Fiber Channel/Gigabit Ethernet Laser Diode Control for Fiber Optic Modules

FEATURES

- Two Digitally Controlled Potentiometers (DCP's)
  - —100 Tap - 10kΩ
  - —256 Tap - 100kΩ
  - —Non-Volatile
  - —Write Protect Function
- 2kbit EEPROM Memory with Write Protect & Block Lock™
- 2-Wire industry standard Serial Interface
  - —Complies to the Gigabit Interface Converter (GBIC) specification
- Single Supply Operation
  - —2.7V to 5.5V
- **Hot Pluggable**
- 20 Ld TSSOP

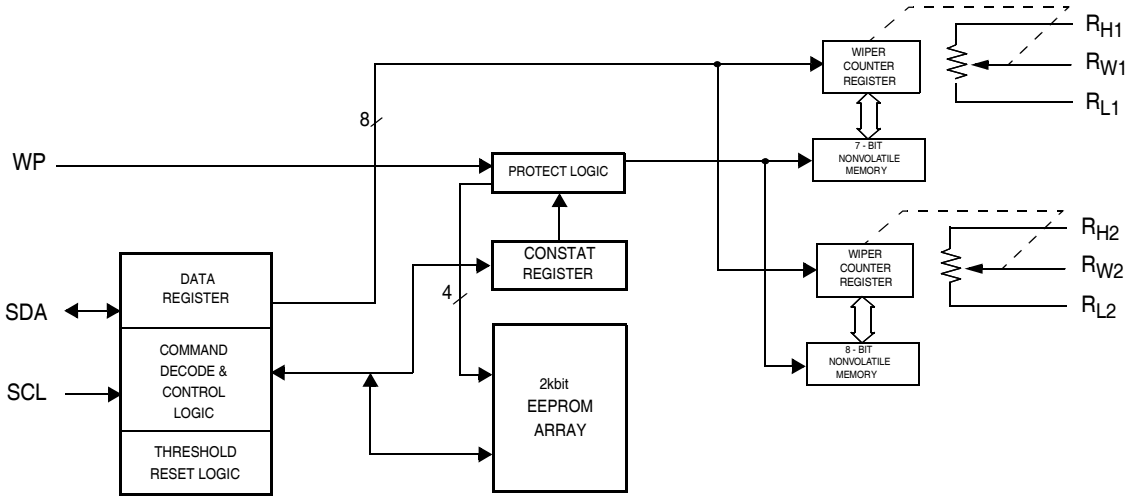
DESCRIPTION

The X9521 combines two Digitally Controlled Potentiometers (DCP's), and integrated EEPROM with Block Lock™ protection. All functions of the X9521 are accessed by an industry standard 2-Wire serial interface.

The DCP's of the X9521 may be utilized to control the bias and modulation currents of the laser diode in a Fiber Optic module. The 2kbit integrated EEPROM may be used to store module definition data.

The features of the X9521 are ideally suited to simplifying the design of fiber optic modules which comply to the Gigabit Interface Converter (GBIC) specification. The integration of these functions into one package significantly reduces board area, cost and increases reliability of laser diode modules.

BLOCK DIAGRAM



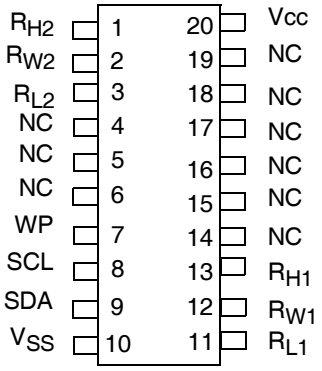
Ordering Information

PART NUMBER	PART MARKING	PRESET (FACTORY SHIPPED) V_{TRIPx} THRESHOLD LEVELS (x = 2, 3)	TEMP RANGE (°C)	PACKAGE
X9521V20I-A	X9521VIA	Optimized for 3.3V system monitoring	-40 to +85	20 Ld TSSOP
X9521V20I-B	X9521VIB	Optimized for 5V system monitoring	-40 to +85	20 Ld TSSOP
X9521V20IZ-A (Note)	X9521VZIA	Optimized for 3.3V system monitoring	-40 to +85	20 Ld TSSOP (Pb-free)
X9521V20IZ-B (Note)	X9521VZIB	Optimized for 5V system monitoring	-40 to +85	20 Ld TSSOP (Pb-free)

NOTE: Intersil Pb-free plus anneal products employ special Pb-free material sets; molding compounds/die attach materials and 100% matte tin plate termination finish, which are RoHS compliant and compatible with both SnPb and Pb-free soldering operations. Intersil Pb-free products are MSL classified at Pb-free peak reflow temperatures that meet or exceed the Pb-free requirements of IPC/JEDEC J STD-020.

PIN CONFIGURATION

20 Pin TSSOP



PIN ASSIGNMENT

Pin	Name	Function
1	R_{H2}	Connection to end of resistor array for (the 256 Tap) DCP 2.
2	R_{w2}	Connection to terminal equivalent to the “Wiper” of a mechanical potentiometer for DCP 2.
3	R_{L2}	Connection to other end of resistor array for (the 256 Tap) DCP2.
7	WP	Write Protect Control Pin. WP pin is a TTL level compatible input. When held HIGH, Write Protection is enabled. In the enabled state, this pin prevents all nonvolatile “write” operations. Also, when the Write Protection is enabled, and the device Block Lock feature is active (i.e. the Block Lock bits are NOT [0,0]), then no “write” (volatile or nonvolatile) operations can be performed in the device (including the wiper position of any of the integrated Digitally Controlled Potentiometers (DCPs). The WP pin uses an internal “pull-down” resistor, thus if left floating the write protection feature is disabled.
8	SCL	Serial Clock. This is a TTL level compatible input pin used to control the serial bus timing for data input and output.
9	SDA	Serial Data. SDA is a bidirectional TTL level compatible pin used to transfer data into and out of the device. The SDA pin input buffer is always active (not gated). This pin requires an external pull up resistor.
10	Vss	Ground.
11	R_{L1}	Connection to other end of resistor for (the 100 Tap) DCP 1.
12	R_{w1}	Connection to terminal equivalent to the “Wiper” of a mechanical potentiometer for DCP 1
13	R_{H1}	Connection to end of resistor array for (the 100 Tap) DCP 1.
20	Vcc	Supply Voltage.
4, 5, 6, 14, 15, 16, 17, 18, 19	NC	No connect.

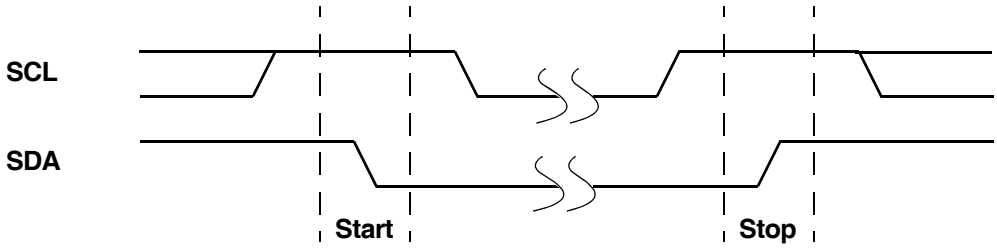


Figure 2. Valid Start and Stop Conditions

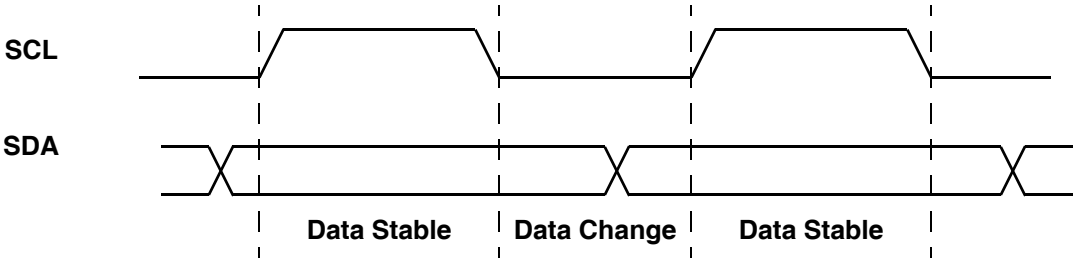


Figure 1. Valid Data Changes on the SDA Bus

PRINCIPLES OF OPERATION

SERIAL INTERFACE

Serial Interface Conventions

The device supports a bidirectional bus oriented protocol. The protocol defines any device that sends data onto the bus as a transmitter, and the receiving device as the receiver. The device controlling the transfer is called the master and the device being controlled is called the slave. The master always initiates data transfers, and provides the clock for both transmit and receive operations. Therefore, the X9521 operates as a slave in all applications.

Serial Clock and Data

Data states on the SDA line can change only while SCL is LOW. SDA state changes while SCL is HIGH are reserved for indicating START and STOP conditions. See Figure 1. On power-up of the X9521, the SDA pin is in the input mode.

Serial Start Condition

All commands are preceded by the START condition, which is a HIGH to LOW transition of SDA while SCL is HIGH. The device continuously monitors the SDA and SCL lines for the START condition and does not respond to any command until this condition has been met. See Figure 2.

Serial Stop Condition

All communications must be terminated by a STOP condition, which is a LOW to HIGH transition of SDA while SCL is HIGH. The STOP condition is also used to place the device into the Standby power mode after a read sequence. A STOP condition can only be issued after the transmitting device has released the bus. See Figure 2.

Serial Acknowledge

An ACKNOWLEDGE (ACK) is a software convention used to indicate a successful data transfer. The transmitting device, either master or slave, will release the bus after transmitting eight bits. During the ninth clock cycle, the receiver will pull the SDA line LOW to ACKNOWLEDGE that it received the eight bits of data. Refer to Figure 3.

The device will respond with an ACKNOWLEDGE after recognition of a START condition if the correct Device Identifier bits are contained in the Slave Address Byte. If a write operation is selected, the device will respond with an ACKNOWLEDGE after the receipt of each subsequent eight bit word.

In the read mode, the device will transmit eight bits of data, release the SDA line, then monitor the line for an ACKNOWLEDGE. If an ACKNOWLEDGE is detected and no STOP condition is generated by the master, the device will continue to transmit data. The device will terminate further data transmissions if an ACKNOWLEDGE is not detected. The master must then issue a STOP condition to place the device into a known state.

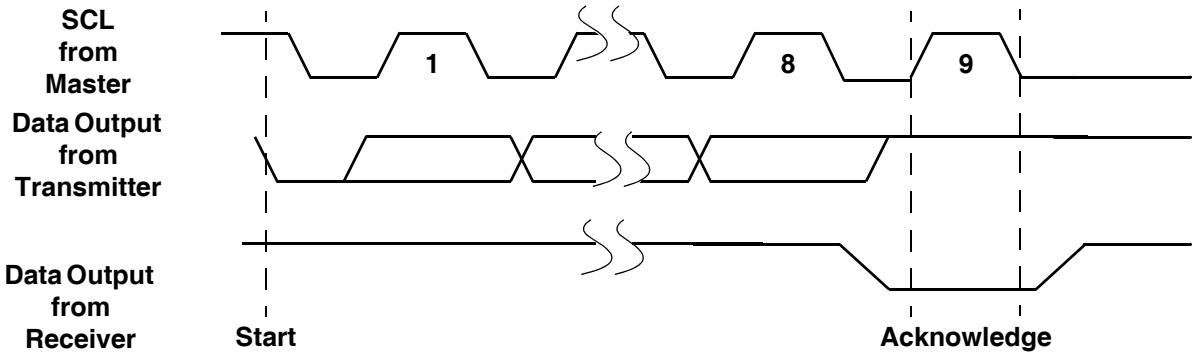


Figure 3. Acknowledge Response From Receiver

DEVICE INTERNAL ADDRESSING

Addressing Protocol Overview

The user addressable internal components of the X9521 can be split up into three main parts:

—Two Digitally Controlled Potentiometers (DCPs)

—EEPROM array

—Control and Status (CONSTAT) Register

Depending upon the operation to be performed on each of these individual parts, a 1, 2 or 3 Byte protocol is used. All operations however must begin with the Slave Address Byte being issued on the SDA pin. The Slave address selects the part of the X9521 to be addressed, and specifies if a Read or Write operation is to be performed.

It should be noted that in order to perform a write operation to either a DCP or the EEPROM array, the Write Enable Latch (WEL) bit must first be set (See “BL1, BL0: Block Lock protection bits - (Nonvolatile)” on page 12.)

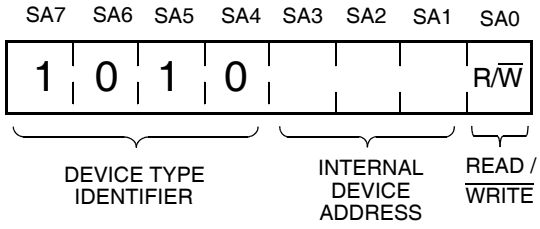
Slave Address Byte

Following a START condition, the master must output a Slave Address Byte (Refer to Figure 4.). This byte consists of three parts:

—The Device Type Identifier which consists of the most significant four bits of the Slave Address (SA7 - SA4). The Device Type Identifier must always be set to 1010 in order to select the X9521.

—The next three bits (SA3 - SA1) are the Internal Device Address bits. Setting these bits to 000 internally selects the EEPROM array, while setting these bits to 111 selects the DCP structures in the X9521. The CONSTAT Register may be selected using the Internal Device Address 010.

—The Least Significant Bit of the Slave Address (SA0) Byte is the R/$\overline{W}$ bit. This bit defines the operation to be performed on the device being addressed (as defined in the bits SA3 - SA1). When the R/$\overline{W}$ bit is “1”, then a READ operation is selected. A “0” selects a WRITE operation (Refer to Figure 4.)



Internal Address (SA3 - SA1)	Internally Addressed Device
000	EEPROM Array
010	CONSTAT Register
111	DCP

Bit SA0	Operation
0	WRITE
1	READ

Figure 4. Slave Address Format

Nonvolatile Write Acknowledge Polling

After a nonvolatile write command sequence (for either the EEPROM array, the Non Volatile Memory of a DCP (NVM), or the CONSTAT Register) has been correctly issued (including the final STOP condition), the X9521 initiates an internal high voltage write cycle. This cycle typically requires 5 ms. During this time, no further Read or Write commands can be issued to the device. Write Acknowledge Polling is used to determine when this high voltage write cycle has been completed.

To perform acknowledge polling, the master issues a START condition followed by a Slave Address Byte. The Slave Address issued must contain a valid Internal Device Address. The LSB of the Slave Address (R/$\overline{W}$) can be set to either 1 or 0 in this case. If the device is still busy with the high voltage cycle then no ACKNOWLEDGE will be returned. If the device has completed the write operation, an ACKNOWLEDGE will be returned and the host can then proceed with a read or write operation. (Refer to Figure 5.).

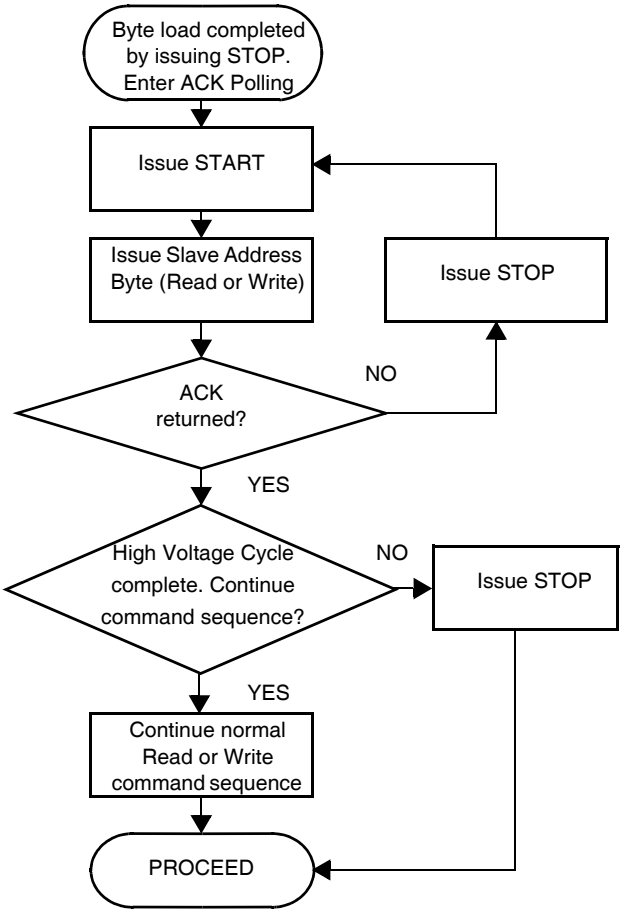


Figure 5. Acknowledge Polling Sequence

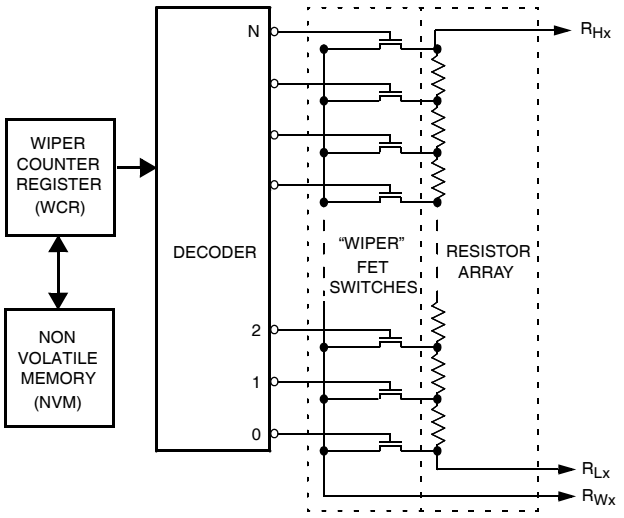


Figure 6. DCP Internal Structure

DIGITALLY CONTROLLED POTENTIOMETERS

DCP Functionality

The X9521 includes two independent resistor arrays. These arrays respectively contain 99 and 255 discrete resistive segments that are connected in series. The physical ends of each array are equivalent to the fixed terminals of a mechanical potentiometer (R_{Hx} and R_{Lx} inputs - where x = 1,2).

At both ends of each array and between each resistor segment there is a CMOS switch connected to the wiper (R_{Wx}) output. Within each individual array, only one switch may be turned on at any one time. These switches are controlled by the Wiper Counter Register (WCR) (See Figure 6). The WCR is a volatile register.

On power-up of the X9521, wiper position data is automatically loaded into the WCR from its associated Non Volatile Memory (NVM) Register. The Table below shows the Initial Values of the DCP WCR's before the contents of the NVM is loaded into the WCR.

DCP	Initial Values Before Recall
R_1 / 100 TAP	V_L / TAP = 0
R_2 / 256 TAP	V_H / TAP = 255

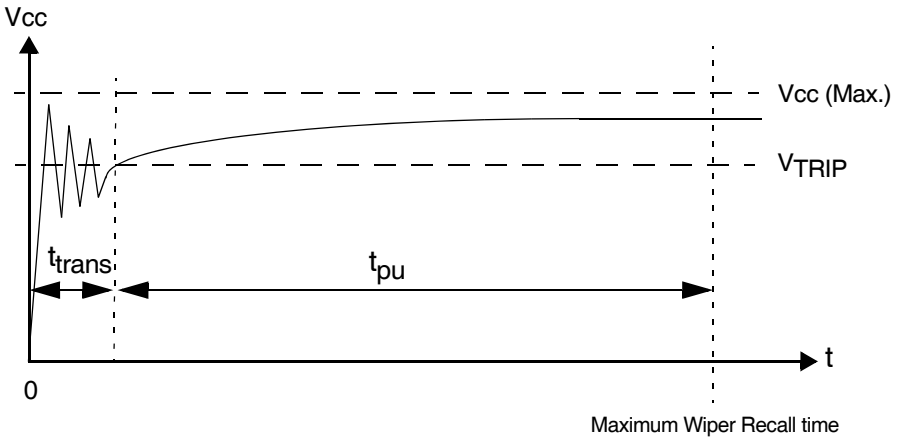


Figure 7. DCP Power-up

The data in the WCR is then decoded to select and enable one of the respective FET switches. A “make before break” sequence is used internally for the FET switches when the wiper is moved from one tap position to another.

Hot Pluggability

Figure 7 shows a typical waveform that the X9521 might experience in a Hot Pluggable situation. On power-up, Vcc applied to the X9521 may exhibit some amount of ringing, before it settles to the required value.

The device is designed such that the wiper terminal (R_{Wx}) is recalled to the correct position (as per the last stored in the DCP NVM), when the voltage applied to Vcc exceeds V_{TRIP} for a time exceeding t_{pu}.

Therefore, if t_{trans} is defined as the time taken for Vcc to settle above V_{TRIP} (Figure 7): then the desired wiper terminal position is recalled by (a maximum) time: t_{trans} + t_{pu}. It should be noted that t_{trans} is determined by system hot plug conditions.

DCP Operations

In total there are three operations that can be performed on any internal DCP structure:

—DCP Nonvolatile Write

—DCP Volatile Write

—DCP Read

A nonvolatile write to a DCP will change the “wiper position” by simultaneously writing new data to the associated WCR and NVM. Therefore, the new “wiper position” setting is recalled into the WCR after Vcc of the X9521 is powered down and then powered back up.

A volatile write operation to a DCP however, changes the “wiper position” by writing new data to the associated WCR only. The contents of the associated NVM register remains unchanged. Therefore, when Vcc to the device is powered down then back up, the “wiper position” reverts to that last position written to the DCP using a nonvolatile write operation.

Both volatile and nonvolatile write operations are executed using a three byte command sequence: (DCP) Slave Address Byte, Instruction Byte, followed by a Data Byte (See Figure 9).

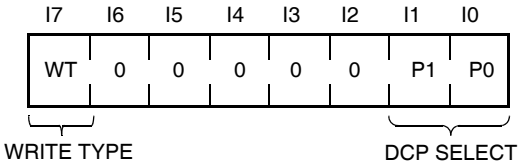
A DCP Read operation allows the user to “read out” the current “wiper position” of the DCP, as stored in the associated WCR. This operation is executed using the Random Address Read command sequence, consisting of the (DCP) Slave Address Byte followed by an Instruction Byte and the Slave Address Byte again (Refer to Figure 11.).

Instruction Byte

While the Slave Address Byte is used to select the DCP devices, an Instruction Byte is used to determine which DCP is being addressed.

The Instruction Byte (Figure 8) is valid only when the Device Type Identifier and the Internal Device Address bits of the Slave Address are set to 1010111. In this case, the two Least Significant Bit’s (I1 - I0) of the Instruction Byte are used to select the particular DCP (0 - 2). In the case of a Write to any of the DCPs (i.e. the LSB of the Slave Address is 0), the Most Significant Bit of the Instruction Byte (I7), determines the Write Type (WT) performed.

If WT is “1”, then a Nonvolatile Write to the DCP occurs. In this case, the “wiper position” of the DCP is changed by simultaneously writing new data to the associated WCR and NVM. Therefore, the new “wiper position” setting is recalled into the WCR after Vcc of the X9521 has been powered down then powered back up



WT†	Description
0	Select a Volatile Write operation to be performed on the DCP pointed to by bits P1 and P0
1	Select a Nonvolatile Write operation to be performed on the DCP pointed to by bits P1 and P0

†This bit has no effect when a Read operation is being performed.

Figure 8. Instruction Byte Format

If WT is “0” then a DCP Volatile Write is performed. This operation changes the DCP “wiper position” by writing new data to the associated WCR only. The contents of the associated NVM register remains unchanged. Therefore, when Vcc to the device is powered down then back up, the “wiper position” reverts to that last written to the DCP using a nonvolatile write operation.

DCP Write Operation

A write to DCPx (x = 1,2) can be performed using the three byte command sequence shown in Figure 9.

In order to perform a write operation on a particular DCP, the Write Enable Latch (WEL) bit of the CONSTAT Register must first be set (See “BL1, BL0: Block Lock protection bits - (Nonvolatile)” on page 12.)

The Slave Address Byte 10101110 specifies that a Write to a DCP is to be conducted. An ACKNOWLEDGE is returned by the X9521 after the Slave Address, if it has been received correctly.

Next, an Instruction Byte is issued on SDA. Bits P1 and P0 of the Instruction Byte determine which WCR is to be written, while the WT bit determines if the Write is to be volatile or nonvolatile. If the Instruction Byte format is valid, another ACKNOWLEDGE is then returned by the X9521.

Following the Instruction Byte, a Data Byte is issued to the X9521 over SDA. The Data Byte contents is latched into the WCR of the DCP on the first rising edge of the clock signal, after the LSB of the Data Byte (D0) has been issued on SDA (See Figure 25).

The Data Byte determines the “wiper position” (which FET switch of the DCP resistive array is switched ON) of the DCP. The maximum value for the Data Byte depends upon which DCP is being addressed (see Table below).

P1 - P0		DCPx	# Taps	Max. Data Byte
0	0	Reserved		
0	1	x = 1	100	Refer to Appendix 1
1	0	x = 2	256	FFh
1	1	Reserved		

Using a Data Byte larger than the values specified above results in the “wiper terminal” being set to the highest tap position. The “wiper position” does NOT roll-over to the lowest tap position.

For DCP2 (256 Tap), the Data Byte maps one to one to the “wiper position” of the DCP “wiper terminal”. Therefore, the Data Byte 00001111 (15_{10}) corresponds to setting the “wiper terminal” to tap position 15. Similarly, the Data Byte 00011100 (28_{10}) corresponds to setting the “wiper terminal” to tap position 28. The mapping of the Data Byte to “wiper position” data for DCP1 (100 Tap), is shown in “APPENDIX 1” . An example of a simple C language function which “translates” between the tap position (decimal) and the Data Byte (binary) for DCP1, is given in “APPENDIX 2” .

It should be noted that all writes to any DCP of the X9521 are random in nature. Therefore, the Data Byte of consecutive write operations to any DCP can differ by an arbitrary number of bits. Also, setting the bits (P1 = 0,

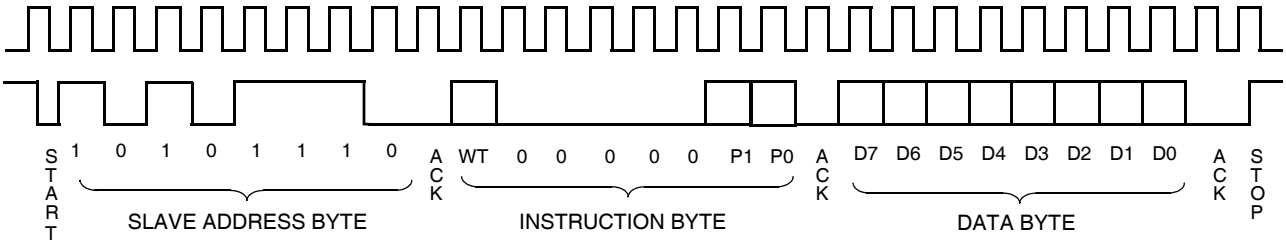


Figure 9. DCP Write Command Sequence

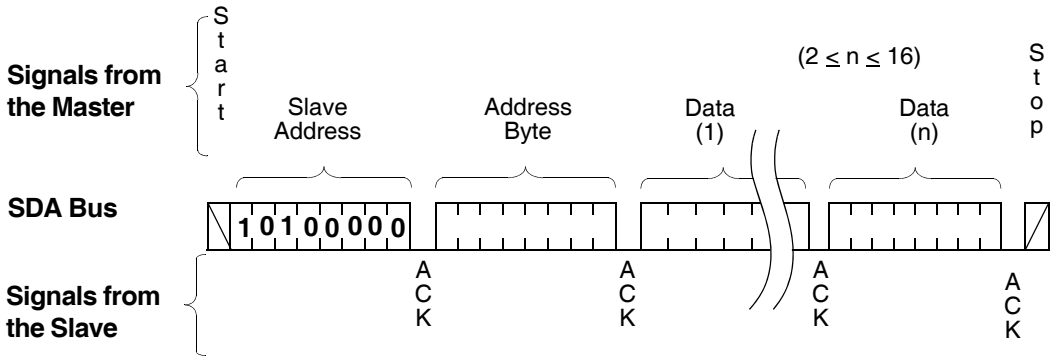


Figure 10. EEPROM Page Write Operation

P0 = 0) or (P1 = 1, P0 = 1) are reserved sequences, and will result in no ACKNOWLEDGE after sending an Instruction Byte on SDA.

The factory default setting of all “wiper position” settings is with 00h stored in the NVM of the DCPs. This corresponds to having the “wiper teminal” R_{WX} (x=1,2) at the “lowest” tap position, Therefore, the resistance between R_{WX} and R_{LX} is a minimum (essentially only the Wiper Resistance, R_W).

DCP Read Operation

A read of DCPx (x = 1,2) can be performed using the three byte random read command sequence shown in Figure 11.

The master issues the START condition and the Slave Address Byte 10101110 which specifies that a “dummy” write” is to be conducted. This “dummy” write operation sets which DCP is to be read (in the preceding Read operation). An ACKNOWLEDGE is returned by the X9521 after the Slave Address if received correctly. Next, an Instruction Byte is issued on SDA. Bits P1 - P0 of the Instruction Byte determine which DCP “wiper position” is to be read. In this case, the state of the WT bit is “don’t care”. If the Instruction Byte format is valid, then another ACKNOWLEDGE is returned by the X9521.

Following this ACKNOWLEDGE, the master immediately issues another START condition and a valid Slave address byte with the R/$\overline{W}$ bit set to 1. Then the X9521 issues an ACKNOWLEDGE followed by Data Byte, and finally, the master issues a STOP condition. The Data Byte read in this operation, corresponds to the “wiper position” (value of the WCR) of the DCP pointed to by bits P1 and P0.

It should be noted that when reading out the data byte for DCP1 (100 Tap), the upper most significant bit is an “unknown”. For DCP2 (256 Tap) however, all bits of the data byte are relevant (See Figure 11).

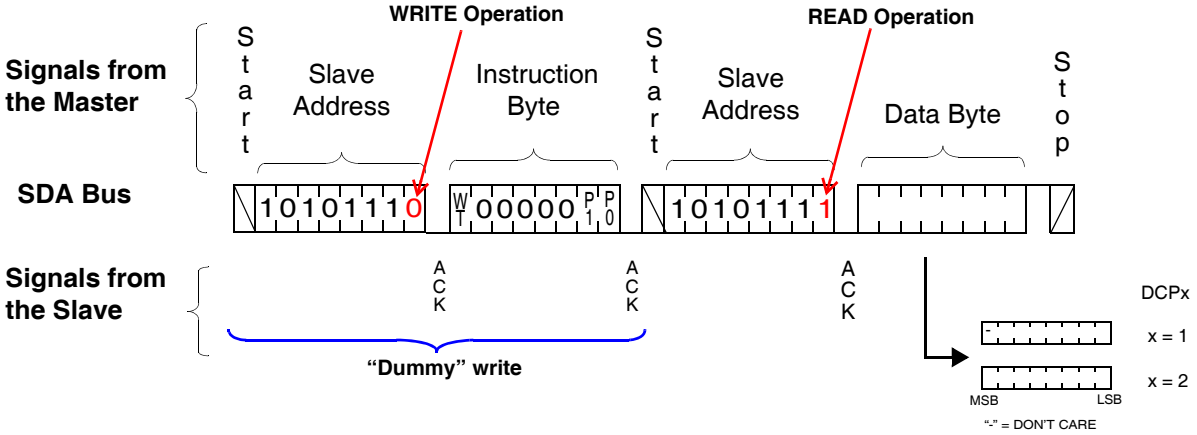


Figure 11. DCP Read Sequence

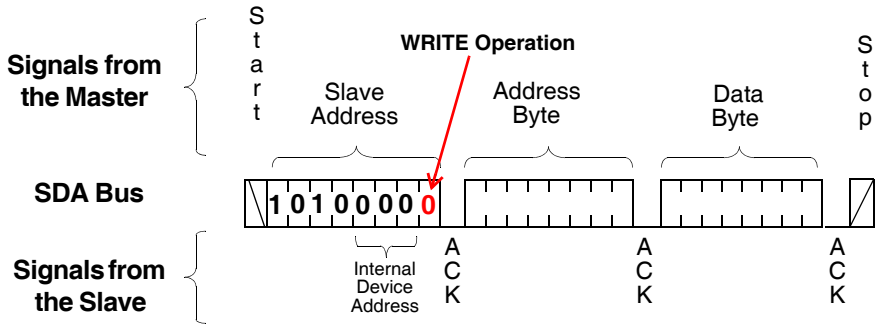


Figure 12. EEPROM Byte Write Sequence

2kbit EEPROM ARRAY

Operations on the 2kbit EEPROM Array, consist of either 1, 2 or 3 byte command sequences. All operations on the EEPROM must begin with the Device Type Identifier of the Slave Address set to 1010000. A Read or Write to the EEPROM is selected by setting the LSB of the Slave Address to the appropriate value R/$\overline{W}$ (Read = “1”, Write = ”0”).

In some cases when performing a Read or Write to the EEPROM, an Address Byte may also need to be specified. This Address Byte can contain the values 00h to FFh.

EEPROM Byte Write

In order to perform an EEPROM Byte Write operation to the EEPROM array, the Write Enable Latch (WEL) bit of the CONSTAT Register must first be set (See “BL1, BL0: Block Lock protection bits - (Nonvolatile)” on page 12.)

For a write operation, the X9521 requires the Slave Address Byte and an Address Byte. This gives the master access to any one of the words in the array. After receipt of the Address Byte, the X9521 responds with an ACKNOWLEDGE, and awaits the next eight bits of data. After receiving the 8 bits of the Data Byte, it again responds with an ACKNOWLEDGE. The master then terminates the transfer by generating a STOP condition, at which time the X9521 begins the internal write cycle to the nonvolatile memory (See Figure 12). During this internal write cycle, the X9521 inputs are disabled, so it does not respond to any requests from the master. The SDA output is at high impedance. A write to a region of EEPROM memory which has been protected with the Block-Lock feature (See “BL1, BL0: Block Lock protection bits - (Nonvolatile)” on page 12.), suppresses the ACKNOWLEDGE bit after the Address Byte.

EEPROM Page Write

In order to perform an EEPROM Page Write operation to the EEPROM array, the Write Enable Latch (WEL) bit of the CONSTAT Register must first be set (See “BL1, BL0: Block Lock protection bits - (Nonvolatile)” on page 12.)

The X9521 is capable of a page write operation. It is initiated in the same manner as the byte write operation; but instead of terminating the write cycle after the first data byte is transferred, the master can transmit an unlimited number of 8-bit bytes. After the receipt of each byte, the X9521 responds with an ACKNOWLEDGE, and the address is internally incremented by one. The page address remains constant. When the counter reaches the end of the page, it “rolls over” and goes back to ‘0’ on the same page.

For example, if the master writes 12 bytes to the page starting at location 11 (decimal), the first 5 bytes are written to locations 11 through 15, while the last 7 bytes are written to locations 0 through 6. Afterwards, the address counter would point to location 7. If the master supplies more than 16 bytes of data, then new data overwrites the previous data, one byte at a time (See Figure 13).

The master terminates the Data Byte loading by issuing a STOP condition, which causes the X9521 to begin the nonvolatile write cycle. As with the byte write operation, all inputs are disabled until completion of the internal write cycle. See Figure 10 for the address, ACKNOWLEDGE, and data transfer sequence.

Stops and EEPROM Write Modes

Stop conditions that terminate write operations must be sent by the master after sending at least 1 full data byte and receiving the subsequent ACKNOWLEDGE signal. If the master issues a STOP within a Data Byte, or before the X9521 issues a corresponding ACKNOWLEDGE, the X9521 cancels the write operation. Therefore, the contents of the EEPROM array does not change.

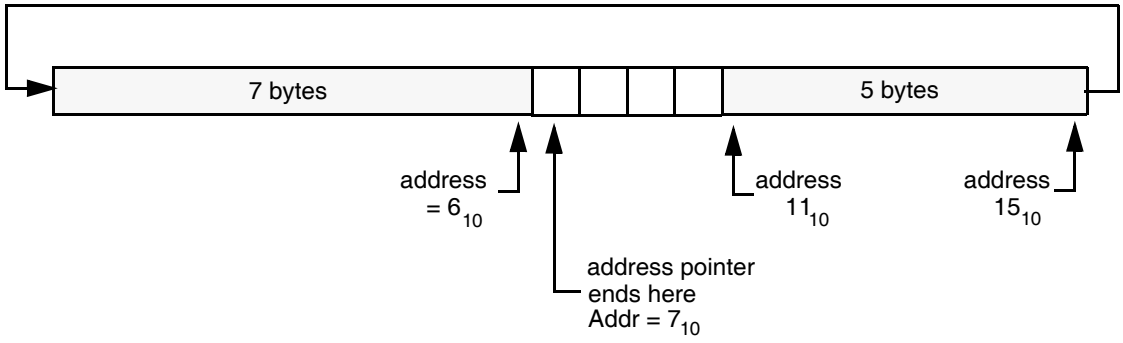


Figure 13. Example: Writing 12 bytes to a 16-byte page starting at location 11.

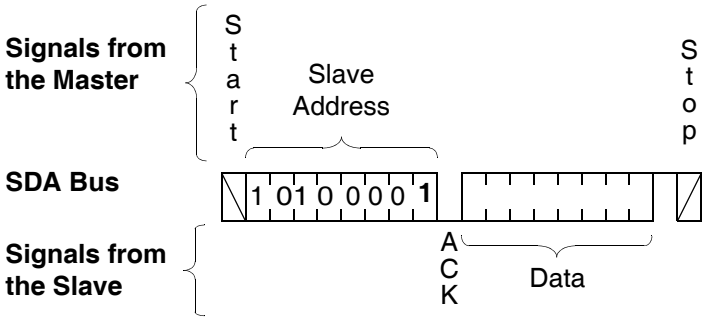


Figure 14. Current EEPROM Address Read Sequence

EEPROM Array Read Operations

Read operations are initiated in the same manner as write operations with the exception that the R/$\overline{W}$ bit of the Slave Address Byte is set to one. There are three basic read operations: Current EEPROM Address Read, Random EEPROM Read, and Sequential EEPROM Read.

Current EEPROM Address Read

Internally the device contains an address counter that maintains the address of the last word read incremented by one. Therefore, if the last read was to address n, the next read operation would access data from address n+1. On power-up, the address of the address counter is undefined, requiring a read or write operation for initialization.

Upon receipt of the Slave Address Byte with the R/$\overline{W}$ bit set to one, the device issues an ACKNOWLEDGE and then transmits the eight bits of the Data Byte. The master terminates the read operation when it does not respond with an ACKNOWLEDGE during the ninth clock and then issues a STOP condition (See Figure 14 for the address, ACKNOWLEDGE, and data transfer sequence).

It should be noted that the ninth clock cycle of the read operation is not a “don’t care.” To terminate a read operation, the master must either issue a STOP condition during the ninth cycle or hold SDA HIGH during the ninth clock cycle and then issue a STOP condition.

Another important point to note regarding the “Current EEPROM Address Read” , is that this operation is not available if the last executed operation was an access to a DCP or the CONSTAT Register (i.e.: an operation using the Device Type Identifier 1010111 or 1010010). Immediately after an operation to a DCP or CONSTAT Register is performed, only a “Random EEPROM Read” is available. Immediately following a “Random EEPROM Read” , a “Current EEPROM Address Read” or “Sequential EEPROM Read” is once again available (assuming that no access to a DCP or CONSTAT Register occur in the interim).

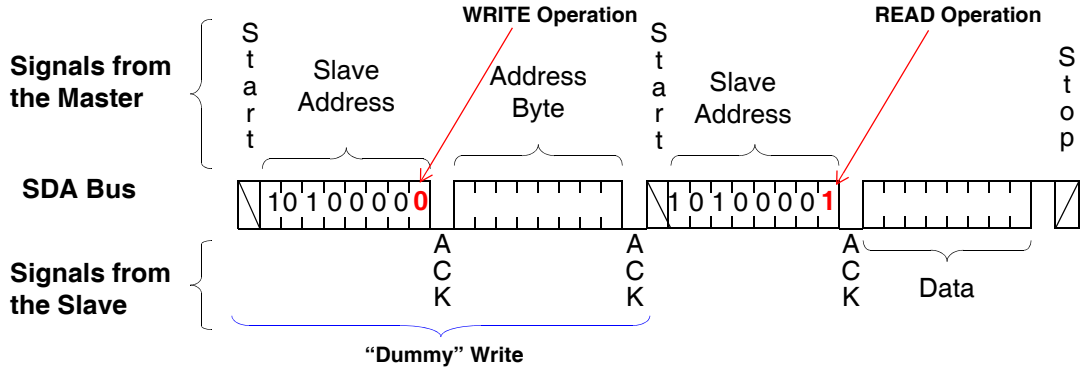


Figure 15. Random EEPROM Address Read Sequence

Random EEPROM Read

Random read operation allows the master to access any memory location in the array. Prior to issuing the Slave Address Byte with the R/$\overline{W}$ bit set to one, the master must first perform a "dummy" write operation. The master issues the START condition and the Slave Address Byte, receives an ACKNOWLEDGE, then issues an Address Byte. This "dummy" Write operation sets the address pointer to the address from which to begin the random EEPROM read operation.

After the X9521 acknowledges the receipt of the Address Byte, the master immediately issues another START condition and the Slave Address Byte with the R/$\overline{W}$ bit set to one. This is followed by an ACKNOWLEDGE from the X9521 and then by the eight bit word. The master terminates the read operation by not responding with an ACKNOWLEDGE and instead issuing a STOP condition (Refer to Figure 15.).

A similar operation called "Set Current Address" also exists. This operation is performed if a STOP is issued instead of the second START shown in Figure 15. In this case, the device sets the address pointer to that of the Address Byte, and then goes into standby mode after the STOP bit. All bus activity will be ignored until another START is detected.

Sequential EEPROM Read

Sequential reads can be initiated as either a current address read or random address read. The first Data Byte is transmitted as with the other modes; however, the master now responds with an ACKNOWLEDGE, indicating it requires additional data. The X9521 continues to output a Data Byte for each ACKNOWLEDGE received. The master terminates the read operation by not responding with an ACKNOWLEDGE and instead issuing a STOP condition.

The data output is sequential, with the data from address n followed by the data from address n + 1. The address counter for read operations increments through the entire memory contents to be serially read during one operation. At the end of the address space the counter "rolls over" to address 00h and the device continues to output data for each ACKNOWLEDGE received (Refer to Figure 16.).

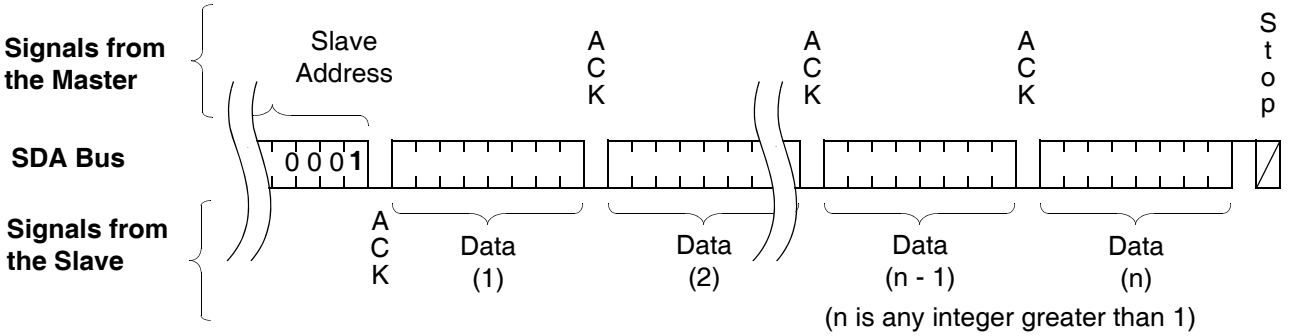
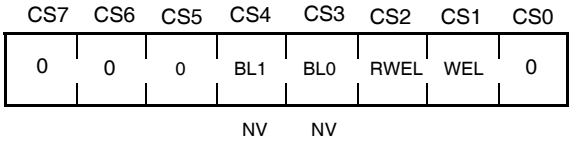


Figure 16. Sequential EEPROM Read Sequence



Bit(s)	Description
CS7 - CS5	Always “0”(RESERVED)
BL1 - BL0	Sets the Block Lock partition
RWEL	Register Write Enable Latch bit
WEL	Write Enable Latch bit
CS0	Always “0” (RESERVED)

NOTE: Bits labelled NV are nonvolatile (See “CONTROL AND STATUS REGISTER”).

Figure 17. CONSTAT Register Format

CONTROL AND STATUS REGISTER

The Control and Status (CONSTAT) Register provides the user with a mechanism for changing and reading the status of various parameters of the X9521 (See Figure 17).

The CONSTAT register is a combination of both volatile and nonvolatile bits. The nonvolatile bits of the CONSTAT register retain their stored values even when Vcc is powered down, then powered back up. The volatile bits however, will always power-up to a known logic state “0” (irrespective of their value at power-down).

A detailed description of the function of each of the CONSTAT register bits follows:

WEL: Write Enable Latch (Volatile)

The WEL bit controls the Write Enable status of the entire X9521 device. This bit must first be enabled before ANY write operation (to DCPs, EEPROM memory array, or the CONSTAT register). If the WEL bit is not first enabled, then ANY proceeding (volatile or nonvolatile) write operation to DCPs, EEPROM array, as well as the CONSTAT register, is aborted and no ACKNOWLEDGE is issued after a Data Byte.

The WEL bit is a volatile latch that powers up in the disabled, LOW (0) state. The WEL bit is enabled / set by writing 00000010 to the CONSTAT register. Once enabled, the WEL bit remains set to “1” until either it is reset to “0” (by writing 00000000 to the CONSTAT register) or until the X9521 powers down, and then up again.

Writes to the WEL bit do not cause an internal high voltage write cycle. Therefore, the device is ready for another operation immediately after a STOP condition is executed in the CONSTAT Write command sequence (See Figure 18).

RWEL: Register Write Enable Latch (Volatile)

The RWEL bit controls the (CONSTAT) Register Write Enable status of the X9521. Therefore, in order to write to any of the bits of the CONSTAT Register (except WEL), the RWEL bit must first be set to “1”. The RWEL bit is a volatile bit that powers up in the disabled, LOW (“0”) state.

It must be noted that the RWEL bit can only be set, once the WEL bit has first been enabled (See "CONSTAT Register Write Operation").

The RWEL bit will reset itself to the default “0” state, in one of three cases:

—After a successful write operation to any bits of the CONSTAT register has been completed (See Figure 18).

—When the X9521 is powered down.

—When attempting to write to a Block Lock protected region of the EEPROM memory (See "BL1, BL0: Block Lock protection bits - (Nonvolatile)", below).

BL1, BL0: Block Lock protection bits - (Nonvolatile)

The Block Lock protection bits (BL1 and BL0) are used to:

—Inhibit a write operation from being performed to certain addresses of the EEPROM memory array

—Inhibit a DCP write operation (changing the “wiper position”).

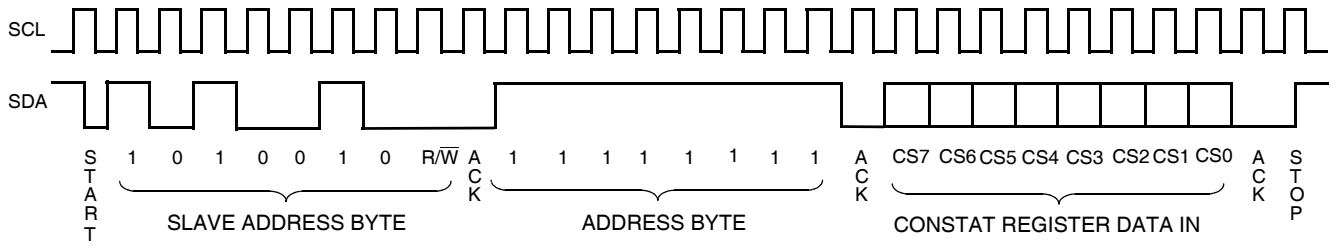


Figure 18. CONSTAT Register Write Command Sequence

The region of EEPROM memory which is protected / locked is determined by the combination of the BL1 and BL0 bits written to the CONSTAT register. It is possible to lock the regions of EEPROM memory shown in the table below:

BL1	BL0	Protected Addresses (Size)	Partition of array locked
0	0	None (Default)	None (Default)
0	1	C0h - FFh (64 bytes)	Upper 1/4
1	0	80h - FFh (128 bytes)	Upper 1/2
1	1	00h - FFh (256 bytes)	All

If the user attempts to perform a write operation on a protected region of EEPROM memory, the operation is aborted without changing any data in the array.

When the Block Lock bits of the CONSTAT register are set to something other than BL1 = 0 and BL0 = 0, then the “wiper position” of the DCPs cannot be changed - i.e. DCP write operations cannot be conducted:

BL1	BL0	DCP Write Operation Permissible
0	0	YES (Default)
0	1	NO
1	0	NO
1	1	NO

The factory default setting for these bits are BL1 = 0, BL0 = 0.

IMPORTANT NOTE: If the Write Protect (WP) pin of the X9521 is active (HIGH), then all nonvolatile write operations to both the EEPROM memory and DCPs are inhibited, irrespective of the Block Lock bit settings (See "WP: Write Protection Pin").

CONSTAT Register Write Operation

The CONSTAT register is accessed using the Slave Address set to 1010010 (Refer to Figure 4.). Following the Slave Address Byte, access to the CONSTAT register requires an Address Byte which must be set to FFh.

Only one data byte is allowed to be written for each CONSTAT register Write operation. The user must issue a STOP, after sending this byte to the register, to initiate the nonvolatile cycle that stores the BP1and BP0 bits. The X9521 will not ACKNOWLEDGE any data bytes written after the first byte is entered (Refer to Figure 18.).

When writing to the CONSTAT register, the bits CS7-CS5 and CS0 must all be set to “0”. Writing any other bit sequence to bits CS7-CS5 and CS0 of the CONSTAT register is reserved.

Prior to writing to the CONSTAT register, the WEL and RWEL bits must be set using a two step process, with the whole sequence requiring 3 steps

—Write a 02H to the CONSTAT Register to set the Write Enable Latch (WEL). This is a volatile operation, so there is no delay after the write. (Operation preceded by a START and ended with a STOP).

—Write a 06H to the CONSTAT Register to set the Register Write Enable Latch (RWEL) AND the WEL bit. This is also a volatile cycle. The zeros in the data byte are required. (Operation preceded by a START and ended with a STOP).

—Write a one byte value to the CONSTAT Register that has all the bits set to the desired state. The CONSTAT register can be represented as 000st010 in binary, where st are the Block Lock Protection (BL1 and BL0) bits. This operation is proceeded by a START and ended with a STOP bit. Since this is a nonvolatile write cycle, it will typically take 5ms to complete. The RWEL bit is reset by this cycle and the sequence must be repeated to change the nonvolatile bits again. If bit 2 is set to ‘1’ in this third step (000s t110) then the RWEL bit is set, but the BL1 and BL0 bits remain unchanged. Writing a second byte to the control register is not allowed. Doing so aborts the write operation and the X9521 does not return an ACKNOWLEDGE.

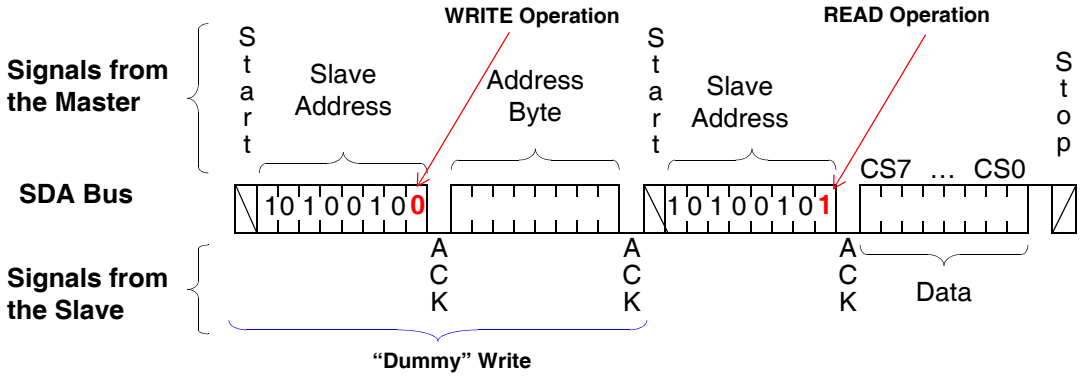


Figure 19. CONSTAT Register Read Command Sequence

For example, a sequence of writes to the device CONSTAT register consisting of [02H, 06H, 02H] will reset the BL0 and BL0 bits in the CONSTAT Register to "0".

It should be noted that a write to any nonvolatile bit of CONSTAT register will be ignored if the Write Protect pin of the X9521 is active (HIGH) (See "WP: Write Protection Pin").

CONSTAT Register Read Operation

The contents of the CONSTAT Register can be read at any time by performing a random read (See Figure 19). Using the Slave Address Byte set to 10100101, and an Address Byte of FFh. Only one byte is read by each register read operation. The X9521 resets itself after the first byte is read. The master should supply a STOP condition to be consistent with the bus protocol.

After setting the WEL and / or the RWEL bit(s) to a "1", a CONSTAT register read operation may occur, without interrupting a proceeding CONSTAT register write operation.

When reading the contents of the CONSTAT register, the bits CS7 - CS5 and CS0 will always return "0".

DATA PROTECTION

There are a number of levels of data protection features designed into the X9521. Any write to the device first requires setting of the WEL bit in the CONSTAT register. A write to the CONSTAT register itself, further requires the setting of the RWEL bit. Block Lock protection of the device enables the user to inhibit writes to certain regions of the EEPROM memory, as well as to all the DCPs. One further level of data protection in the X9521, is incorporated in the form of the Write Protection pin.

WP: Write Protection Pin

When the Write Protection (WP) pin is active (HIGH), it disables nonvolatile write operations to the X9521.

The table below (X9521 Write Permission Status) summarizes the effect of the WP pin (and Block Lock), on the write permission status of the device.

Additional Data Protection Features

In addition to the preceding features, the X9521 also incorporates the following data protection functionality:

—The proper clock count and data bit sequence is required prior to the STOP bit in order to start a nonvolatile write cycle.

X9521 Write Permission Status

Block Lock Bits		WP	DCP Volatile Write Permitted	DCP Nonvolatile Write Permitted	Write to EEPROM Permitted	Write to CONSTAT Register Permitted	
BL0	BL1					Volatile Bits	Nonvolatile Bits
x	1	1	NO	NO	NO	NO	NO
1	x	1	NO	NO	NO	NO	NO
0	0	1	YES	NO	NO	NO	NO
x	1	0	NO	NO	Not in locked region	YES	YES
1	x	0	NO	NO	Not in locked region	YES	YES
0	0	0	YES	YES	Yes (All Array)	YES	YES

ABSOLUTE MAXIMUM RATINGS

Parameter	Min.	Max.	Units
Temperature under Bias	-65	+135	°C
Storage Temperature	-65	+150	°C
Voltage on WP pin (With respect to Vss)	-1.0	+15	V
Voltage on other pins (With respect to Vss)	-1.0	+7	V
| Voltage on R_{Hx} - Voltage on R_{Lx} | (x = 1,2. Referenced to Vss)		Vcc	V
D.C. Output Current (SDA)	0	5	mA
Lead Temperature (Soldering, 10 seconds)		300	°C
Supply Voltage Limits (Applied Vcc voltage, referenced to Vss)	2.7	5.5	V

RECOMMENDED OPERATING CONDITIONS

Temperature	Min.	Max.	Units
Industrial	-40	+85	°C

NOTE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and the functional operation of the device at these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability

Figure 20. Equivalent A.C. Circuit

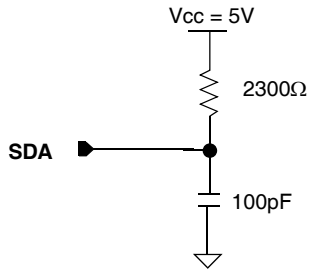
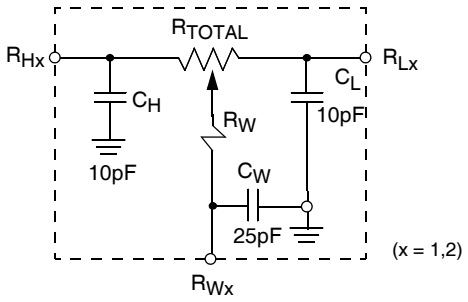


Figure 21. DCP SPICE Macromodel



TIMING DIAGRAMS

Figure 22. Bus Timing

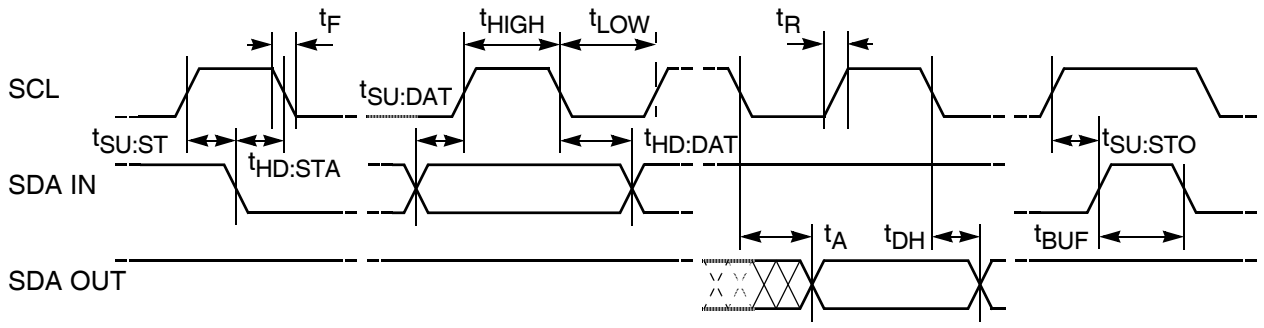


Figure 23. WP Pin Timing

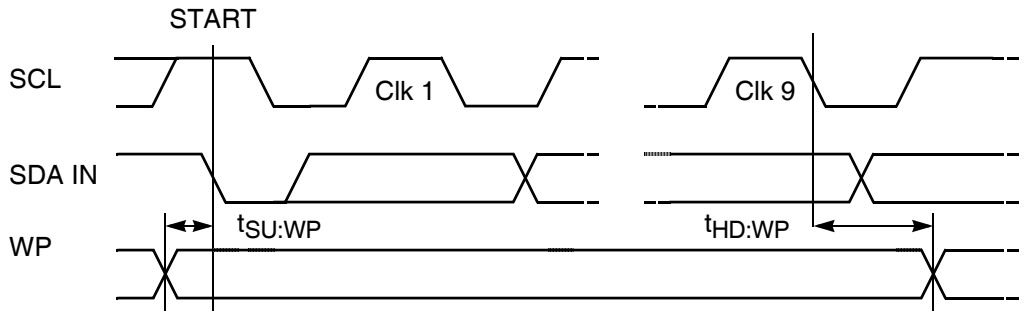


Figure 24. Write Cycle Timing

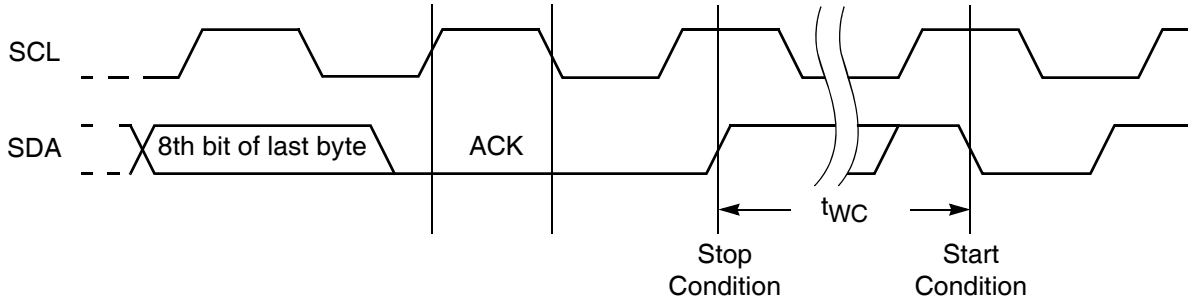
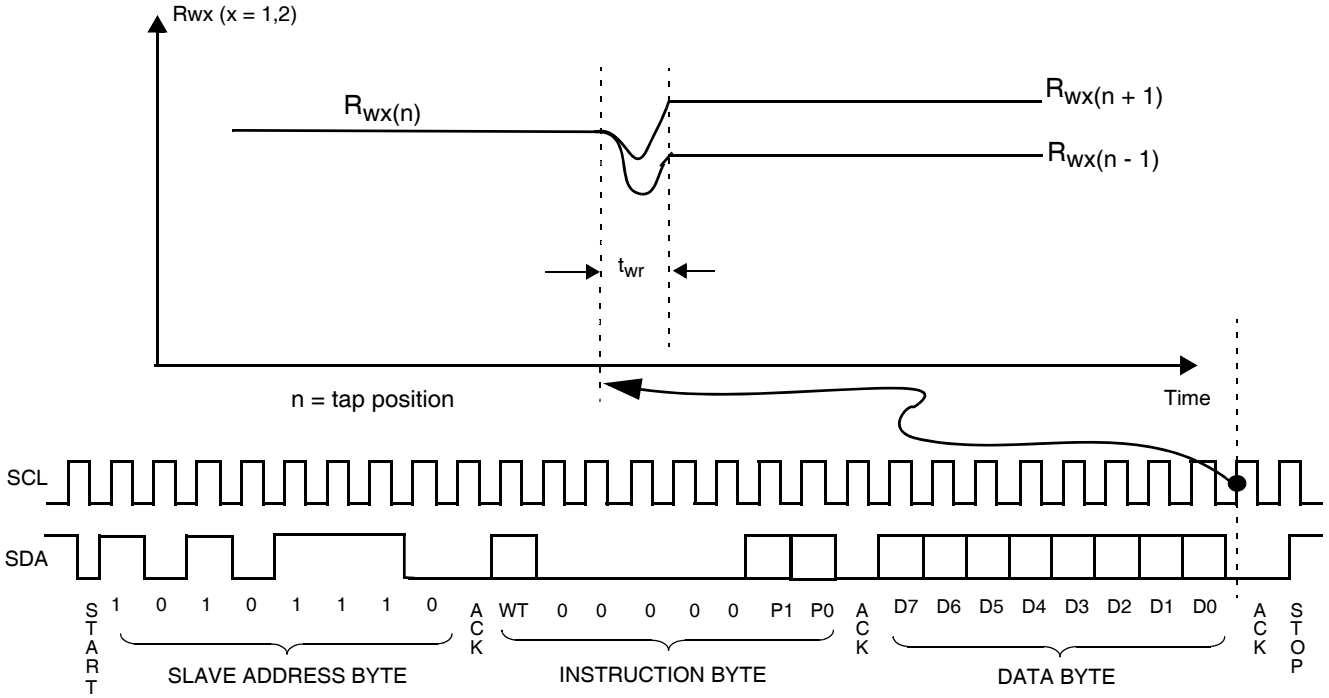


Figure 25. DCP "Wiper Position" Timing



D.C. OPERATING CHARACTERISTICS

Symbol	Parameter	Min	Typ	Max	Unit	Test Conditions / Notes
I_{CC1} (1)	Current into V_{CC} Pin (X9521: Active) Read memory array (3) Write nonvolatile memory			0.4 1.5	mA	f_{SCL} = 400kHz
I_{CC2} (2)	Current into V_{CC} Pin (X9521:Standby) With 2-Wire bus activity (3) No 2-Wire bus activity			50 50	µA	$V_{SDA} = V_{CC}$ WP = Vss or Open/Floating $V_{SCL} = V_{CC}$ (when no bus activity else f_{SCL} = 400kHz)
I_{LI}	Input Leakage Current (SCL, SDA)		0.1	10	µA	V_{IN} (4) = GND to V_{CC}.
	Input Leakage Current (WP)			10	µA	
I_{ai}	Analog Input Leakage		1	10	µA	$V_{IN} = V_{SS}$ to V_{CC} with all other analog pins floating
I_{LO}	Output Leakage Current (SDA)		0.1	10	µA	V_{OUT} (5) = GND to V_{CC}. X9521 is in Standby(2)
V_{IL} (6)	Input LOW Voltage (SCL, SDA, WP)	-0.5		0.8	V	
V_{IH} (6)	Input HIGH Voltage (SCL,SDA, WP)	2.0		V_{CC} +0.5	V	
V_{OLx}	SDA Output Low Voltage			0.4	V	I_{SINK} = 2.0mA

Notes: 1. The device enters the Active state after any START, and remains active until: 9 clock cycles later if the Device Select Bits in the Slave Address Byte are incorrect; 200nS after a STOP ending a read operation; or t_{WC} after a STOP ending a write operation.

Notes: 2.The device goes into Standby: 200nS after any STOP, except those that initiate a high voltage write cycle; t_{WC} after a STOP that initiates a high voltage cycle; or 9 clock cycles after any START that is not followed by the correct Device Select Bits in the Slave Address Byte.

Notes: 3.Current through external pull up resistor not included.

Notes: 4.V_{IN} = Voltage applied to input pin.

Notes: 5.V_{OUT} = Voltage applied to output pin.

Notes: 6.V_{IL} Min. and V_{IH} Max. are for reference only and are not tested.

A.C. CHARACTERISTICS (See Figure 22, Figure 23, Figure 24)

Symbol	Parameter	400kHz		
		Min	Max	Units
f_{SCL}	SCL Clock Frequency	0	400	kHz
t_{IN} (5)	Pulse width Suppression Time at inputs	50		ns
t_{AA} (5)	SCL LOW to SDA Data Out Valid	0.1	0.9	µs
t_{BUF} (5)	Time the bus free before start of new transmission	1.3		µs
t_{LOW}	Clock LOW Time	1.3		µs
t_{HIGH}	Clock HIGH Time	0.6		µs
$t_{SU:STA}$	Start Condition Setup Time	0.6		µs
$t_{HD:STA}$	Start Condition Hold Time	0.6		µs
$t_{SU:DAT}$	Data In Setup Time	100		ns
$t_{HD:DAT}$	Data In Hold Time	0		µs
$t_{SU:STO}$	Stop Condition Setup Time	0.6		µs
t_{DH} (5)	Data Output Hold Time	50		ns
t_R (5)	SDA and SCL Rise Time	20 +.1Cb (2)	300	ns
t_F (5)	SDA and SCL Fall Time	20 +.1Cb (2)	300	ns
$t_{SU:WP}$	WP Setup Time	0.6		µs
$t_{HD:WP}$	WP Hold Time	0		µs
Cb	Capacitive load for each bus line		400	pF

A.C. TEST CONDITIONS

Input Pulse Levels	$0.1V_{CC}$ to $0.9V_{CC}$
Input Rise and Fall Times	10ns
Input and Output Timing Levels	$0.5V_{CC}$
Output Load	See Figure 20

NONVOLATILE WRITE CYCLE TIMING

Symbol	Parameter	Min.	Typ.(1)	Max.	Units
t_{WC}(4)	Nonvolatile Write Cycle Time		5	10	ms

CAPACITANCE (T_A = 25°C, f = 1.0 MHz, V_{CC} = 5V)

Symbol	Parameter	Max	Units	Test Conditions
C_{OUT} (5)	Output Capacitance (SDA, V1RO, V2RO, V3RO)	8	pF	V_{OUT} = 0V
C_{IN} (5)	Input Capacitance (SCL, WP)	6	pF	V_{IN} = 0V

Notes: 1. Typical values are for T_A = 25°C and V_{CC} = 5.0V

Notes: 2.Cb = total capacitance of one bus line in pF.

Notes: 3.Over recommended operating conditions, unless otherwise specified

Notes: 4.t_{WC} is the time from a valid STOP condition at the end of a write sequence to the end of the self-timed internal nonvolatile write cycle. It is the minimum cycle time to be allowed for any nonvolatile write by the user, unless Acknowledge Polling is used.

Notes: 5.This parameter is not 100% tested.

POTENTIOMETER CHARACTERISTICS

Symbol	Parameter	Limits				Test Conditions/Notes
		Min.	Typ.	Max.	Units	
R_{TOL}	End to End Resistance Tolerance	-20		+20	%	
V_{RHx}	R_H Terminal Voltage (x = 1,2)	Vss		V_{CC}	V	
V_{RLx}	R_L Terminal Voltage (x = 1,2)	Vss		V_{CC}	V	
P_R	Power Rating (1)(6)			10	mW	R_{TOTAL} = 10kΩ (DCP1)
				5	mW	R_{TOTAL} = 100kΩ (DCP2)
R_W	DCP Wiper Resistance		200	400	Ω	I_W = 1mA, V_{CC} = 5 V, V_{RHx} = Vcc, V_{RLx} = Vss (x = 1,2).
			400	1200	Ω	I_W = 1mA, V_{CC} = 2.7 V, V_{RHx} = Vcc, V_{RLx} = Vss (x = 1,2)
I_W	Wiper Current (6)			4.4	mA	
	Noise				mV/ sqt(Hz)	R_{TOTAL} = 10kΩ (DCP1)
					mV/ sqt(Hz)	R_{TOTAL} = 100kΩ (DCP2)
	Absolute Linearity (2)	-1		+1	MI(4)	$R_{w(n)(actual)}$ - $R_{w(n)(expected)}$
	Relative Linearity (3)	-1		+1	MI(4)	$R_{w(n+1)}$ - [$R_{w(n)}$ + MI]
	R_{TOTAL} Temperature Coefficient		±300		ppm/°C	R_{TOTAL} = 10kΩ (DCP1)
			±300		ppm/°C	R_{TOTAL} = 100kΩ (DCP2)
$C_H/C_L/C_W$	Potentiometer Capacitances		10/10/25		pF	See Figure 21.
t_{wcr}	Wiper Response time (6)			200	µs	See Figure 25.
V_{TRIP}	Vcc power-up DCP recall threshold				V	
t_{PU}	Vcc power-up DCP recall delay time (6)	25	50	75	ms	

Notes: 1. Power Rating between the wiper terminal $R_{WX(n)}$ and the end terminals R_{HX} or R_{LX} - for ANY tap position n, (x = 1,2).

Notes: 2.Absolute Linearity is utilized to determine actual wiper resistance versus, expected resistance = ($R_{wx(n)}$(actual) - $R_{wx(n)}$(expected)) = ±1 MI Maximum (x = 1,2).

Notes: 3.Relative Linearity is a measure of the error in step size between taps = $R_{Wx(n+1)}$ - [$R_{wx(n)}$ + MI] = ±1 MI (x = 0,1,2)

Notes: 4.1 MI = Minimum Increment = R_{TOT} / (Number of taps in DCP - 1).

Notes: 5.Typical values are for T_A = 25°C and nominal supply voltage.

Notes: 6.This parameter is periodically sampled and not 100% tested.

APPENDIX 1

DCP1 (100 Tap) Tap position to Data Byte translation Table

Tap Position	Data Byte	
	Decimal	Binary
0	0	0000 0000
1	1	0000 0001
.	.	.
.	.	.
23	23	0001 0111
24	24	0001 1000
25	56	0011 1000
26	55	0011 0111
.	.	.
.	.	.
48	33	0010 0001
49	32	0010 0000
50	64	0100 0000
51	65	0100 0001
.	.	.
.	.	.
73	87	0101 0111
74	88	0101 1000
75	120	0111 1000
76	119	0111 0111
.	.	.
.	.	.
98	97	0110 0001
99	96	0110 0000

APPENDIX 2

DCP1 (100 Tap) tap position to Data Byte translation algorithm example. (Example 1)

```
unsigned DCP1_TAP_Position(int tap_pos)
{
     int block;
     int i;
     int offset;
     int wcr_val;

     offset= 0;
     block = tap_pos / 25;

     if (block < 0) return ((unsigned)0);

     else if (block <= 3)
     {    switch(block)
          {    case (0): return ((unsigned)tap_pos) ;

               case (1):
               {
                    wcr_val = 56;
                    offset = tap_pos - 25;
                    for (i=0; i<= offset; i++) wcr_val-- ;
                    return ((unsigned)++wcr_val);
               }

               case (2):
               {
                    wcr_val = 64;
                    offset = tap_pos - 50;
                    for (i=0; i<= offset; i++) wcr_val++ ;
                    return ((unsigned)--wcr_val);
               }

               case (3):
               {
                    wcr_val = 120;
                    offset = tap_pos - 75;
                    for (i=0; i<= offset; i++) wcr_val-- ;
                    return ((unsigned)++wcr_val);
               }
          }
     }
     return((unsigned)01100000);
}
```

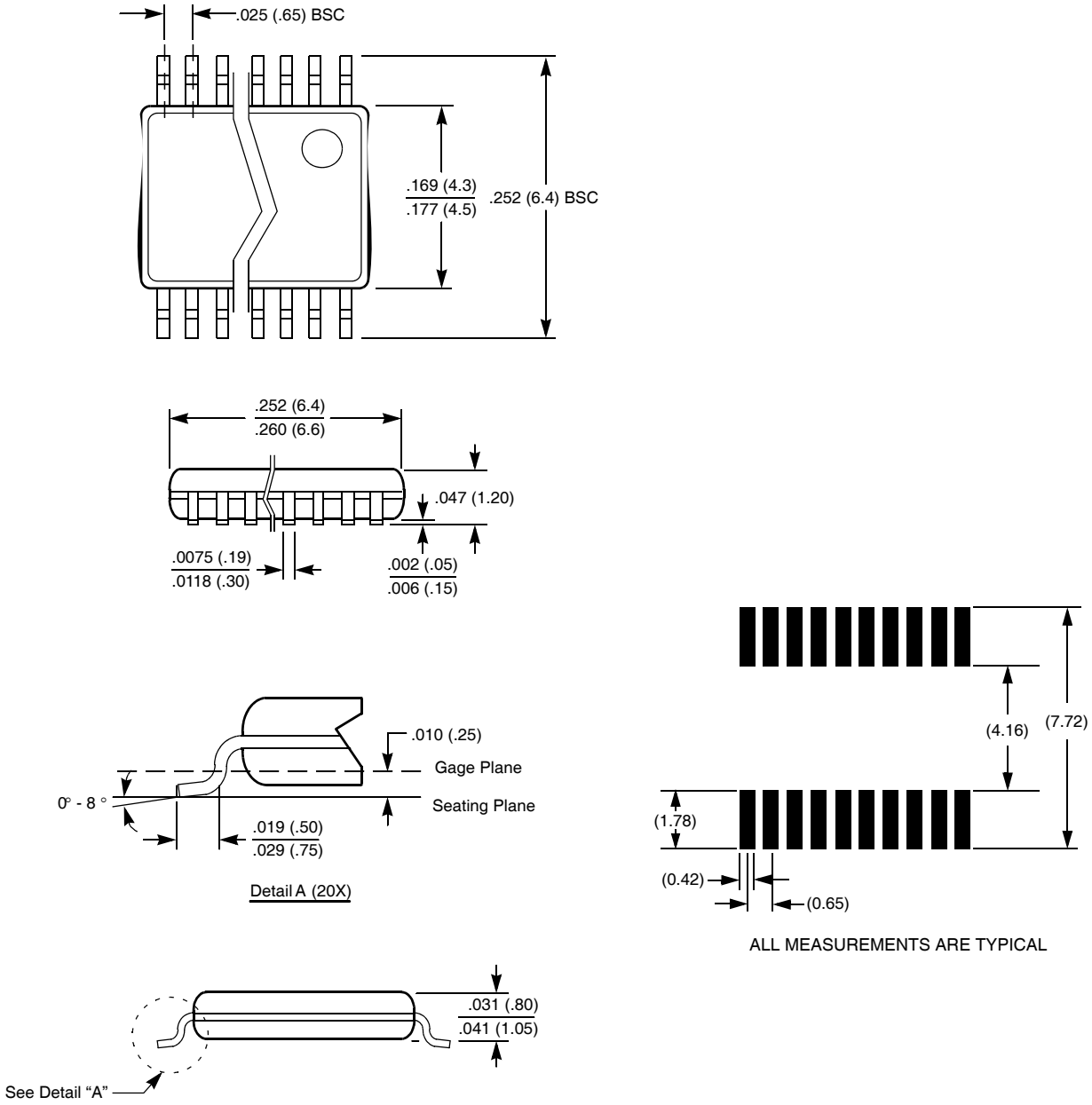
APPENDIX 2

DCP1 (100 Tap) tap position to Data Byte translation algorithm example. (Example 2)

```
unsigned DCP100_TAP_Position(int tap_pos)
{
/* optional range checking
*/ if (tap_pos < 0) return ((unsigned)0);           /* set to min val */
   else if (tap_pos >99) return ((unsigned) 96);    /* set to max val */

/* 100 Tap DCP encoding formula */
if (tap_pos > 74)
   return ((unsigned) (195 - tap_pos));
   else if (tap_pos > 49)
      return ((unsigned) (14 + tap_pos));
      else if (tap_pos > 24)
         return ((unsigned) (81 - tap_pos));
         else return (tap_pos);
}
```

20-LEAD PLASTIC, TSSOP PACKAGE TYPE V



NOTE: ALL DIMENSIONS IN INCHES (IN PARENTHESES IN MILLIMETERS)