachment. It then becomes the responsibility of host system software to resume operation and to determine what servicing is required for the 1-Wire device.

USB CONFIGURATION Figure 9



Endpoint (EP) Summary Description

EP0 is the endpoint for the bi-directional default control pipe. It is used for the USB enumeration process, USB core request communication, and all DS2490 specific command communication.

EP1 is the endpoint for an interrupt pipe (device to host) and is used to relay DS2490 status register data and specific command execution completion and/or error information to the host. It is also used to inform the host of 1-Wire device-attach detection. The polling period requested for EP1 is either 10ms or 1ms, depending on the alternate interface settings of the 1-Wire interface. The default polling period for EP1 is 10ms.

EP2 is an endpoint for a bulk data out pipe (data from host) and is used to transmit 1-Wire device data from the host to the DS2490. The information received at this endpoint will be transmitted as data on the 1-Wire bus.

EP3 is an endpoint for a bulk data in pipe (data to host) and is used to send data received by the DS2490 from the 1-Wire bus back to the host for processing.

1-Wire Interface Summary Description

The interface is the USB collection point for the four endpoints. Four alternate settings exist for the 1-Wire interface which correspond to different operational modes for the pipes to EP1, EP2, and EP3. As shown in Table 2, the alternate settings specify different polling periods for the interrupt pipe and different maximum packet sizes for the two bulk pipes.

1-Wire Interface Alternate Setting Summary Table 2

| Alt Setting | EP1 poll interval | EP2/EP3 max packet size | Alternate Setting Description |
|-------------|----------------------|----------------------------|--|
| 0 | 10ms | 16 bytes | Long interrupt polling interval, small packet size for bulk pipes |
| 1 | 10ms | 64 bytes | Long interrupt polling interval, large packet size for bulk pipes |
| 2 | 1ms | 16 bytes | Short interrupt polling interval, small packet size for bulk pipes |
| 3 | 1ms | 64 bytes | Short interrupt polling interval, large packet size for bulk pipes |

Endpoint Feature Summary Table 3

| Endpoint # | Transaction Type | Direction ¹ | Max. Packet Size |
|------------|------------------|------------------------|------------------------|
| 0 | CONTROL | IN/OUT | ALT I/F-0..3: 8 bytes |
| 1 | INTERRUPT | IN | ALT I/F-0..3: 32 bytes |
| 2 | BULK | OUT | ALT I/F-0: 16 bytes |
| | | | ALT I/F-1: 64 bytes |
| | | | ALT I/F-2: 16 bytes |
| | | | ALT I/F-3: 64 bytes |
| 3 | BULK | IN | ALT I/F-0: 16 bytes |
| | | | ALT I/F-1: 64 bytes |
| | | | ALT I/F-2: 16 bytes |
| | | | ALT I/F-3: 64 bytes |

1. Direction in this table is in reference to the Host

USB Core Commands Table 4

| Standard Device Requests | Target Element | Range | Notes |
|--------------------------|-----------------------------------|--------------------------|-------|
| SET_ADDRESS | Device | 0x01 - 0x1F | |
| SET_CONFIGURATION | Device | 0x00 - 0x01 | 1 |
| GET_CONFIGURATION | Device | 0x00 - 0x01 | |
| GET_DESCRIPTOR | Device | | 2 |
| GET_INTERFACE | Interface 0 | Interface 0: 0x00 - 0x03 | 3 |
| SET_INTERFACE | Interface 0 | Interface 0: 0x00 - 0x03 | 3 |
| SET_FEATURE | Device | | 4 |
| CLEAR_FEATURE | Device | | 4 |
| GET_STATUS | Device, Interfaces, Endpoints 0-3 | | 5, 6 |

Notes:

1. The only valid configuration values are 0 and 1 for the SET_CONFIGURATION request. Configuration value 0 corresponds to the non-configured state.
2. Only the DEVICE and CONFIGURATION descriptor types are supported for the GET_DESCRIPTOR request.
3. Interface 0 is the only valid interface value for the DS2490.
4. The only valid feature selector for the command is DEVICE_REMOTE_WAKE-UP.
5. The DS2490 is intended, at a minimum, to be partially bus powered. Remote wake-up can be enabled or disabled. GET_STATUS requests to the device will return a logic 0 for the self-powered bit field and the current state value for the remote wake-up bit.
6. There is no USB core level status information defined for interfaces in USB Specification v1.1. The DS2490 will always return 0 value data for GET_STATUS requests to interfaces.

DS2490 Vendor-Specific USB Commands

Three different vendor-specific command types exist to control and communicate with the DS2490: Control, Communication, and Mode. Control commands are used to manage various device functions including the processing of communication commands, buffer clearing, and SW reset. Communication commands are used for 1-Wire data and command I/O. Mode commands are used to establish the 1-Wire operational characteristics of the DS2490 such as slew rate, low time, strong pullup, etc. Control, Communication and Mode commands, like USB core requests, are communicated over the default control pipe at EP0. With one exception as noted below, each command and any associated parameter data are individually formatted into the 8-byte control transfer setup packet as follows:

| bmRequestType | bRequest | wValue | wIndex | wLength |
|----------------------------------|--------------------------|------------------------------------|---------------------------------|-------------------------|
| Request type bit map (1 byte) | Command Type (1 byte) | Command (2 bytes) See Note 1 | Command Parameters (2 bytes) | (2 bytes) See Note 2 |

Notes:

1. The one exception to **wValue** formatting is for the READ STRAIGHT Communication command. This command requires three parameter bytes that exceed the 2-byte parameter encoding space available in the **wIndex** field. For this command only, the **wValue** field will be formatted as 1 command byte and 1 parameter byte. See the command description in Appendix 2 for details.
2. The **wLength** field is used by the GET COMM CMDS Control command to specify the number of command/parameter bytes to retrieve from the DS2490 command FIFO; see the command description for specifics. In all other cases the **wLength** field is not used and must be set to 0x0000.

The bit-mapped **bmRequestType** field identifies the characteristics of the USB command per Chapter 9 of the USB Specification. Fields included in this command describe the direction of the transfer, type of request (core, class, or vendor specific) and the target of the command recipient (device, interface, or endpoint). The only variation on the **bmRequestType** field for the different DS2490 vendor-specific command will be for the transfer direction: host to device, or device to host; the recipient for all commands will be the device.

The **bRequest** field contains a 1-byte constant identifying which of the three supported DS2490 command types is being sent. The command constants are defined in Appendix 4 and are summarized as follows:

| <u>Command Type</u> | <u>Description</u> |
|----------------------------|--|
| CONTROL_CMD | 1-Wire interface control commands |
| COMM_CMD | 1-Wire interface communication commands |
| MODE_CMD | 1-Wire interface operational mode commands |

The 2-byte **wValue** field is encoded with the specific command. For Control and Mode commands, the field value corresponds to a specific command constant as defined in Appendix 4. For Communication commands, the field has additional embedded command parameters as indicated in Appendix 2. The exception is the READ STRAIGHT command as described previously.

The 2-byte **wIndex** is used to hold additional command parameter data when required by the specific command.

The **wLength** field is used in the control transfer setup packet to specify the number of bytes to be sent during the data stage of a control transfer. The “GET COMM CMDS” control command is the only DS2490 command that uses the data stage of the control transfer. Typically, all command data is embedded in the setup stage. Except for the GET COMM CMDS, this field must be set to 0x0000.

DS2490 Control, Communication, and Mode vendor-specific command descriptions and formatting are defined in Appendices 1, 2, and 3. Vendor-specific commands not defined in the appendices are not supported by the DS2490 and the device will respond with a STATE if an unsupported command is received.

MODE COMMANDS

DS2490 1-Wire characteristics and features (speed, durations, slew rate, etc.) are controlled with discrete Mode commands and/or embedded Communication command parameters. Parameter values and enable/disable settings are used for characteristic control. Mode settings are stored in the DS2490 State Registers and can be read at any time or at the polling interval of EP1. See the section “DEVICE FEEDBACK” for State Register details.

Enable/disable settings are used to control two global features:

- Strong pullup to +5V
- Dynamic 1-Wire bus communication speed change through a communication command

The settings for these three global controls are accessible only through Mode commands; the features are either enabled or disabled with the appropriate Mode command. Specific Communication commands exist to issue a strong pullup, or speed change. When the corresponding setting is enabled the feature is usable as part of a communication command when disabled the feature is not usable.

Parameter values and codes are used to specify and control 1-Wire bus characteristics:

- 1-Wire bus communication speed
- +5V strong pullup duration
- Pulldown slew rate
- Write-1 low time
- Data sample offset/Write-0 Recovery time

These five settings are controllable with discrete Mode commands or as embedded command/parameter values in Communications commands.

As listed in Table 5 and summarized previously, there are eight Mode commands. The USB control transfer setup packet coding to transmit these commands is detailed in Appendix 3. Mode commands are immediately processed by the DS2490 when they are received. The DS2490 power-on default values for these parameters are as listed in Table 11. Each command and parameter control is detailed in the following paragraphs.

Mode Command Set Summary Table 5

| Command | Function |
|------------------------|--|
| ENABLE PULSE | Enable/disable 1-Wire strong pullup pulse to 5V. |
| ENABLE SPEED CHANGE | Enable/disable dynamic change of the 1-Wire speed through a communication command. |
| 1-WIRE SPEED | Communication speed of the 1-Wire bus. |
| STRONG PULLUP DURATION | Duration of 1-Wire strong pullup. |
| PULLDOWN SLEW RATE | Slew rate of 1-Wire pulldown. |
| WRITE-1 LOW TIME | Duration of 1-Wire Write-1 low time. |
| DSOW0 RECOVERY TIME | Recovery time for the 1-Wire data sample offset Write-0 transaction. |

Mode Command Set Descriptions:

ENABLE PULSE—This command is used to enable or disable a 1-Wire strong pullup pulse to 5V. One bit position in the parameter byte is used to control the enabled/disabled state for the pulse. The pulse is enabled when the respective bit is set to a 1 and disabled when set to a 0. *The DS2490 power-up default state for strong pullup is disabled.*

ENABLE SPEED CHANGE—This command is used to enable or disable a 1-Wire communication speed change. Enabled when a TRUE parameter value is passed with the command, disabled with a FALSE. *The DS2490 power-up default state for speed change is disabled.*

1-WIRE SPEED—This command is used to set the speed of 1-Wire communication; three settings are possible. The parameter codes to select the desired or required speed are as listed in Table 6. As shown, speed codes 0x3..0xF are undefined. The DS2490 decodes the 3 LSBs of the 1-Wire speed code. Sending a code other than the defined values in Table 6 will result in undefined behavior. *The DS2490 power-up default communication speed is regular.*

1-Wire Bus Speed Codes Table 6

| Code | Speed | Data Rate |
|----------|-----------|---|
| 0x0 | Regular | 65 μ s time slot (15.4kbps) |
| 0x1 | Flexible | 65 μ s to 72 μ s time slot (13.9kbps to 15.4kbps) |
| 0x2 | Overdrive | 10 μ s time slot (100kbps) |
| 0x3..0xF | reserved | undefined |

STRONG PULLUP DURATION—This command is used to set the time duration of a 1-Wire strong pullup. As shown in Table 7, the time is controlled with an unsigned 8-bit binary number between 0x00 and 0xFE which specifies the duration in multiples of 16ms. A value of 0x01 specifies 16ms, 0x02 equals 32ms, etc. A value of 0x00 specifies infinite duration. Parameter value 0xFF is reserved and will cause the device to deliver a pullup duration of $<1\mu\text{s}$. To terminate an infinite duration pullup use either the HALT EXECUTION WHEN DONE or HALT EXECUTION WHEN IDLE Control commands as described in Appendix 1. *The DS2490 power-up default strong pullup duration register value is 512ms.*

Strong Pullup Duration Codes Table 7

| Value | Nominal Strong Pullup Duration |
|-------|--------------------------------|
| 0x00 | infinite |
| 0x01 | 16ms |
| 0x02 | 32ms |
| ... | ... |
| 0xFE | 4.064s |
| 0xFF | reserved |

PULLDOWN SLEW RATE—This command is used to select the pulldown slew rate for 1-Wire bus Flexible Speed operation; eight pulldown slew rates are possible. The parameter codes to select the desired or required slew rate are as listed in Table 8. The slew rate numbers in the table represent nominal values. The nominal pulldown slew rate for Regular speed is $0.83\text{V}/\mu\text{s}$ and for Overdrive speeds it is $15\text{V}/\mu\text{s}$. The DS2490 decodes the three LSBs of the slew rate code. Sending a code other than the defined values in Table 8 will result in undefined behavior. *The DS2490 pulldown slew rate power-up default value for Flexible speed is $0.83\text{V}/\mu\text{s}$.*

Flexible Speed Pulldown Slew Rate Codes Table 8

| Code | Nominal Pulldown slew rate |
|----------|----------------------------|
| 0x0 | $15\text{V}/\mu\text{s}$ |
| 0x1 | $2.20\text{V}/\mu\text{s}$ |
| 0x2 | $1.65\text{V}/\mu\text{s}$ |
| 0x3 | $1.37\text{V}/\mu\text{s}$ |
| 0x4 | $1.10\text{V}/\mu\text{s}$ |
| 0x5 | $0.83\text{V}/\mu\text{s}$ |
| 0x6 | $0.70\text{V}/\mu\text{s}$ |
| 0x7 | $0.55\text{V}/\mu\text{s}$ |
| 0x8..0xF | reserved |

WRITE-1 LOW TIME—This command is used to select the Write-1 low time for 1-Wire bus Flexible speed operation; eight Write-1 low time durations are possible. The parameter codes to select the desired or required low time are as listed in Table 9. The low time numbers in the table represent nominal values. The nominal Write-1 Low Time for Regular speed is $4\mu\text{s}$, at Overdrive speed it is $1.4\mu\text{s}$. The DS2490 decodes the three LSBs of the low time code. Sending a code other than the defined values in Table 9 will result in undefined behavior. *The DS2490 Write-1 Low Time power-up default value for Flexible speed is $8\mu\text{s}$.*

Flexible Speed Write-1 Low Time Codes Table 9

| Code | Nominal Write-1 Low Time |
|----------|--------------------------|
| 0x0 | 4 μ s |
| 0x1 | 5 μ s |
| 0x2 | 6 μ s |
| 0x3 | 7 μ s |
| 0x4 | 8 μ s |
| 0x5 | 9 μ s |
| 0x6 | 10 μ s |
| 0x7 | 11 μ s |
| 0x8..0xF | reserved |

DSOW0 RECOVERY TIME—This command is used to select the Data Sample Offset (t_{DSO}) / Write-0 recovery (t_{W0R}) time (DSO/W0R) for 1-Wire bus Flexible Speed operation; eight DSO/W0R times are possible. The parameter codes to select the desired or required recovery time are as listed in Table 10. The numbers in the table represent nominal values. The nominal DSO/W0R time for Regular speed is 16 μ s, for Overdrive speed the Data Sample Offset is 1 μ s and the Write-0 Recovery Time is 3 μ s. The DS2490 decodes the three LSBs of the DSO/W0R code. Sending a code other than the defined values in Table 10 will result in undefined behavior. *The DS2490 DSO/W0R power-up default value for Flexible speed is 18 μ s.*

Flexible Speed DSO/ W0R Time Codes Table 10

| Code | Nominal DSO/W0R Time |
|----------|----------------------|
| 0x0 | 10 μ s |
| 0x1 | 12 μ s |
| 0x2 | 14 μ s |
| 0x3 | 16 μ s |
| 0x4 | 18 μ s |
| 0x5 | 20 μ s |
| 0x6 | 22 μ s |
| 0x7 | 24 μ s |
| 0x8..0xF | reserved |

Power-on Default Mode Values

Table 11 summarizes the power-up default values for the various DS2490 mode registers. Note that the power-up default communication speed is regular and several of the values listed in Table 11 are specific to Flexible speed operation. To change a value the appropriate Mode or Communication command must be sent to the device.

1-Wire Power-On Default Modes Table 11

| Setting/Parameter | Power-On Default Value |
|--|------------------------------|
| Strong pullup to 5V | FALSE |
| Dynamic Speed Change | FALSE |
| 1-Wire Bus Communication Speed | 0x0 (Regular Speed, ~16kbps) |
| +5V Strong Pullup Duration | 0x20 (512ms) |
| Pulldown Slew Rate Control | 0x5 (0.83V/ μ s) |
| Write-1 Low Time | 0x4 (8 μ s) |
| Data Sample Offset / Write-0 Recovery Time | 0x4 (18 μ s) |

CONTROL COMMANDS

DS2490 Control commands are used to control the processing of Communication commands/data and can also be used to issue a device Reset. As listed in Table 12 there are 9 Control commands. The USB control transfer setup packet coding to transmit these commands is detailed in Appendix 1. Like Mode commands, Control commands are immediately processed by the DS2490 when they are received.

Control Command Set Summary Table 12

| Command | Function |
|--------------------------|--|
| RESET DEVICE | Perform a hardware reset. |
| START EXECUTION | Start execution of Communication commands. |
| RESUME EXECUTION | Resume execution of Communication commands. |
| HALT EXECUTION WHEN IDLE | Halt Communication command execution when the 1-Wire bus is idle. |
| HALT EXECUTION WHEN DONE | Halt further Communication command execution when the current command execution completes. |
| FLUSH COMM CMDS | Clear unexecuted Communication commands from the command buffer. |
| FLUSH DATA RCV BUFFER | Clear the data receive buffer (data from a 1-Wire device). |
| FLUSH DATA XMT BUFFER | Clear the data transmit buffer (data to a 1-Wire device). |
| GET COMM CMDS | Retrieve unexecuted Communication commands from the command buffer. |

Control Command Set Descriptions:

RESET DEVICE—This command performs a hardware reset equivalent to the power-on reset. This includes clearing all endpoint buffers and loading the Mode control registers with their default values.

START EXECUTION—This command starts execution of Communication commands. This command is also required to start the execution of Communication commands with an IM (immediate execution control) bit set to logic 0.

RESUME EXECUTION—This command is used to resume execution of a Communication command that was halted with either of the HALT EXECUTION commands.

HALT EXECUTION WHEN IDLE—This command is used to halt the execution of the current Communication command after the 1-Wire bus has returned to the idle state. Further Communication command processing is stopped until a RESUME EXECUTION command is received. This command, or the HALT EXECUTION WHEN DONE command, is also used to terminate a strong pullup of semi-infinite or infinite duration.

HALT EXECUTION WHEN DONE—This command is used to halt the execution of a Communication command after the current command execution is complete. Further Communication command processing is stopped until a RESUME EXECUTION command is received. This command, or the HALT EXECUTION WHEN IDLE command, is also used to terminate a strong pullup of semi-infinite or infinite duration.

FLUSH COMM CMDS—This command is used to clear all unexecuted Communication commands from the command FIFO. The DS2490 must be in a halted state before the FLUSH COMM CMDS command can be processed.

FLUSH DATA RCV BUFFER—This command is used to clear EP3 receive data FIFO (data from 1-Wire device). The DS2490 must be in a halted state before the FLUSH DATA RCV BUFFER command can be processed.

FLUSH DATA XMT BUFFER—This command is used to clear EP2 transmit data FIFO (data to 1-Wire device). The DS2490 must be in a halted state before the FLUSH DATA XMT BUFFER command can be processed.

GET COMM CMDS—This command is used to retrieve unexecuted Communication commands and parameters from the command FIFO. The DS2490 must be in a halted state before the GET COMM CMDS command can be processed. Unexecuted commands are returned over EP0 in the control transfer data phase. Host software is responsible for determining the number of command/parameter bytes to be returned and specifying the value in the wLength field of the control transfer setup packet. Commands/parameters are deleted from the FIFO as they are transmitted to the host; the command pointer used with the FIFO is updated as values are read. Any commands/parameters that are not transferred remain in the FIFO and will be processed when command execution resumes. If the wLength value passed is larger than the number of command/parameter bytes, the DS2490 will terminate the control transfer with a short data packet.

COMMUNICATION COMMANDS

Communication commands are used to communicate with and control an attached 1-Wire device(s). There are 14 Communication commands as summarized in Table 13. Communication commands also contain embedded command bits for conditional control or additional functionality. Appendix 2 details the embedded command bit purpose/formatting as well as the control transfer setup packet coding to transmit the command/parameter data to the DS2490.

Unlike Mode and Control commands, Communication command processing is controlled by host software by setting embedded command bits within a Communication command. For example, a command macro form is supported in which several Communication commands and parameter data are sent/buffered in the DS2490 and then processed as a group. Embedded command bit ICP is used to create macro command sets. Also, unlike Mode and Control commands, several forms of Communication command monitoring exist for host software to track processing progress.

Depending on the settings of embedded communication command bits ICP and NTF, see APPENDIX 2, a Result Register value may be generated to provide command processing feedback. Communication command buffer (FIFO) status and 1-Wire data transmit/received buffers (EP23 FIFOs) status can also be monitored via the State Registers. See the section “DEVICE FEEDBACK” for details on command monitoring.

Communication Command Set Summary Table 13

| COMMAND | FUNCTION |
|--------------------------|--|
| SET DURATION | Change duration of strong pullup. |
| PULSE | Perform strong pullup. |
| 1-WIRE RESET | Generate 1-Wire reset pulse. |
| BIT I/O | 1-Wire single data bit I/O. |
| BYTE I/O | 1-Wire data byte I/O. |
| BLOCK I/O | 1-Wire data block I/O. |
| MATCH ACCESS | Address a device on the 1-Wire bus. |
| READ STRAIGHT | Transmit a data preamble and read back data. |
| DO & RELEASE | Control Crypto iButton CPU activity. |
| SET PATH | Activate a series of couplers that provide access to a target 1-Wire device. |
| WRITE SRAM PAGE | Write data to the scratchpad of a 1-Wire SRAM device. |
| READ CRC PROT PAGE | Read CRC protected pages of a 1-Wire EPROM device or tamper-detect bytes and counter value of Monetary iButtons. |
| READ REDIRECT PAGE W/CRC | Read a single page of data and check the CRC. Follow page redirections if set. |
| SEARCH ACCESS | Access a device using the Search ROM command or identifying devices on the active segments of a 1-Wire Network. |

Communication Command Set Descriptions:

SET DURATION—This command changes the State Register pulse duration value for the strong pullup. The new duration value is loaded into the duration register as detailed in Appendix 2. The duration value can alternatively be changed using a Mode command. The new duration setting remains effective until changed by another SET DURATION or MODE Command. See section “MODE COMMANDS” for duration parameter value setting.

Note:

An infinite duration pulse is terminated by using either of the HALT EXECUTION Control commands. To resume 1-Wire activity after the termination, use the RESUME EXECUTION Control command.

| | |
|----------------------------|---|
| Command parameters: | 1 byte specifying the new duration. |
| EP2 data: | None, this command gets its input data as a command parameter. |
| EP3 data: | None |
| Result Register: | If ICP = 1: no result data is generated. If ICP = 0 and NTF = 1: a result value of 0x00 will be generated. If IPC = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command. |

PULSE—This command is used to generate a strong pullup to 5V in order to provide extra power for an attached iButton device, e.g., temperature sensor, EEPROM, SHA-1, or crypto iButton. The pulse duration is determined by the value in the mode register.

| Command parameters: | None | | | | | | | | |
|----------------------------|---|------|-----------|------|-------------------------------------|------|-------------------------------------|------|---------|
| EP2 data: | None | | | | | | | | |
| EP3 data: | None | | | | | | | | |
| Result Register: | <table><tr><th>Code</th><th>Condition</th></tr><tr><td>0x00</td><td>ICP = 0, NTF = 1: no error detected</td></tr><tr><td>None</td><td>ICP = 0, NTF = 0: no error detected</td></tr><tr><td>None</td><td>ICP = 1</td></tr></table> | Code | Condition | 0x00 | ICP = 0, NTF = 1: no error detected | None | ICP = 0, NTF = 0: no error detected | None | ICP = 1 |
| Code | Condition | | | | | | | | |
| 0x00 | ICP = 0, NTF = 1: no error detected | | | | | | | | |
| None | ICP = 0, NTF = 0: no error detected | | | | | | | | |
| None | ICP = 1 | | | | | | | | |

1-Wire RESET—This command is used to generate a reset pulse on the 1-Wire bus and to optionally change the 1-Wire speed. The new speed will take effect only if the embedded command bit SE is set to 1. When the 1-WIRE RESET is sent after an Overdrive Skip ROM command, SE must be 1 and the new speed must be 0x02. See MODE CONTROL section for additional communication speed information. To switch back to regular speed, set SE = 1 and the new speed parameter to 0x00 (regular) or 0x01 (flexible).

Command parameters: 1 byte specifying new speed after the reset.

EP2 data: None

EP3 data: None

Result Register:

| Code | Condition |
|--------------|---|
| SH, NRS, APP | ICP = 0, NTF = 0 or 1: An abnormal condition or an Alarming Presence Pulse was detected |
| 0x00 | ICP = 0, NTF = 1: no error detected |
| None | ICP = 0, NTF = 0: no error detected |
| None | ICP = 1 |

BIT I/O - This command generates a single time slot on the 1-Wire bus and reads back the response. Data bit d3 (D) of command byte 1 specifies the value to be written to the 1-Wire bus. This time slot may optionally be followed by a strong pullup using embedded command bits SPN and CIB. With CIB = 1, a requested strong pullup will only occur if the read-back revealed a 0. Data is returned to the host only if the embedded command bit ICP = 0. If ICP = 0, the bit read from the 1-Wire device is stored in the EP3 FIFO and is read by the host using an EP3 bulk transaction.

Command parameters: None.

EP2 data: None, this command gets its input data as a command parameter.

EP3 data:

| Length | Description |
|--------------------|---------------------------------------|
| If ICP = 0: 1 byte | Bit read back from the 1-Wire device. |
| If ICP = 1: None | No data is returned. |

Result Register:

- If ICP = 1: no result data is generated.
- If ICP = 0 and NTF = 1: a result value of 0x00 will be generated.
- If ICP = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command.

BIT I/O Read Back Byte Table 14

| bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|------|------|------|------|------|------|------|------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | D |

BYTE I/O—This command accomplishes a direct 1-Wire write and read with optional strong pullup after the last bit of the byte. The optional strong pullup is controlled using embedded command bit SPU. For a write sequence, the data byte to be written is included in the command setup packet as shown in Appendix 2. For a read sequence, the setup packet data byte value is set to 0xFF. Data is returned to the host only if the embedded command bit ICP = 0. If ICP = 0, the byte read from the 1-Wire device is stored in the EP3 FIFO and is read by the host using an EP3 bulk transaction.

| Command parameters: | 1 byte: Data byte to be sent to the 1-Wire bus. To read only, the byte should be 0xFF. | | | | | | |
|----------------------------|--|--------|-------------|--------------------|--|------------------|----------------------|
| EP2 data: | None, this command gets its input data as a command parameter. | | | | | | |
| EP3 data: | <table> <tr> <th>Length</th><th>Description</th></tr> <tr> <td>If ICP = 0: 1 byte</td><td>Byte read back from the 1-Wire device.</td></tr> <tr> <td>If ICP = 1: None</td><td>No data is returned.</td></tr> </table> | Length | Description | If ICP = 0: 1 byte | Byte read back from the 1-Wire device. | If ICP = 1: None | No data is returned. |
| Length | Description | | | | | | |
| If ICP = 0: 1 byte | Byte read back from the 1-Wire device. | | | | | | |
| If ICP = 1: None | No data is returned. | | | | | | |
| Result Register: | <p>If ICP = 1: no result data is generated.</p> <p>If ICP = 0 and NTF = 1: a result value of 0x00 will be generated.</p> <p>If IPC = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command.</p> | | | | | | |

BLOCK I/O—This command accomplishes a direct 1-Wire write or read with optional strong pullup after the last byte of the block. The optional strong pullup is controlled using embedded command bit SPU. Embedded command bit RST enables a 1-Wire reset before the command executes. To accomplish a READ function all input data should be 0xFF, otherwise the data read from the 1-Wire bus will be masked. For a block write sequence the EP2 FIFO must be pre-filled with data before command execution. Additionally, for block sizes greater than the FIFO size, the FIFO content status must be monitored by host SW so that additional data can be sent to the FIFO when necessary. A similar EP3 FIFO content monitoring requirement exists for block read sequences. During a block read the number of bytes loaded into the EP3 FIFO must be monitored so that the data can be read before the FIFO overflows.

| Command parameters: | 2 bytes specifying block size. | | | | |
|----------------------------|--|--------|-------------|--------------------|--|
| EP2 data: | <table> <tr> <th>Length</th><th>Description</th></tr> <tr> <td>Same as block size</td><td>Data block to be written to the 1-Wire device.</td></tr> </table> | Length | Description | Same as block size | Data block to be written to the 1-Wire device. |
| Length | Description | | | | |
| Same as block size | Data block to be written to the 1-Wire device. | | | | |
| EP3 data: | <table> <tr> <th>Length</th><th>Description</th></tr> <tr> <td>Same as block size</td><td>Data block read from the 1-Wire device.</td></tr> </table> | Length | Description | Same as block size | Data block read from the 1-Wire device. |
| Length | Description | | | | |
| Same as block size | Data block read from the 1-Wire device. | | | | |
| Result Register: | <p>If ICP = 1: no result data is generated.</p> <p>If ICP = 0 and NTF = 1: a result value of 0x00 will be generated.</p> <p>If IPC = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command.</p> | | | | |

MATCH ACCESS—This command is used to address a device on the active section of the 1-Wire bus using the Match ROM or Overdrive Match command code. The EP2 FIFO must be pre-filled with the 8 bytes target ROM ID before command execution. Embedded command bit RST enables a 1-Wire reset before the command executes, and embedded command bit SE enables a 1-Wire speed change that takes effect before the command executes.

| | | |
|----------------------------|--|--------------------------------------|
| Command parameters: | 2 bytes: match command byte, new 1-Wire speed byte (optional). | |
| EP2 data: | Length | Description |
| | 8 bytes | ROM ID of the device to be accessed. |
| EP3 data: | None | |
| Result Register: | If ICP = 1: no result data is generated. | |
| | If ICP = 0 and NTF = 1: a result value of 0x00 will be generated. | |
| | If IPC = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command. | |

READ STRAIGHT—This command transmits a user-specified preamble of data to the 1-Wire bus and then reads back as many bytes as specified. The preamble typically consists of a 1-Wire command code followed by TA1 and TA2. It is possible to include a Match ROM command and ROM ID in the preamble to also address a device at the current speed. This command can also be used to copy the scratchpad or disconnect a path. Embedded command bit RST enables a 1-Wire reset before the command executes. The EP2 FIFO must be pre-filled with preamble data before command execution. Additionally, for preamble sizes greater than the FIFO size, the FIFO content status must be monitored by host SW so that additional data can be sent to the FIFO when necessary. A similar EP3 FIFO content monitoring requirement exists for the block read. During a block read, the number of bytes loaded into the EP3 FIFO must be monitored so that the data can be read before the FIFO overflows.

| | | |
|----------------------------|--|---|
| Command parameters: | 3 bytes: 2-byte parameter specifying block read size, byte specifying preamble size. | |
| EP2 data: | Length | Description |
| | Same as preamble size | Preamble data to be written to the 1-Wire device. |
| EP3 data: | Length | Description |
| | Same as block size | Data block read from the 1-Wire device. |
| Result Register: | If ICP = 1: no result data is generated. | |
| | If ICP = 0 and NTF = 1: a result value of 0x00 will be generated. | |
| | If IPC = 0 and NTF = 0: no result data is generated since there are no error codes associated with this command. | |

DO & RELEASE—This command is typically used to control the activity of the CPU of the crypto iButton, which requires a release sequence. This command can also be used to read from or write to the I/O buffer or Status Register of the crypto iButton. A short preamble (3 bytes) consists of a 1-Wire command code followed by the release sequence. A four-byte preamble with embedded command bit R = 1 consists of a 1-Wire command code followed by length byte and the release sequence. A 4-byte preamble with embedded command bit R = 0 consists of a 1-Wire command code followed by a status byte and the release sequence. The long preamble (minimum 5 bytes, embedded command bit R = 0) consists of a 1-Wire command, a length byte, data bytes and the release sequence. In this case, the length byte indicates the total number of bytes between the length byte and release sequence. The least significant byte of the release sequence is transmitted first. Embedded command bit SPU must be 1 when trying to run (start, continue) the CPU of the crypto iButton. In all other cases, SPU should be 0. Additionally, embedded command bit F can be used to clear the communication command buffer and EP2/3 FIFOs if an error occurs during execution of the command. The EP2 FIFO must be pre-filled with preamble data before command execution. Additionally, for preamble sizes greater than the FIFO size,

the FIFO content status must be monitored by host SW so that additional data can be sent to the FIFO when necessary.

A similar EP3 FIFO content monitoring requirement exists for the block read. During a block read the number of bytes loaded into the EP3 FIFO must be monitored so that the data can be read before the FIFO overflows.

| | | |
|---------------------------------------|---|--|
| Command parameters: | 1 byte specifying preamble size. | |
| EP2 data: | Length | Description |
| | Same as preamble size | Preamble data to be written to the 1-Wire device. |
| EP3 data: | Length | Description |
| If R = 1 and preamble size ≥ 4 : | As specified in the second byte of the preamble | Data read from the 1-Wire device. |
| If R = 0: | None | No data is returned (this is a write application). |
| Result Register: | Code | Condition |
| | CRC, CMP | ICP = 0, NTF = 0 or 1: Bad CRC or the release sequence was not accepted. |
| | 0x00 | ICP = 0, NTF = 1: no error detected |
| | None | ICP = 0, NTF = 0: no error detected |
| | None | ICP = 1 |

SET PATH—This command is used to activate a series of couplers that finally provide the access to the target 1-Wire device. Embedded command bit RST enables a 1-Wire reset before the command executes. Additionally, embedded command bit F can be used to clear the communication command buffer and EP2/3 FIFOs if an error occurs during execution of the command. The EP2 FIFO must be pre-filled with coupler data before command execution. Additionally, for coupler data sizes greater than the EP2 FIFO size, the FIFO content status must be monitored by host SW so that additional data can be sent to the FIFO when necessary. Upon completion of command execution, the EP3 FIFO is loaded with a 1-byte value that indicates the number of couplers activated. An NRS error code is an indication that there was no presence pulse on the branch that was to be connected.

| | | |
|----------------------------|---|---|
| Command parameters: | 1 byte specifying number of couplers that need to be activated to set-up the path to the target device. | |
| EP2 data: | Length | Description |
| | 9 bytes per level | 8 bytes ROM ID of the coupler to be activated followed by the Smart-On command code for either the main or auxiliary 1-Wire coupler output. |
| EP3 data: | Length | Description |
| | 1 byte | Number of couplers successfully activated; should be the same as the number of levels. |
| Result Register: | Code | Condition |
| | CMP, SH, NRS | ICP = 0, NTF = 0 or 1: An error condition is detected. |
| | 0x00 | ICP = 0, NTF = 1: no error detected |
| | None | ICP = 0, NTF = 0: no error detected |
| | None | ICP = 1 |

WRITE SRAM PAGE—This command is used to write data to the scratchpad of an SRAM device and optionally check the CRC. To copy the data to its final memory location, it is required to send the READ STRAIGHT command with a “copy scratchpad” preamble and a 0 block size. Embedded command bit DT = 1 activates the CRC16 generator. This command is also applicable to the Write IPR function of the Crypto iButton if embedded command bit CIB = 1. To write a partial page the page size parameter and the target address need to be set accordingly. The EP2 FIFO must be pre-filled with preamble and SRAM data before command execution. Additionally, for data page sizes greater than the EP2 FIFO size, the FIFO content status must be monitored by host SW so that additional data can be sent to the FIFO when necessary.

| | | |
|----------------------------|--|---|
| Command parameters: | 1 byte specifying page size. This specifies the number of bytes to be sent to the 1-Wire device following a 2- or 3-byte preamble. This number is typically identical to the page size; a value of 0x00 indicates a 256-byte page. | |
| EP2 data: | Length | Description |
| | If CIB = 0: 3-bytes + page size | Preamble and data bytes to be sent to the 1-Wire SRAM device. 3-byte preamble: 1-Wire command code, TA1, TA2 (typical). |
| | If CIB = 1: 2-bytes + page size | Preamble and data bytes to be sent to the 1-Wire SRAM device. 2-byte preamble: 1-Wire command code, length information. |
| EP3 data: | None | |
| Result Register: | Code | Condition |
| | CRC | ICP = 0, NTF = 0 or 1: and an error condition (only if DT = 1). |
| | 0x00 | ICP = 0, NTF = 1: no error detected |
| | None | ICP = 0, NTF = 0: no error detected |
| | None | ICP = 1 |

READ CRC PROT PAGE—This command is used to read one or multiple CRC-protected pages of EPROM devices and to read tamper-detect bytes and counter value of Monetary iButtons. If embedded command bit DT = 1, the CRC16 generator is selected; if DT = 0, CRC8 is used. Embedded command bit CIB is used to specify either a 2- or 3-byte read preamble. Additionally, embedded command bit F can be used to clear the Communication command buffer and EP2/3 FIFOs if an error occurs during execution of the command. The command can also be used to read a single partial page up to the end of that page and reading the PIOs of a DS2406. This command is also applicable to the Read IPR function of the crypto iButton if CIB = 1. The EP2 FIFO must be pre-filled with preamble data before command execution. To prevent overflow, the EP3 FIFO must be monitored (and read if necessary) during command execution if reading more data than the size of the EP3 FIFO.

Command parameters: 2 bytes: 1 byte page size, 1 byte number of pages. Note that the Page Size byte parameter value must be of the form $\log_2(\text{actual page size})$. For example, if the actual page size is 32_{10} , then the Page Size parameter would need to be 0x05. Typical page sizes are 32 decimal for data memory and 8 for status memory. A page size of 0x00 indicates a 256-byte page.

EP2 data: **Length**

| | | |
|-------------|---------|---|
| If CIB = 0: | 3-bytes | 3-byte preamble: 1-Wire command code, TA1, TA2 (typical). |
| If CIB = 1: | 2-bytes | 2-byte preamble: 1-Wire command code, length information |

EP3 data: **Length** **Description**

| | |
|-------------------------------|-----------------------------------|
| (page size x number of pages) | Data read from the 1-Wire device. |
|-------------------------------|-----------------------------------|

Result Register: **Code** **Condition**

| | |
|------|--|
| CRC | ICP = 0, NTF = 0 or 1: and an error condition. |
| 0x00 | ICP = 0, NTF = 1: no error detected |
| None | ICP = 0, NTF = 0: no error detected |
| None | ICP = 1 |

READ REDIRECT PAGE W/CRC—This command is used to read a single page of data and check the CRC. If the page is not redirected, the host will receive the page number and all its data. If the page is redirected and embedded command bit CH = 1 the device will follow the trail of redirections until a page is found that is not redirected. When found, the host will receive the page number and all data of that particular page. If CH = 0 and the first read attempt reveals a redirection the host will only receive the number of the page the addressed page is redirected to. Reading starts at a page boundary. Additionally, embedded command bit F can be used to clear the communication command buffer and EP2/3 FIFOs if an error occurs during execution of the command. An NRS error code is an indication of an infinite loop. The EP2 FIFO must be pre-filled with ROM ID and command code data before command execution. To prevent overflow, the EP3 FIFO must be monitored (and read if necessary) during command execution if reading more data than the size of the EP3 FIFO.

| | | |
|--------------------------------------|---|--|
| Command parameters: | 2 bytes: 1 byte page size, 1 byte page address. Page size byte indicates the number of bytes that are considered a page. Typical page sizes are 32 decimal for data memory, 8 for status memory, and 40 decimal for Monetary iButtons. A page size of 0x00 indicates a 256-byte page. The page address specifies the page number to be used for the first read attempt. | |
| EP2 data: | Length 9 bytes | 8 bytes ROM ID of the 1-Wire device to be accessed followed by the 1-Wire command code for “Extended Read Memory”. |
| EP3 data: | Length | Description |
| If CH = 1 or page is not redirected: | 1 byte + page size | Page number and contents of the page. |
| If CH = 0 and page is redirected: | 1 byte | Number of the page the addressed page is redirected to. |
| Result Register: | Code | Condition |
| | RDP, CRC, NRS | ICP = 0, NTF = 0 or 1: and an error condition. |
| | 0x00 | ICP = 0, NTF = 1: no error detected |
| | None | ICP = 0, NTF = 0: no error detected |
| | None | ICP = 1 |

SEARCH ACCESS—The SEARCH ACCESS command is used to either access a device using the Search ROM command or to identify the devices on the active segments of a 1-Wire Network. Embedded command bit SM is used to specify the type of search. Embedded command bit RTS is used to control discrepancy information reporting. Additionally, embedded command bit F can be used to clear the Communication command buffer and EP2/3 FIFOs if an error occurs during execution of the command. An NRS error code indicates that there was no response on at least one of the 64 bits of the ROM search. The EP2 FIFO must be pre-filled with ROM ID data before command execution. To prevent overflow, the EP3 FIFO must be monitored (and read if necessary) during command execution if returning more data than the size of the EP3 FIFO.

| | | | |
|----------------------------|---|---|--|
| Command parameters: | 2 bytes: 1 byte 1-Wire command (Search ROM or Conditional Search ROM), 1 byte for number of devices. The number of devices byte specifies the maximum number of devices to be discovered in a single command call. A value of 0x00 indicates that all devices on the 1-Wire Network are to be discovered. | | |
| EP2 data: | Length | | |
| | 8 bytes | ROM ID of the 1-Wire device to be accessed (SM = 0) or ROM ID with which to start the search (SM = 1). | |
| EP3 data: | Length | Description | |
| | If SM = 0: None | No data is returned. | |
| | If SM = 1: Varies, blocks of 8 bytes. | ROM IDs of the devices discovered followed by 8 bytes discrepancy information (if RTS = 1 and the numbers of devices on the 1-Wire Network is higher than specified in the second parameter). | |
| Result Register: | Code | Condition | |
| | EOS, NRS | ICP = 0, NTF = 0 or 1: and an error condition. | |
| | 0x00 | ICP = 0, NTF = 1: no error detected | |
| | None | ICP = 0, NTF = 0: no error detected | |
| | None | ICP = 1 | |

Not Recommended for New Designs

DEVICE FEEDBACK

DS2490 state and result feedback are provided to the host over the EP1 interrupt pipe. The host interrupt polling period for EP1 transfers is controlled by the alternate setting of the 1-Wire USB Interface as described in the section: 1-Wire Interface Summary Description. As shown in Table 15, up to 32 bytes of feedback data are returned. As a minimum, 16 bytes of STATE REGISTER data will be delivered at each polling interval (or USB IN transaction to EP1); the position and definition of these data values are listed in Table 16. In addition, as shown in Table 17, between 0 and 16 extra bytes of RESULT REGISTER data will be delivered with feedback data. This consists of communication command processing result data as controlled by the embedded communication command bits ICP and NTF and notification bytes for 1-Wire device arrival detection. If there are less than 32 bytes of result and state register data to return, the DS2490 will terminate the IN transaction with a short or zero data packet.

Data is loaded into the RESULT REGISTER buffer based on embedded communication command bits NTF and ICP or upon detection of a 1-Wire device attachment. There is no defined offset position for data values in this buffer. Knowledge of communication command ordering sent to the DS2490 must be maintained on the host to properly interpret the returned data values. In addition, a parse of the data is required to determine whether or not a 1-Wire detect value is included; the value of the 1-Wire detect byte is unique. Table 18 defines the 1-Wire detect byte and data values that result from Communication command processing.

Interrupt Transfer Data Summary Table 15

| OFFSET | DATA |
|---------------------------|------------------|
| 0x00 to 0x0F | STATE REGISTERS |
| 0x10 to 0x1F ¹ | RESULT REGISTERS |

Notes:

1. The number of RESULT REGISTER values will vary depending on the setting of embedded communication command bits NTF and ICP, the number of communication commands processed since the previous EP1 polling period or whether a 1-Wire device was detected.

DS2490 STATE REGISTERS Table 16

| Description | | Offset | Data | | | | | | | | | |
|----------------------------|--|--------|---------------------------------|------|---|------|------|------|------|------|--|--|
| Enable Flags | | 0x00 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | | |
| | | | -- | -- | -- | -- | -- | SPCE | 0 | SPUE | | |
| | | | SPUE | | If set to 1, the strong pullup to 5V is enabled, if set to 0, it is disabled. | | | | | | | |
| | | | SPCE | | If set to 1, a dynamic 1-Wire bus speed change through a Communication command is enabled, if set to 0, it is disabled. | | | | | | | |
| | | | bit 3..7 | | These bits are reserved. | | | | | | | |
| 1-Wire Speed | | 0x01 | current 1-Wire bus speed code | | | | | | | | | |
| Strong Pullup Duration | | 0x02 | current pullup duration | | | | | | | | | |
| (Reserved) | | 0x03 | | | | | | | | | | |
| Pulldown Slew Rate Control | | 0x04 | current pulldown slew rate code | | | | | | | | | |

| Description | Offset | Data | | | | | | | |
|--|--------|--|---|------|------|------|------|------|------|
| Write-1 Low Time | 0x05 | current Write-1 low time code | | | | | | | |
| Data Sample Offset / Write-0 Recovery Time | 0x06 | current data sample offset/ Write-0 recovery time code | | | | | | | |
| Reserved (Test Register) | 0x07 | | | | | | | | |
| Device Status Flags | 0x08 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | EP0F | -- | IDLE | HALT | PMOD | -- | -- | SPUA |
| | | Bit Definitions: | | | | | | | |
| | | SPUA | if set to 1, the strong pullup to 5V is currently active, if set to 0, it is inactive. | | | | | | |
| | | bit 1: | Reserved | | | | | | |
| | | bit 2: | Reserved | | | | | | |
| | | PMOD | if set to 1, the DS2490 is powered from USB and external sources, if set to 0, all DS2490 power is provided from USB. | | | | | | |
| | | HALT | if set to 1, the DS2490 is currently halted, if set to 0, the device is not halted. | | | | | | |
| | | IDLE | if set to 1, the DS2490 is currently idle, if set to 0, the device is not idle. | | | | | | |
| | | bit 6: | Reserved | | | | | | |
| | | EPOF: | Endpoint 0 FIFO status, see Note 1. | | | | | | |
| Communication Command, Byte 1 | 0x09 | Communication command currently being processed. If the device is idle, a register value of 0x00 is sent. | | | | | | | |
| Communication Command, Byte 2 | 0x0A | Communication command currently being processed. If the device is idle, a register value of 0x00 is sent. | | | | | | | |
| Communication Command Buffer Status | 0x0B | Number of data bytes currently contained in the 16-byte FIFO used to hold communication commands. | | | | | | | |
| 1-Wire Data Out Buffer Status | 0x0C | Number of data bytes currently contained in the 128-byte FIFO used to write data to the 1-Wire bus. | | | | | | | |
| 1-Wire Data In Buffer Status | 0x0D | Number of data bytes currently contained in the 128-byte command FIFO used to read data from the 1-Wire bus. | | | | | | | |
| Reserved (Test Register) | 0x0E | Reserved | | | | | | | |
| Reserved (Test Register) | 0x0F | Reserved | | | | | | | |

Notes:

1. If EP0F is set to 1, the Endpoint 0 FIFO was full when a new control transfer setup packet was received. This is an error condition in that the setup packet received is discarded due to the full condition. To recover from this state the USB host must send a CTL_RESET_DEVICE command; the device will also recover with a power on reset cycle. Note that the DS2490 will accept and process a CTL_RESET_DEVICE command if the EP0F = 1 state occurs. If EP0F = 0, no FIFO error condition exists.

DS2490 RESULT REGISTERS Table 17

| Offset ^{1,2} | Data |
|-----------------------|-------|
| 0x10 | value |
| 0x11 | value |
| ... | ... |
| 0x1F | value |

Notes:

1. Since Result Registers follow State Registers for interrupt transfers, the register offset starting value shown in Table 17 for Result Registers follows the last value of Table 16 State Registers.
2. The number of RESULT REGISTER values will vary depending on the setting of embedded communication command bits NTF and ICP, the number of communication commands processed since the previous EP1 polling period, or whether a 1-Wire device was detected.

DS2490 RESULT REGISTER VALUE DEFINITION Table 18

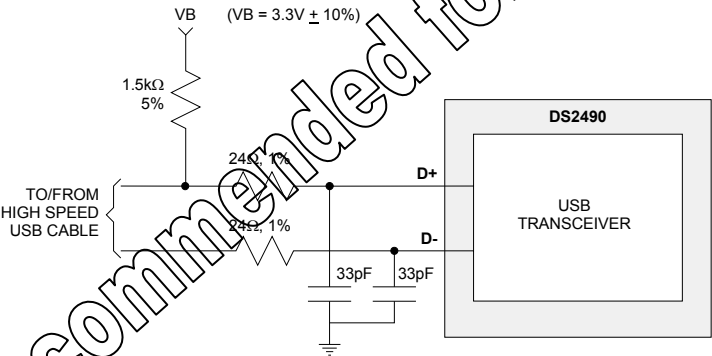
| Data Value Description | Data | | | | | | | |
|------------------------------------|-----------------|--|------|------|------|------|------|------|
| 1-Wire Device Detect Byte | 0x45 | | | | | | | |
| Communication Command Error Result | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | EOS | RDP | CRC | CMP | 0 | APP | SH | NRS |
| | Bit Definitions | | | | | | | |
| | EOS | A value of 1 indicates that a SEARCH ACCESS with SM = 1 ended sooner than expected reporting less ROM IDs than specified in the “number of devices” parameter. | | | | | | |
| | RDP | A value of 1 indicates that a READ REDIRECT PAGE WITH/CRC encountered a page that is redirected. | | | | | | |
| | CRC | A value of 1 indicates that a CRC error occurred when executing one of the following commands: WRITE SRAM PAGE, READ CRC PROT PAGE, or READ REDIRECT PAGE W/CRC. | | | | | | |
| | CMP | A value of 1 indicates an error with one of the following: Error when reading the confirmation byte with a SET PATH command. There was a difference between the byte written and then read back with a BYTE I/O command | | | | | | |
| | APP | A value of 1 indicates that a 1-WIRE RESET revealed an Alarming Presence Pulse. | | | | | | |
| | SH | A value of 1 indicates that a 1-WIRE RESET revealed a short to the 1-Wire bus or the SET PATH command could not successfully connect a branch due to a short. | | | | | | |

| Data Value Description | Data | |
|------------------------|--|--|
| | NRS | A value of 1 indicates an error with one of the following: 1-WIRE RESET did not reveal a Presence Pulse. SET PATH command did not get a Presence Pulse from the branch that was to be connected. No response from one or more ROM ID bits during a SEARCH ACCESS command. |
| | A value of 0 in any of these fields indicates that the error condition was not detected. | |

USB TRANSCEIVER

Integrated within the DS2490 is a USB-compliant bus transceiver. To comply with USB cable termination requirements, resistors external to the DS2490 must be installed as shown in Figure 10. The 24Ω, 1% tolerance resistors are necessary to bring the total steady state resistance of each driver to the 28Ω to 43Ω range required by the USB core specification. The 1.5kΩ pullup resistor is required to identify the DS2490 as a high speed USB device to the up-stream HUB. The 33pF capacitors are optional but are useful for edge rate control and for bypassing high frequency energy to ground for EMI reduction purposes.

USB Transceiver Termination Figure 10



SUSPEND OUTPUT

Suspend output signal ($\overline{\text{SUSO}}$) operation is a function of both the USB device state of the DS2490 (configured or unconfigured) and the USB suspend state. A truth table for signal operation is listed in Table 19. This signal could be used stand-alone or combined with other external signals to power down circuitry external to the DS2490, such as power sources. The $\overline{\text{SUSO}}$ signal is an open drain output and requires an external pullup.

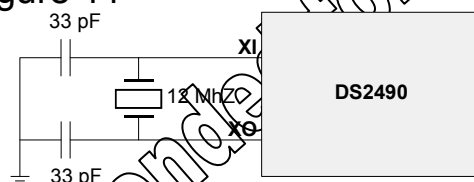
Suspend Output Operation Table 19

| Device State | Suspend State | $\overline{\text{SUSO}}$ |
|--------------|---------------|--------------------------|
| unconfigured | don't care | HIGH |
| configured | not suspended | HIGH |
| configured | suspended | LOW |

OSCILLATOR

The DS2490 requires a single 12.0MHz crystal or crystal oscillator clock source to operate. Crystals or CMOS crystal oscillators may be used to provide clock sources. For crystals, use parallel resonant, fundamental mode with a CL specification between 10pF and 20pF at the required frequency. Parallel loading capacitors with a value of approximately twice CL are also required. The connection scheme shown in Figure 11 should be used. For external CMOS crystal oscillators, make a connection directly to the DS2490 XI input. The XO output must be left floating for this arrangement.

Oscillator Connection Figure 11



ABSOLUTE MAXIMUM RATINGS*

| | |
|---------------------------------------|------------------------------|
| Voltage on Any Pin Relative to Ground | -0.5V to +6.0V |
| Operating Temperature Range | 0 °C to +70 °C |
| Storage Temperature Range | -55 °C to +125 °C |
| Soldering Temperature | See J-STD-020A 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 reliability.

RECOMMENDED DC OPERATING CONDITIONS

| PARAMETER | SYMBOL | MIN | MAX | UNITS | NOTES |
|------------------------|--------|-----|-----|-------|---------|
| USB I/F Supply Voltage | VB | 3.0 | 3.6 | V | 1, 3 |
| Digital Supply Voltage | VD | 4.4 | 5.5 | V | 1, 3 |
| Second Supply Voltage | VD2 | 4.4 | 5.5 | V | 1, 2, 3 |
| Operating Temperature | TA | 0 | 70 | °C | |

NOTES:

- Voltage referenced to ground.
- Must be tied to VD.
- During device power-up, the supply application sequence order must be VD/VD2, VB.

DC ELECTRICAL CHARACTERISTICS

TA: 0 °C to +70 °C

VD: 4.4V to 5.5V, VB: 3.0V to 3.6V

| PARAMETER | SYMBOL | MIN | MAX | UNITS | NOTES |
|--|-----------------------|-----------|-----------|-------|-------|
| VD Operating Current | I _D | | 20 | mA | 2 |
| VB Operating Current | I _B | | 7 | mA | 2 |
| VD Suspend Current | I _{DS} | | 1200 | μA | 3 |
| VB Suspend Current | I _{BS} | | 10 | μA | 3 |
| VD2 Operating Current | I _{D2} | | 1.0 | μA | |
| USB D+/- Input High | V _{IHU} | 2.35 | VB + 0.3 | V | 4 |
| USB D+/- Input Low | V _{ILU} | -0.3 | 0.7 | V | 4 |
| 1-Wire Input High | V _{IH1} | 3.55 | VDD + 0.3 | V | 4 |
| 1-Wire Input Low | V _{IL1} | -0.3 | 1.35 | V | 4 |
| SUSO Output Low @ 1mA | V _{OL(SUSO)} | | 100 | mV | 4 |
| Active Pullup Timer Threshold | V _{IAPTO} | VD - 1.75 | VDD - 0.3 | V | 4 |
| Active Pullup on Threshold | V _{IAPO} | 0.25 | 1.1 | V | 4 |
| 1-Wire Weak Pullup Current | I _{WEAKPU} | 0.9 | 6.0 | mA | |
| 1-Wire Active Pullup current | I _{ACTPU} | 5.9 | 30.4 | mA | |
| Strong Pullup Voltage Drop @ 50mA load on 1-Wire | ΔV _{STRPU} | 170 | 1540 | mV | 1 |

NOTES:

1. Voltage difference between VD and 1-WIRE.
2. Applies for both unconfigured and configured USB states.
3. Applies for the USB suspend state.
4. Voltage referenced to ground

AC CHARACTERISTICS CONDITIONS:

TA: 0 °C to +70 °C
 VD: 4.4V to 5.5V, VB: 3.0V to 3.6V

| PARAMETER | SPECIFICATION |
|------------|--|
| USB I/F | As defined in Chapter 7 of the USB Specification |
| 1-Wire I/F | See the section “1-WIRE INTERFACE CONTROLLER” of this document |

CAPACITANCE CONDITIONS:

TA: +25 °C

| PARAMETER | SYMBOL | MIN | MAX | UNITS | NOTES |
|--------------------------|--------|-----|-----|-------|-------|
| Input Capacitance | CIN | | 15 | pF | |
| Output Capacitance | COUT | | 15 | pF | |
| Input/Output Capacitance | CIO | | 15 | pF | |

HW/SW APPLICATION RECOMMENDATIONS

A schematic example of a USB to 1-Wire Adapter peripheral is shown in Figure 12.

APPENDIX 1: DS2490 CONTROL COMMANDS, SETUP PACKET ENCODING

| CONTROL COMMANDS | |
|---------------------------------|-------------------|
| RESET DEVICE | |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_RESET_DEVICE |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| START EXECUTION | |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_START_EXE |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| RESUME EXECUTION | |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_RESUME_EXE |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| HALT EXECUTION WHEN IDLE | |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_HALT_EXE_IDLE |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |

| CONTROL COMMANDS | |
|---------------------------------|---|
| HALT EXECUTION WHEN DONE | |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_HALT_EXE_DONE |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| FLUSH COMM CMDS | The DS2490 must be in a halted state before the FLUSH COMM CMDS command can be processed. |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_FLUSH_COMM_CMDS |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| FLUSH DATA RCV BUFFER | The DS2490 must be in a halted state before the FLUSH DATA RCV BUFFER command can be processed. |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_FLUSH_RCV_BUFFER |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |
| FLUSH DATA XMT BUFFER | The DS2490 must be in a halted state before the FLUSH DATA XMT BUFFER command can be processed. |
| Setup Packet Encoding: | |
| bmRequestType | 0x40 |
| bRequest | CONTROL_CMD |
| wValue | CTL_FLUSH_XMT_BUFFER |
| wIndex | 0x0000 |
| wLength | 0x0000 |
| | |

| CONTROL COMMANDS | |
|------------------------|---|
| GET COMM CMDS | The DS2490 must be in a halted state before the GET COMM CMDS command can be processed. |
| Setup Packet Encoding: | |
| bmRequestType | 0xC0 |
| bRequest | CONTROL_CMD |
| wValue | CTL_GET_COMM_CMDS |
| wIndex | 0x0000 |
| wLength | variable |
| | The value of the wLength field depends on the number of communication commands and parameters loaded in the FIFO. |

APPENDIX 2: DS2490 COMMUNICATION COMMANDS, EMBEDDED COMMAND BITS, SETUP PACKET ENCODING.

Communication commands contain embedded command parameter bits in the wValue field. The definitions of bit parameters are described below.

| BIT NAME | DESCRIPTION |
|----------|---|
| CH | CH = 1 follows the chain if the page is redirected. CH = 0 stops reading if the page is redirected. |
| CIB | CIB = 1 prevents a strong pullup to 5V if SPU = 1 and the bit read back from the 1-Wire bus is 1. CIB = 0 generally enables the strong pullup to 5V. |
| PS | PS = 1 reduces the preamble size to 2 bytes (rather than 3). PS = 0 sets preamble size to 3 bytes. |
| D | Data bit value to be written to the 1-Wire bus. |
| DT | DT = 1 activates/selects the CRC16 generator DT = 0 specifies no CRC. |
| F | F = 1 clears the buffers in case an error occurred during the execution of the previous command; requires that ICP = 0 in the previous command. F = 0 prevents the buffers from being cleared. |
| ICP | ICP = 1 indicates that the command is not the last one of a macro; as a consequence command processing result feedback messages are suppressed. ICP = 0 indicates that the command is the last one of a macro or single command operation; enables command processing result feedback signaling. |
| IM | IM = 1 enables immediate execution of the command. Assumes that all 1-Wire device data required by the command has been received at EP2. IM = 0 prevents immediate execution of the command; execution must be started through a control function command. |
| NTF | NTF = 1 always generate communication command processing result feedback if ICP = 0 NTF = 0 generate communication command processing result feedback only if an error occurs and ICP = 0. If ICP = 1 command result feedback is suppressed for either case, see the ICP bit above. |

| BIT NAME | DESCRIPTION |
|-----------------|---|
| PST | PST = 1 continuously generate 1-Wire Reset sequences until a presence pulse is discovered. PST = 0 generate only one 1-Wire Reset sequence. |
| R | R = 1 performs a read function. R = 0 performs a write function. |
| RST | RST = 1 inserts a 1-Wire Reset before executing the command. RST = 0 no 1-Wire Reset inserted. |
| RTS | RTS = 1 returns the discrepancy information to the host if SM = 1 and there are more devices than could be discovered in the current pass. RTS = 0 does not return discrepancy information. |
| SE | SE = 1 enable the speed change on the 1-Wire bus. SE = 0 disable the speed change on the 1-Wire bus. |
| SM | SM = 1 searches for and reports ROM Ids without really accessing a particular device. SM = 0 makes a “Strong Access” to a particular device. |
| SPU | SPU = 1 inserts a strong pullup to 5V after a Bit or Byte or Block I/O or Do & Release command. SPU = 0 no strong pullup. |
| Z | Z = 1 checks if the 0-bits in the byte to be written are 0-bits in the byte read back from the device. Z = 0 checks if the byte to be written is identical to the one read back from the device. |

COMMUNICATION COMMANDS

| SET DURATION | | | | | | | | | | | |
|------------------------|--|--|---|---|---|---|--------|-----|-----|----|--|
| Setup Packet Encoding: | | | | | | | | | | | |
| bmRequestType | | 0x40 | | | | | | | | | |
| bRequest | | COMM_CMD | | | | | | | | | |
| wValue | | byte 2 | | | | | byte 1 | | | | |
| | | byte 2: | 0 | 0 | 0 | 0 | 0 | NTF | ICP | 0 | |
| | | byte 1: | 0 | 0 | 0 | 1 | 0 | 0 | 1 | IM | |
| wIndex | | byte 2 | | | | | byte 1 | | | | |
| | | byte 2 – 0x00 (unused) | | | | | | | | | |
| | | byte 1 – new duration: Unsigned 8-bit binary number specifying the new duration. See MODE CONTROL section for byte bit weighting and additional format information. | | | | | | | | | |
| wLength | | 0x0000 | | | | | | | | | |

| COMMUNICATION COMMANDS | | | | | | | | | |
|------------------------|---|---|-----|---|-----|--------|-----|-----|----|
| PULSE | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | |
| | byte 2: | 0 | 0 | 0 | 0 | F | NTF | ICP | 0 |
| | byte 1: | 0 | 0 | 1 | 1 | 0 | 0 | 0 | IM |
| wIndex | 0x0000 | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |
| | | | | | | | | | |
| 1-WIRE RESET | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | |
| | byte 2: | 0 | PST | 0 | 0 | F | NTF | ICP | 0 |
| | byte 1: | 0 | 1 | 0 | 0 | SE | 0 | 1 | IM |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 – 0x00 (unused) | | | | | | | | |
| | byte 1 – new speed: 0x02 with SE = 1 after Overdrive Skip ROM 0x01 or 0x00 with SE = 1 in all other cases don't care with SE = 0 | | | | | | | | |
| | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |
| | | | | | | | | | |
| BIT I/O | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | |
| | byte 2: | 0 | CIB | 0 | SPU | 0 | NTF | ICP | 0 |
| | byte 1: | 0 | 0 | 1 | 0 | D | 0 | 0 | IM |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 – 0x00 (unused) | | | | | | | | |
| | byte 1 – 0x00 (unused) | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |
| | | | | | | | | | |

| COMMUNICATION COMMANDS | | | | | | | | | | |
|------------------------|---|---|---|---|-----|--------|-----|-----|-----|--|
| BYTE I/O | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | 0 | 0 | SPU | 0 | NTF | ICP | 0 | |
| | byte 1: | 0 | 1 | 0 | 1 | 0 | 0 | 1 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – 0x00 (unused) | | | | | | | | | |
| | byte 1 – data byte to be sent to the 1-Wire bus. To read only, the byte should be 0xFF. | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |
| | | | | | | | | | | |
| BLOCK I/O | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | 0 | 0 | SPU | 0 | NTF | ICP | RST | |
| | byte 1: | 0 | 1 | 1 | 1 | 0 | 1 | 0 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – block size (HI), most significant byte | | | | | | | | | |
| | byte 1 – block size (LOW), least significant byte | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |
| | | | | | | | | | | |

| COMMUNICATION COMMANDS | | | | | | | | | | |
|------------------------|---|---|---|---|---|--------|-----|-----|-----|--|
| MATCH ACCESS | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | 0 | 0 | 0 | 0 | NTF | ICP | RST | |
| | byte 1: | 0 | 1 | 1 | 0 | SE | 1 | 0 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – new speed: 0x02 with SE = 1 after Overdrive Skip ROM 0x01 or 0x00 with SE = 1 in all other cases don't care with SE = 0 | | | | | | | | | |
| | byte 1 – 1-Wire command: Command to be used for addressing a 1-Wire device. Valid codes: 0x55 (Match ROM) 0x69 (Overdrive Match ROM) | | | | | | | | | |
| | | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |
| | | | | | | | | | | |
| READ STRAIGHT | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – Preamble size. Number of bytes to be written. | | | | | | | | | |
| | byte 1: | 1 | 0 | 0 | 0 | NTF | ICP | RST | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – Block size HI. Number of bytes to be read, most significant byte. | | | | | | | | | |
| | byte 1 - Block size LOW. Number of bytes to be read, least significant byte. | | | | | | | | | |
| | | | | | | | | | | |
| wLength | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – 0x00 (not used) | | | | | | | | | |
| | byte 1 – Preamble size. Number of bytes to be written. | | | | | | | | | |

| COMMUNICATION COMMANDS | | | | | | | | | | |
|------------------------|--|---|----|----|-----|--------|-----|-----|-----|--|
| DO & RELEASE | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | 1 | 1 | SPU | F | NTF | ICP | 0 | |
| | byte 1: | 1 | 0 | 0 | 1 | R | 0 | 1 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – 0X00 (unused) | | | | | | | | | |
| | byte 1 – size in bytes of preamble to be written. | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |
| | | | | | | | | | | |
| SET PATH | | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | 0 | 0 | 0 | F | NTF | ICP | RST | |
| | byte 1: | 1 | 0 | 0 | 0 | 0 | 0 | 1 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – 0X00 (unused) | | | | | | | | | |
| | byte 1 – size in bytes of number of couplers that need to be activated to set up the path to the target device. | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |
| | | | | | | | | | | |
| WRITE SRAM PAGE | | DT = 1 activates the CRC16 generator. This command is also applicable to the Write IPR and Write I/O Buffer function of the crypto iButton if CIB = 1. To write a partial page the page size parameter and the target address need to be set accordingly. | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2: | 0 | PS | DT | 0 | F | NTF | ICP | 0 | |
| | byte 1: | 1 | 0 | 1 | 1 | 0 | 0 | 1 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 – 0X00 (unused) | | | | | | | | | |
| | byte 1 – number of bytes to be sent to the device following a 3 byte preamble. This number is typically identical to the page size. A value of 0x00 specifies a 256 byte page. | | | | | | | | | |
| wLength | 0x0000 | | | | | | | | | |

COMMUNICATION COMMANDS**READ CRC PROT PAGE**

DT = 1 selects the CRC16 generator, otherwise an 8-bit CRC is assumed. This command is also applicable to the Read IPR and Read I/O Buffer function of the crypto iButton if CIB = 1.

Setup Packet Encoding:

| | | | | | | | | | |
|---------------|--|---|----|----|---|--------|-----|-----|----|
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | |
| | byte 2: | 0 | PS | DT | 0 | F | NTF | ICP | 0 |
| | byte 1: | 1 | 1 | 0 | 1 | 0 | 1 | 0 | IM |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 –number of pages to be read. | | | | | | | | |
| | byte 1 – page size: log(base 2) of the number of bytes that are considered a page. Typical is 32-decimal for data memory and 8 for status memory. Page size 0 indicates a 256-byte page. As an example, if the actual page size is 32 ₁₀ , then the Page Size parameter would need to be 0x05 | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |

READ REDIRECT PAGE W/CRC

Setup Packet Encoding:

| | | | | | | | | | | |
|---------------|--|---|---|---|---|--------|-----|-----|----|--|
| bmRequestType | 0x40 | | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | | |
| | byte 2 | 0 | 0 | 1 | 0 | F | NTF | ICP | 1 | |
| | byte 1 | 1 | 1 | 1 | 0 | CH | 1 | 0 | IM | |
| wIndex | byte 2 | | | | | byte 1 | | | | |
| | byte 2 - page number to be used for the first read attempt. | | | | | | | | | |
| | byte 1 – page size: number of bytes that are considered a page. Typical is 32 decimal for data memory. Page size 0 indicates a 256-byte page. | | | | | | | | | |
| wlength | 0x0000 | | | | | | | | | |

| COMMUNICATION COMMANDS | | | | | | | | | |
|------------------------|--|---|-----|---|---|--------|-----|-----|-----|
| SEARCH ACCESS | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | COMM_CMD | | | | | | | | |
| wValue | byte 2 | | | | | byte 1 | | | |
| | byte 2: | 0 | RTS | 0 | 0 | F | NTF | ICP | RST |
| | byte 1: | 1 | 1 | 1 | 1 | SM | 1 | 0 | IM |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 – number of devices. Maximum number of devices to be discovered in a single command call. A value of 0x00 indicates that all devices on the 1-Wire Network are to be discovered | | | | | | | | |
| | byte 1 – 1-Wire command: Command to be used for addressing a 1-Wire device. Valid codes: 0xF0 (Search ROM) 0xEC (Conditional Search ROM) | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |

APPENDIX 3: DS2490 MODE COMMANDS, SETUP PACKET ENCODING

| MODE COMMANDS | | | | | | | | | |
|------------------------|---|---|---|---|---|--------|---|------|---|
| ENABLE PULSE | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | MODE_CMD | | | | | | | | |
| wValue | MOD_PULSE_EN | | | | | | | | |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 – 0x00 | | | | | | | | |
| | byte 1: | 0 | 0 | 0 | 0 | 0 | 0 | SPUE | 0 |
| | Strong pullup is enabled with SPUE = 1, disabled with SPUE = 0. | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |
| | | | | | | | | | |
| ENABLE SPEED CHANGE | | | | | | | | | |
| Setup Packet Encoding: | | | | | | | | | |
| bmRequestType | 0x40 | | | | | | | | |
| bRequest | MODE_CMD | | | | | | | | |
| wValue | MOD_SPEED_CHANGE_EN | | | | | | | | |
| wIndex | byte 2 | | | | | byte 1 | | | |
| | byte 2 – 0x00 | | | | | | | | |
| | byte 1 – Boolean value. Speed changes are enabled with TRUE, disabled with FALSE. | | | | | | | | |
| wLength | 0x0000 | | | | | | | | |
| | | | | | | | | | |

| MODE COMMANDS | | |
|------------------------|--|--------|
| 1-WIRE SPEED | | |
| Setup Packet Encoding: | | |
| bmRequestType | 0x40 | |
| bRequest | MODE_CMD | |
| wValue | MOD_1WIRE_SPEED | |
| wIndex | byte 2 | byte 1 |
| | byte 2 – 0x00 | |
| | byte 1 – 1-Wire speed: Code specifying communication speed of the 1-Wire bus. See MODE CONTROL section for code definition. | |
| | | |
| wLength | 0x0000 | |
| | | |
| STRONG PULLUP DURATION | | |
| Setup Packet Encoding: | | |
| bmRequestType | 0x40 | |
| bRequest | MODE_CMD | |
| wValue | MOD_STRONG_PU_DURATION | |
| wIndex | byte 2 | byte 1 |
| | byte 2 – 0x00 | |
| | byte 1 – Unsigned 8-bit binary number specifying the pullup duration. See MODE CONTROL section for byte bit weighting and additional format information. | |
| | | |
| wLength | 0x0000 | |
| | | |
| PULLDOWN SLEW RATE | | |
| Setup Packet Encoding: | | |
| bmRequestType | 0x40 | |
| bRequest | MODE_CMD | |
| wValue | MOD_PULLDOWN_SLEWRATE | |
| wIndex | byte 2 | byte 1 |
| | byte 2 – 0x00 | |
| | byte 1 – Code specifying the typical pulldown slew rate. See MODE CONTROL section for code definition. | |
| | | |
| wLength | 0x0000 | |
| | | |

| MODE COMMANDS | | |
|------------------------|--|--------|
| WRITE-1 LOW TIME | | |
| Setup Packet Encoding: | | |
| bmRequestType | 0x40 | |
| bRequest | MODE_CMD | |
| wValue | MOD_WRITE1_LOWTIME | |
| wIndex | byte 2 | byte 1 |
| | byte 2 – 0x00 | |
| | byte 1 – Code specifying the Write-1 low time duration. See MODE CONTROL section for code definition. | |
| wLength | 0x0000 | |
| | | |
| DSOW0 RECOVERY TIME | | |
| Setup Packet Encoding: | | |
| bmRequestType | 0x40 | |
| bRequest | MODE_CMD | |
| wValue | MOD_DSOW0_TREC | |
| wIndex | byte 2 | byte 1 |
| | byte 2 – 0x00 | |
| | byte 1 – Code specifying the recovery time for data sample offset and Write-0. See MODE CONTROL section for code definition. | |
| wLength | 0x0000 | |

APPENDIX 4: COMMAND, COMMAND TYPE CONSTANT CODES

| COMMAND TYPE CODES | |
|--------------------|----------|
| Name | bRequest |
| CONTROL_CMD | 0x00 |
| COMM_CMD | 0x01 |
| MODE_CMD | 0x02 |

| CONTROL COMMAND CODES | |
|-----------------------|--------|
| Name | wValue |
| CTL_RESET_DEVICE | 0x0000 |
| CTL_START_EXE | 0x0001 |
| CTL_RESUME_EXE | 0x0002 |
| CTL_HALT_EXE_IDLE | 0x0003 |
| CTL_HALT_EXE_DONE | 0x0004 |
| CTL_FLUSH_COMM_CMDS | 0x0007 |
| CTL_FLUSH_RCV_BUFFER | 0x0008 |
| CTL_FLUSH_XMT_BUFFER | 0x0009 |
| CTL_GET_COMM_CMDS | 0x000A |

| MODE COMMAND CODES | |
|------------------------|--------|
| Name | wValue |
| MOD_PULSE_EN | 0x0000 |
| MOD_SPEED_CHANGE_EN | 0x0001 |
| MOD_1WIRE_SPEED | 0x0002 |
| MOD_STRONG_PU_DURATION | 0x0003 |
| MOD_PULLDOWN_SLEWRATE | 0x0004 |
| (Reserved) | 0x0005 |
| MOD_WRITE1_LOWTIME | 0x0006 |
| MOD_DSOW0_TREC | 0x0007 |

Not Recommended for New Designs

REVISION HISTORY

| REVISION DATE | DESCRIPTION | PAGES CHANGED |
|--------------------------|--|--|
| 12/07 | <p>Removed 12V V_{PP} EPROM programming feature.</p> <p>Updated ordering info to show lead-free parts only. Removed note to contact the factory for flip chip.</p> <p>Deleted previous Figure 9 and Table 8.</p> <p>Deleted DC EC Table Notes 1 and 3 and renumbered Notes.</p> <p>Updated t_{LOW0}, t_{REC0}, t_{SLOT} values in Figure 7 to match actual silicon.</p> <p>Updated t_{LOW1}, t_{DSO}, t_{HIGH1} and t_{SLOT} values in Figure 6 to match actual silicon.</p> <p>Updated Tables 10, 11, and 12 to match actual silicon.</p> <p>Updated the text paragraph above Figure 6.</p> <p>Some of the updates in items 3 to 6 required numeric updates in related text sections.</p> | <p>1, 2, 4, 9–23, 28, 31–37, 40, 42–50</p> |
| 10/09 | <p>I_{DS} maximum changed from 490 to 200 μA, $V_{OL(SUSO)}$ test current changed from 4mA to 1mA.</p> | 34 |

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

[Maxim Integrated:](#)

[DS2490S](#) [DS2490X/T&R](#) [DS2490Y](#)