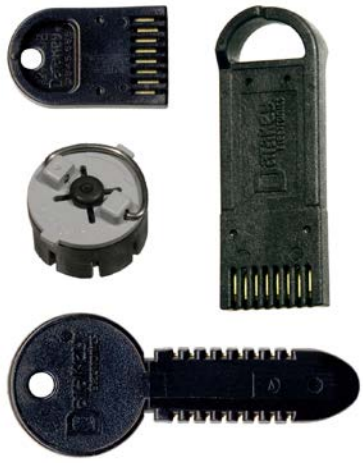
Datakey®

The power of memory. Secured.

I²C
Interface Specification



223-0017-005 REV E 3/14

Table of contents

Notices and other considerations

Important notices

- Datakey guarantees the quality of its devices by testing each device before shipment. However, installing and using Datakey products is the responsibility of the purchaser and is in no way guaranteed by Datakey
- Timing data, electrical characteristics, and signal descriptions are based on a compilation of several approved manufacturers' specifications. Datakey reviews the specifications of all approved vendors, and then "de-rates" the specifications as needed to ensure that all devices meet our published specifications regardless of the vendor used. Customers must design to our published specifications to ensure that all devices operate correctly within an application. Designing to a particular vendor's specifications is not recommended.
- **Design Recommendation:** It is recommended that all new key/token implementations be designed to operate with power supplies in the range of 2.7 to 3.6 volts. Although there is no immediate or certain future difficulties in the procurement of memory devices that operate with V_{cc} in the 4.5 to 5.5 volt range, it is possible the future availability of such memories may be impacted as semiconductor manufacturers continue to shrink their die geometries. Please contact the factory if you have any questions pertaining to this with your current or legacy design.
- While the information in this specification has been carefully reviewed, Datakey assumes no liability for any errors or omissions in this specification. Additionally, Datakey reserves the right to make changes to any part of the information in this specification or the products described herein without further notice.
- No part of this specification may be photocopied, reproduced, or translated to another language without the written consent of Datakey.

Other considerations

- Although portable key/tokens are designed to withstand harsh environments, many of the conditions that prevent them from working properly in such environments are best addressed through properly designed system interface circuits.
- Datakey tests all keys/tokens during the manufacturing process. In some cases after the test, data written to a key/token remain. Users should not rely on this data as a means of identifying keys/tokens.

Introduction

General description

Datakey portable memory keys/tokens contain electrically erasable programmable memory (EEPROM) accessed through a serial bus interface, using the Microwire, I²C, or SPI bus protocol. Each protocol controls input and output pins of the device through separate serial interface formats.

Portable memory device uses

Portable memory devices add functional versatility to many applications. They personalize equipment operations and transfer data in the following applications:

- Access control devices
- Instrument calibration equipment
- Fuel dispensers
- Medical treatment systems

Memory device design criteria

Portable memory applications require memory devices that can survive outside traditional environments, while maintaining data integrity when inserted and removed from the hosts powering them. Therefore, all portable memory devices must comply with the following basic design criteria:

- Resist dirt and other contaminates
- Transfer data reliably
- Tolerate electrostatic discharge
- Retain data when power is removed
- Retain data when exposed to certain environmental hazards

Manufacturers' design responsibility

Portable memory device manufacturers must address the above basic design criteria because they must develop memory devices capable of surviving in harsh environments. When a memory device is integrated into a larger system, the following design considerations become important:

- How to dissipate electrostatic discharge (ESD)
- How to maintain device data integrity
- How to prevent host system disruptions when inserting and removing a key/token

Continued on next page

Introduction, *continued*

Portable memory devices from Datakey

Datakey designs and manufactures portable, rugged keys/tokens containing non-volatile memory. Since 1976 our tough, reliable, and re-programmable keys, tokens, receptacles, and systems have solved data transport and access control problems in the most extreme environments.

Our I²C keys/tokens contain serial EEPROMs accessed through a simple "two-wire" serial interface. Simple instructions control data transfers to and from the I²C memory. This interface specification describes those instructions.

Integrity is ensured by methods described in this specification.

Contents of the specification

The remaining pages in this specification discuss I²C design criteria for portable-memory devices along with how to handle these devices in typical applications.

Functional description

Keys/Tokens

Figure 1 shows examples of serial EEPROM devices available from Datakey. Each type of key/token easily mates to a custom receptacle, which provides access to I²C communication, power, and ground signals.

ISK IST ISX ISP

Figure 1: I²C Memory Devices Available from Datakey

Signals

Communication between the microcontroller and devices on an I²C bus uses two signals:

- Serial Clock (SCL)
- Serial Data/Address (SDA)

These signals, along with the SIZE, V_{CC} *(supply voltage)* and ground signals are present on all keys/tokens.

Table 1 presents KC4210 and SR4210 receptacles signal/pin acronyms and descriptions.

Table 1: KC4210 and SR4210 Signals Acronyms and Descriptions

Signal Acronym	Signal Description
SIZE	Size *(Addressing mode)*
SDA	Serial Data/Address
SCL	Serial Clock
V_{CC}	Supply Voltage
V_{SS}	Ground

Continued on next page

***Functional description,** continued*

SIZE pin

The SIZE pin indicates if a key/token is greater than 16 kilobits in size.

- The SIZE pin on keys/tokens, 16 kbits or less in size is open *(not connected).*
- The SIZE pin on keys/tokens greater than 16 kbits is internally connected to ground. The host application can use a simple hardware detection circuit to read the SIZE pin and adjust the addressing protocol.
- Keys/tokens with memory sizes greater than 16 kbits require two address bytes following the control byte in an I^2C transaction.
- Keys/tokens with memory sizes less than or equal to 16 kbits only require one address byte following the control byte. Bits in the control byte are used as the most significant bits of a memory location.

For more information about addressing, see the Operating Features and the Instructions sections of this specification.

Serial data/address (SDA)

SDA is a bidirectional signal. It enables data and address information transfers between the master *(host)* and the memory device. Address and data information is valid on the SDA signal when the clock (SCL) signal is high. For normal operation, signal level changes only occur when the SCL signal is low. Signal level changes on the SDA while the SCL is high indicate a start or stop condition. *(See Start/Stop conditions in the Operating Features section of this specification).*

The SDA pin is an open-drain terminal, requiring a pull-up resistor to Vcc for proper operation. Typical values for pull-up resistors are 10kΩ for operation at 100 kHz, and 2kΩ for operation at 400 kHz.

Serial clock (SCL)

The SCL signal synchronizes the communication between the master device and the memory chip. The master or host hardware controls the SCL signal.

Supply voltage (V_{CC})

Datakey I^2C keys/tokens will operate throughout a V_{CC} range from 2.7 to 5.5 volts. **The supply voltage must be controlled so that keys/tokens are not inserted into live receptacles. See section entitled "Memory device power and signal control."**

Ground (Vss)

The ground signal and the system ground signal are common.

Operating features

Master/slave operation

All Datakey I²C keys/tokens are compatible with the I²C bus standard. All keys/tokens operate as slave devices on the I²C bus; they will act as a receiver or transmitter depending on data direction. A master device controls the serial clock (SCL) line.

Bus protocol

All Datakey I²C keys/tokens function as serial EEPROM devices on the I²C bus. Data transfers on an I²C bus must begin with an idle bus condition *(SCL and SDA lines high).* All I²C bus transactions begin with a start condition from the master device. A control byte is the first byte sent by the master following the start condition. The control byte specifies that the bus transaction be for serial EEPROM memory, the device or memory address, and the type of operation *(read or write).* Additional bytes are sent or received, depending on the type of instruction or transaction.

The receiving device transmits an acknowledge bit between each byte. It is the responsibility of the master device to generate clock pulses for the acknowledge bit. An I²C bus transaction ends when a stop condition appears on the bus.

Data on the SDA line must be stable when the SCL line is high and can change when the SCL line is low. Any change in the SDA line while the SCL line is high will be interpreted as a start or stop condition.

Start/stop conditions

Start Condition: A high-to-low transition of the SDA line while the SCL is high creates a start condition.

Stop Condition: A low-to-high transition of the SDA line while the SCL is high creates a stop condition.

Figure 2 shows start and stop condition timing.

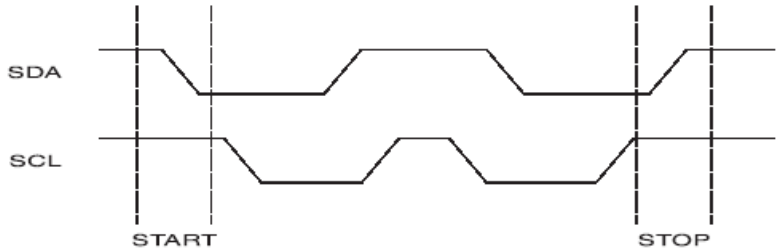


Figure 2: Start and Stop Condition Timing

Continued on next page

Operating features, *continued*

Acknowledge bit

All control bytes, addresses, and data transmit to the memory device as 8-bit bytes. The key/token will transmit an acknowledge bit *(logic "0")* on the ninth clock cycle to indicate receiving the transmitted byte. Similarly, when transmitting data from the memory device *(reading*), the master will provide the logic "0" acknowledge bit on the ninth clock cycle. In either case, the master device is responsible for generating the clock signal. Figure 3 shows the acknowledge-bit timing.

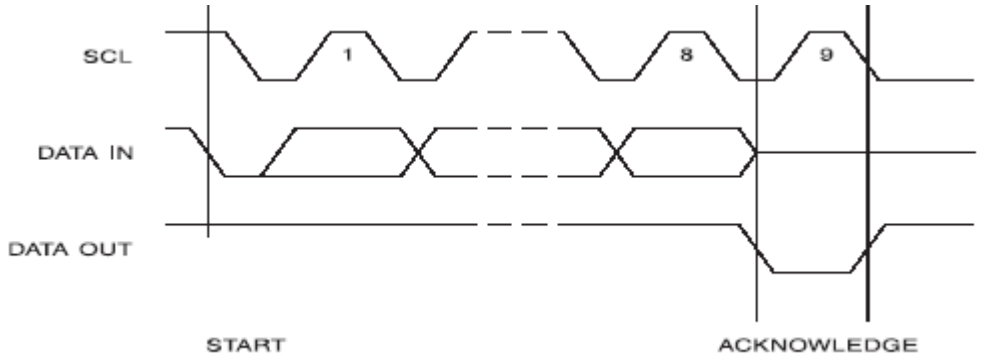


Figure 3: Acknowledge-Bit Timing

Device addressing

The control byte specifies hardware addressing on the I²C bus. It is the first byte sent following a start condition. The most significant four bits *(bits "7" through "4")* of the control byte are 1, 0, 1, 0, respectively, for all serial EEPROM memory devices that are compatible with the I²C bus. Bits "3" through "1" specify the individual device address or internal memory addressing, depending on the size of the key/token. The last bit *(bit "0")* specifies whether the operation is a read *(high)* or write *(low)* transaction.

Continued on next page

Operating features, *continued*

Cascading

Some keys/tokens support cascading of similar EEPROM memory devices on the same I²C bus. "Cascading" refers to permitting multiple serial EEPROM devices to reside on the same I²C bus. It requires the memory device to utilize hardwired device addresses.

Not all keys/tokens from Datakey support cascading. All standard keys/tokens are hardwired for address "0" if cascading is a feature. If cascading is supported, bits "3" through "1" of the control byte specify the address of the individual EEPROM on the bus. Other serial EEPROM devices on the bus must be wired for other addresses.[1]

In some cases, keys/tokens do not support cascading. Bits "3" through "1" of the control byte can be used to provide the higher address bits of the internal device memory.

Table 2 indicates the availability of cascading for each model of I²C keys/tokens from Datakey.

Note: All standard keys/tokens from Datakey are hardwired to address "0."

Page buffer

The page buffer allows multiple-byte writes within one EEPROM write cycle. Page buffer size varies with the total memory size of the key/token. Table 2 shows the page buffer size in bytes for each I²C key/token from Datakey.

Continued on next page

[1] Contact Datakey for custom key/token addressing.

***Operating features,** continued*

I²C interface information

Table 2 shows the Key interface information for each I²C key/token from Datakey.

For specific information about read and write operations, see the Instructions section of this specification.

Table 2: Key Interface Information

Key/token	Size (bytes)	Address Range	Address Bytes	Page Buffer	Cascadable
ISK1000 IST1000	128	0 ~ 0x7F	1	8	No
ISK4000 IST4000 ISP4000	512	0 ~ 0x1FF	1 + 1 bit	16	No
ISK16000 IST16000	2,048	0 ~ 0x7FF	1 + 3 bits	16	No
ISK64K IST64K	8,096	0 ~ 0x1FFF	2	32	Yes, 8
ISK256K IST256K	32,768	0 ~ 0x7FFF	2	64	Yes, 4
ISX512K	65,536	0 ~ 0x7FFF *(block 1)* 0 ~ 0x7FFF *(block 2)*	2	64	No

Notes:

- The address bytes column specifies the number of address bytes that follow the control byte used to specify the internal address of the memory device. The number of bits used in the control word to specify an internal memory address is relevant to keys or tokens of 16 kilobits or less in size.
- In the Cascadeable column, the numbers represent the EEPROM devices that may reside on the bus.
- The ISX512K is addressed as two 32K-byte blocks. Bit position “1” of the control byte specifies which 32K block is being addressed.
- It is recommended that all new key/token implementations not rely on specified page size for achieving a wrap-around effect for the effective memory address. Although Datakey has no intention to deviate from the listed specification, some semiconductor manufacturers offer devices with page sizes that differ from those published here. We feel a good engineering practice would be to not rely on the listed value in the event availability becomes an issue in the future. The page size in our memory products will be at least as large as what is specified here.

Instructions

Byte write

Following the start condition from the master device, the memory device receives a control byte. The control byte consists of the following bits:

- Device code *(1 0 1 0 for all serial EEPROM products)*
- Device or memory address *(depending on product)*
- R/W bit *(logic- level "0" for a write operation)*

The control byte configuration is as follows:

1	0	1	0	P2	P1	P0	R/W

Note: The ISX512K Token is organized as two separate 32k-byte blocks. P0 of the control byte specifies which 32K block is being addressed. This applies to all ISX512K transactions.

The acknowledge bit is transmitted by the key/token on the ninth clock cycle. Depending on memory size, one or two address bytes will follow the control byte. An acknowledge bit is sent after each 8-bit transmission. The data byte for the addressed memory location is transmitted following the address. The master device must terminate the write sequence by generating a stop condition. This will start an internal write cycle to the non-volatile memory. During an internal write cycle, the key/token will not respond to any commands and will not issue an acknowledge bit.

Figures 4 and 5 show one- and two-byte address write operation timing for one-and two-byte addressing keys/tokens.

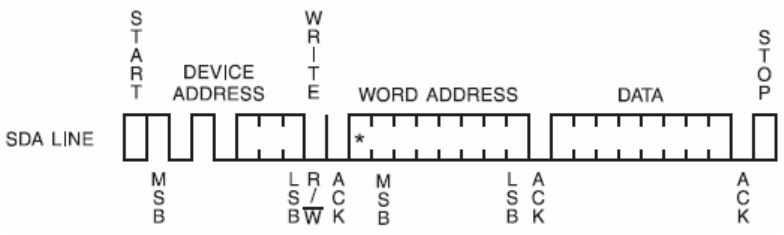


Figure 4: One-Byte Address, Single-Byte Write Sequence Timing

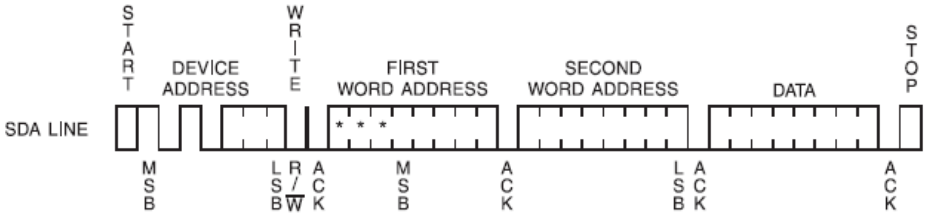


Figure 5: Two-Byte Address, Single-Byte Write Operation Timing

Continued on next page

***Instructions,** continued*

Page write

A page write and a byte write start in a similar manner. However, with page writes instead of generating a stop condition after the first data byte, the host can transmit additional bytes up to the size of the page buffer for the key/token. The size of the page buffer depends on key/token memory size *(see Table 2)*. The key/token transmits an acknowledge bit after each data byte. A stop condition terminates the write sequence, and initiates the internal write cycle. Again, the key/token will not issue acknowledge bits during the internal write cycle.

The page buffer operates on address boundaries equal to the size of the buffer. For example, a 32-byte page buffer will only increment the lower five address bits automatically when writing to the buffer. Data written beyond the upper limit of the page boundary will rollover and is stored at the beginning address of the page buffer. Similarly, if the data bytes written to the page exceed the page buffer size, the previous data will be overwritten.

Figures 6 and 7 show one- and two-byte address page-write operation timing for one- and two-byte addressing keys/tokens.

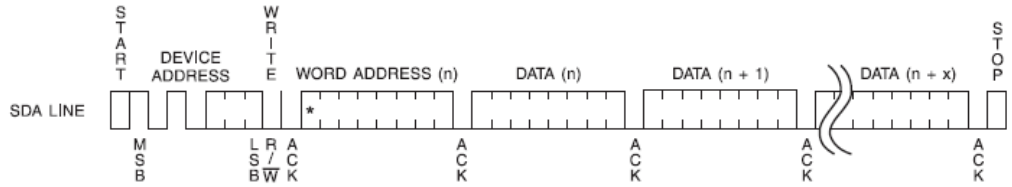


Figure 6: One-Byte Address, Page-Write Operation Timing

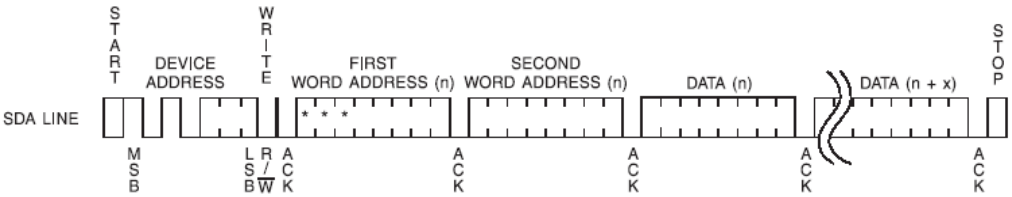


Figure 7: Two-Byte Address, Page-Write Operation Timing

Continued on next page

Instructions, *continued*

Acknowledge polling

Acknowledge polling can determine when the internal-write cycle of the EEPROM is complete. Immediately after the stop condition terminates the byte or page-write operation, a write cycle starts automatically. During the write cycle, the memory device will not respond with a zero-level acknowledge bit.

To test for write-cycle completion, the master will issue a start condition and control byte, and then look for the acknowledge bit from the key/token. If the write cycle is complete, the memory will respond with a zero-level acknowledge bit. If the write cycle is in process, a high-level signal will be read on the ninth clock cycle.

Read operation

Read cycles

Write and read operations initiate in much the same manner. The only difference is the read/write bit in the control byte is set to "1."

There are three basic read operations:

- Current-address read
- Random read
- Sequential Read

Current-address read operation

Provided power has not been removed from the key/token, an internal address counter is maintained that points to the memory location following the last address accessed during a read or write operation. The address pointer rolls over to the beginning of memory if the last address accessed was the last address in the memory range.

The current address can be read by issuing a start condition followed by the control byte with the read/write bit set to "1." The memory device will issue an acknowledge bit on the ninth clock cycle and output the data at the current address on subsequent clock cycles. The master terminates the current address read operation by issuing a "no acknowledge" bit "1" followed by a stop condition. The current-address read operation is the same for single and two-byte addressing keys/tokens.

Figure 8 shows current-address read operation timing.

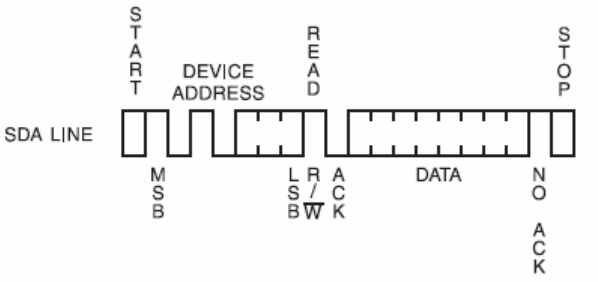


Figure 8: Current-Address Read Operation Timing

Continued on next page

***Read operation,** continued*

Random-address read operation

A random-address read operation requires the internal address pointer to be set with the desired memory location address. This is done by sending the address as part of a write operation. Following a start condition the control byte—including any extended address bits—is sent with the read/write bit set to "0" for a write operation. The memory device will respond with an acknowledge bit. The master device will send the one- or two-byte address.

Once the address is clocked into the device, the master will issue another start condition. The control byte is sent again with the read/write bit set to "1." The data at the address location will be clocked out following the acknowledge bit. An acknowledge bit will be issued by the memory device following each 8-bit address byte. The master terminates the random address read operation by issuing a "no acknowledge" bit "1" followed by a stop condition.

Figures 9 and 10 show the random-address read operation timing for one- and two-byte addressing keys/tokens.

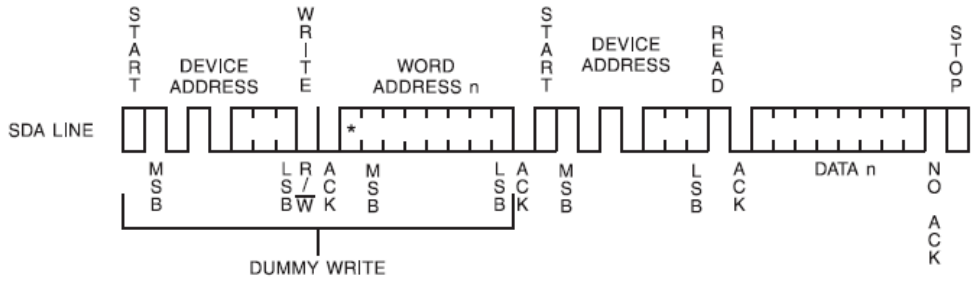


Figure 9: One-Byte Address, Random-Address Read Operation Timing

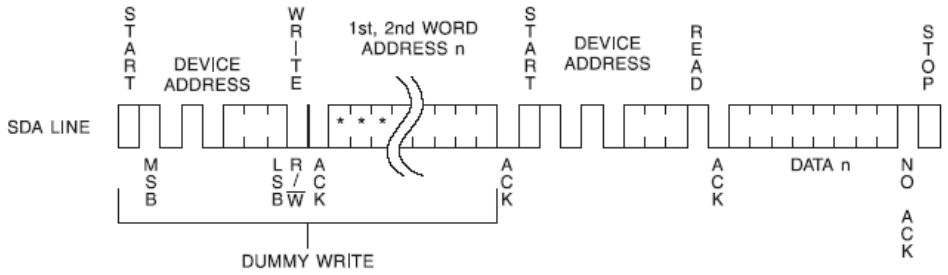


Figure 10: Two-Byte Address, Random-Address Read Operation Timing

Continued on next page

Read operation, *continued*

Sequential read operation

A sequential-read operation reads the entire memory contents with one instruction. The sequential read operation is initiated in the same manner as the random read or current-address read operation. Instead of responding with a no-acknowledge bit followed by a stop condition, the master will issue an acknowledge bit.

The internal address pointer will automatically increment and the data from the next location will clock out. The master can terminate the sequential-read operation after any byte read by sending a no-acknowledge bit followed by a stop condition.

Figure 11 shows the sequential-read operation, following the dummy-write operation timing.

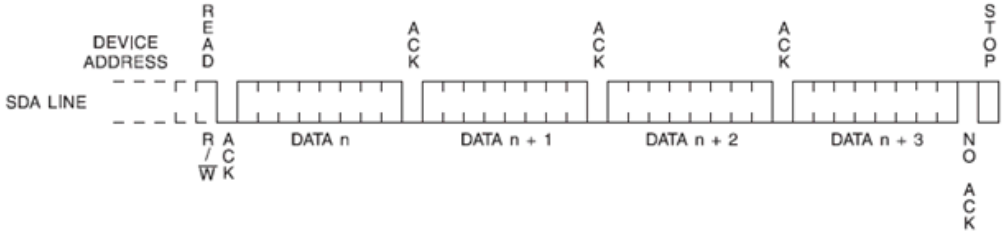


Figure 11: Sequential-Read Operation Timing

Panel/board-mount receptacle descriptions

Receptacles

The receptacles are used to interface the host system directly with specific serial data keys/tokens. The types are the KC4210, KC4210PCB, SR4210, and the SR4210PCB.

A Last On/First Off (LOFO) switch in the key/token receptacle enables the host system to determine when a key/token is present. Upon insertion of a key/token, the LOFO contact connects to ground. Conversely, when the key is removed, the LOFO contact is open. The LOFO contact allows system designers to detect the presence of a key/token, and protects the host bus by applying power only when a key/token is fully inserted into the receptacle.

KC4210 panel-mount receptacle

The KC4210 panel-mount design is for applications that require easy mounting in a front-panel configuration. To mount the receptacle, simply cut a one-inch square hole in the desired panel location and then snap the receptacle into place. Use a standard 10-pin connector cable *(5 x 2)* to connect the device to the host. Figure 12 is a picture of the receptacle.

Figure 12: KC4210 Panel-Mount Receptacle

Note: It is recommended that the total length of signal conductors, PC board traces, and ribbon cables not exceed eight inches.

Continued on next page

Panel/board-mount receptacle descriptions, *continued*

KC4210 orthographic drawing

Figure 13 shows the KC4210 panel-mount receptacle. Refer to spec sheet for dimensions.

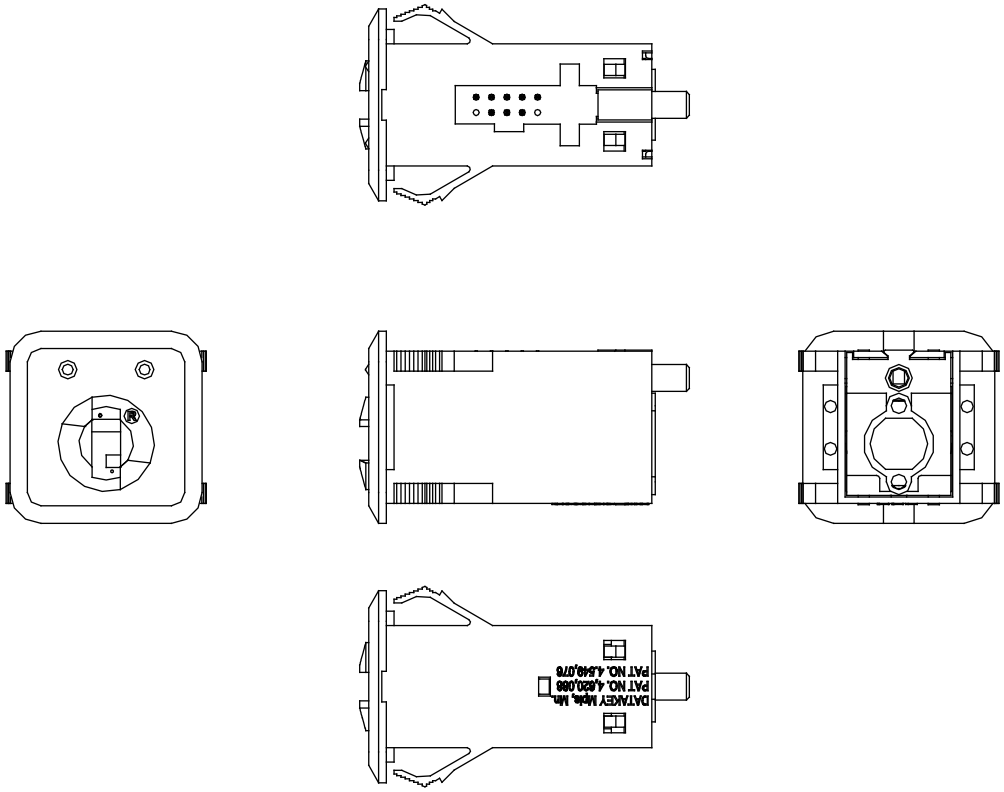


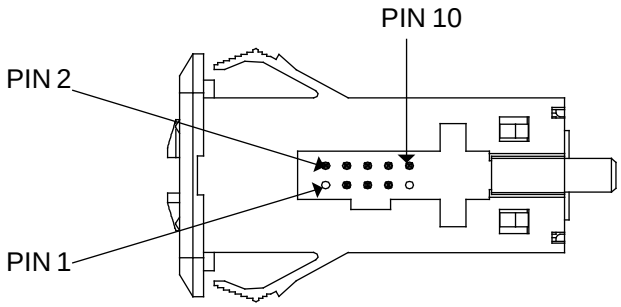
Figure 13: KC4210 Panel-Mount Receptacle Orthographic Drawing

Continued on next page

Panel/board-mount receptacle descriptions, *continued*

KC4210 pin outs

Figure 14 shows a KC4210 receptacle diagram and a description of its pin outs.



Receptacle
Bottom View

I²C	
Pin No.	**Description**
Pin 1	NC
Pin 2	Power (V_{CC})
Pin 3	Ground (V_{SS})
Pin 4	Size
Pin 5	NC
Pin 6	NC
Pin 7	Serial Clock (SCL)
Pin 8	Serial Data/Address (SDA)
Pin 9	NC
Pin 10	LOFO

Figure 14: KC4210 Receptacle and Pin Outs

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

KC4210PCB - mount receptacle

The KC4210PCB-mount receptacle design is for applications where the designer wants to mount the device directly onto a printed circuit board (PCB). In such applications, connect the PCB-mount receptacle to the host by soldering its leads onto a PCB. Figure 15 shows a picture of the receptacle.

Figure 15: KC4210PCB-Mount Receptacle

KC4210PCB orthographic drawing

Figure 16 shows the KC4210PCB-mount receptacle. Refer to spec sheet for dimensions.

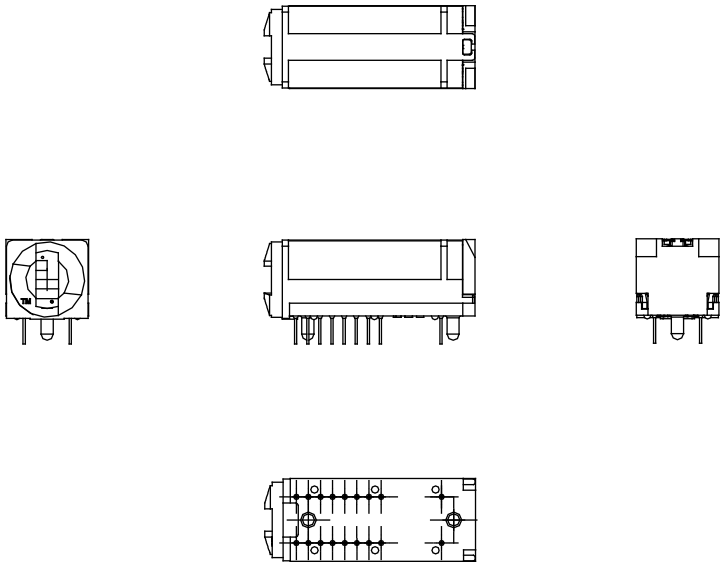


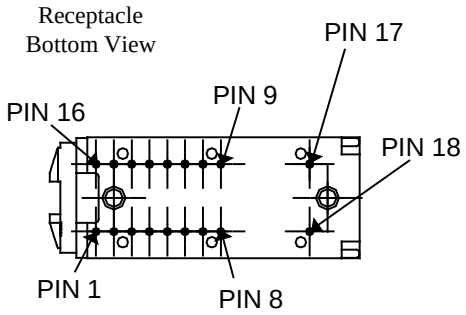
Figure 16: KC4210PCB-Mount Receptacle Orthographic Drawing

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

KC4210PCB pin outs

Figure 17 shows a receptacle diagram and a description of its pin outs.

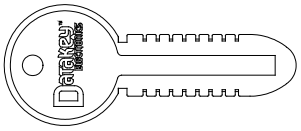


I²C	
Pin No.	**Description**
Pin 1	Size
Pin 2	Ground (V_{SS})
Pin 3	Power (V_{CC})
Pin 4	NC
Pin 5	Serial Data/Address (SDA)
Pin 6	NC
Pin 7	Serial Clock (SCL)
Pin 8	NC
Pin 9	NC
Pin 10	Serial Clock (SCL)
Pin 11	NC
Pin 12	Serial Data/Address (SDA)
Pin 13	NC
Pin 14	Power (V_{CC})
Pin 15	Ground (V_{SS})
Pin 16	Size
Pin 17	LOFO
Pin 18	LOFO

Figure 17: KC4210PCB Receptacle and Pin Outs

KC4210 key style

The KC4210 panel- and board-mount receptacles accept the ISK style key, shown in Figure 18.



ISK

Figure 18: ISK Key for the KC4210 Receptacle

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

SR4210 panel-mount receptacle

The SR4210 panel-mount version is designed for applications that require easy mounting in a front-panel configuration. To mount the SR4210 panel-mount receptacle, simply cut a hole based on the dimensions shown on the SR4210 spec sheet in the desired panel location and then slip it into place. A standard 10-pin connector cable *(5 x 2)* is used to connect the device to the host. A Last On/First Off (LOFO) switch in the receptacle enables the host system to determine when a token is present.

Figure 19: SR4210 Panel-Mount Receptacle and Clip

Note: It is recommended that the total length of signal conductors, PC board traces, and ribbon cables not exceed eight inches.

SR4210 orthographic drawing

Figure 20 shows the SR4210 panel-mount receptacle. Refer to spec sheet for dimensions.

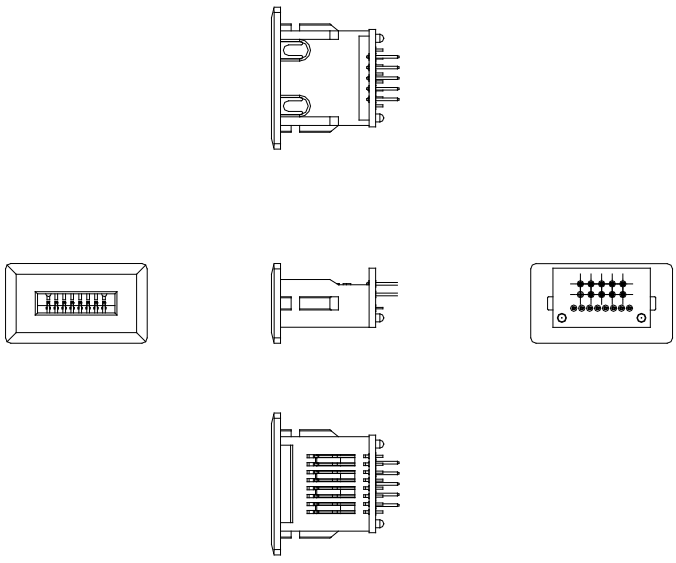


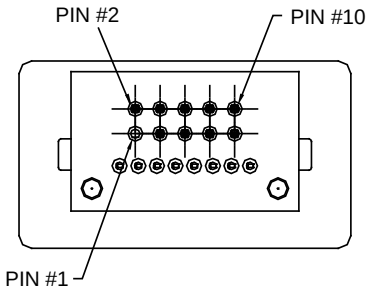
Figure 20: SR4210 Panel-Mount Receptacle Dimensions

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

SR4210 pin outs

Figure 21 shows a receptacle diagram and a description of its pin outs.



I²C	
Pin	**Description**
Pin 1	NC
Pin 2	Power (V_{CC})
Pin 3	Ground (V_{SS})
Pin 4	Size
Pin 5	NC
Pin 6	NC
Pin 7	Serial Clock (SCL)
Pin 8	Serial Data/Address (SDA)
Pin 9	NC
Pin 10	LOFO

Figure 21: SR4210 Receptacle and Token Diagrams and Pin Outs

SR42XXPCB mount receptacles

The information on the SR4210PCB below also applies to the SR4220, SR4230 board-mount receptacles. Dimensions can be found in the corresponding spec sheets. Contact the factory for information on SMT options.

SR4210PCB-mount receptacle

The SR4210PCB-mount receptacle design is for applications where the designer wants to mount the receptacle directly onto a PCB. In such applications, mount the receptacle to the host by soldering its leads onto a PCB. Figure 22 shows a picture of the receptacle.

Figure 22: SR4210PCB-Mount Receptacle

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

SR4210PCB orthographic drawing

Figure 23 shows the SR4210PCB board-mount receptacle. Refer to spec sheet for dimensions.

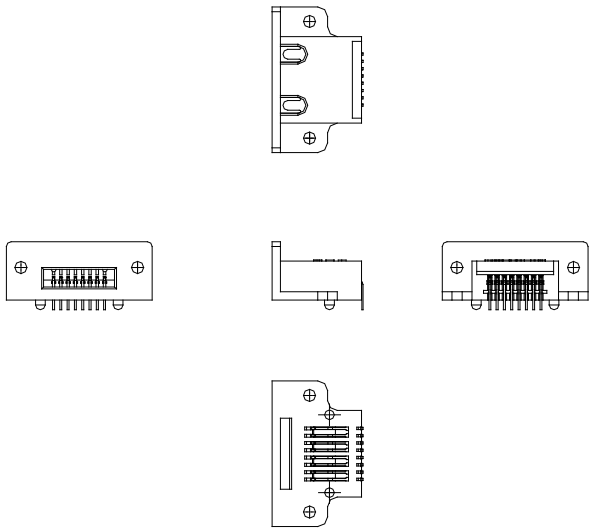
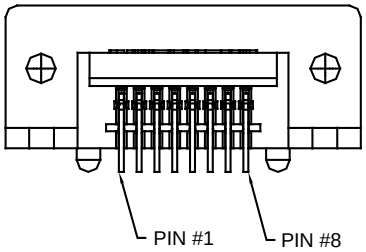


Figure 23: SR4210PCB-Mount Receptacle Orthographic Drawing

SR4210PCB pin outs

Figure 24 shows a SR4210PCB-mount receptacle and a description of its pin outs.



I²C	
Pin	**Description**
Pin 1	LOFO
Pin 2	Power (V_{CC})
Pin 3	Size
Pin 4	Serial Clock (SCL)
Pin 5	NC
Pin 6	Serial Data/Address (SDA)
Pin 7	Ground (V_{SS})
Pin 8	LOFO

Figure 24: SR4210PCB-Mount Receptacle and Token Diagrams and Pin Outs

Continued on next page

Panel/board-mount receptacle descriptions, *continued*

Slim token styles & pin outs

The SR4210 panel- and PCB-mount receptacles accept the IST- and ISX-token style with memory sizes from 1Kb to 512Kb. Figure 25 shows the token styles and pinout. **Note:** the token has redundant contacts, therefore the pinout shown applies to both views of the token.

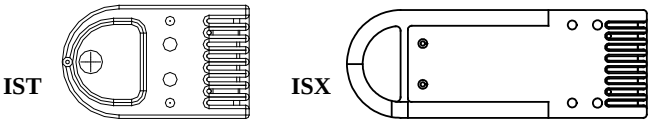


Figure 25: IST Token and ISX Extended Token Styles for the SR4210 Receptacle and Pin Outs

KSD receptacle

The KSD receptacle accepts ISP plugs. It can be used in board- and panel-mount applications. Figure 26 and 27 show pictures of the KSD receptacle and ISP plug.

Figure 26: KSD Receptacle

Figure 27: ISP Plug

Continued on next page

***Panel/board-mount receptacle descriptions,** continued*

KSD receptacle orthographic drawing

Figure 28 shows the KSD board-mount receptacle. Refer to spec sheet for dimensions.

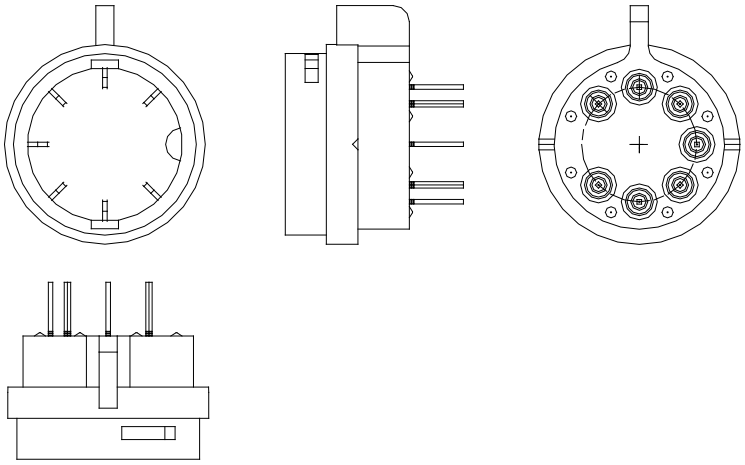
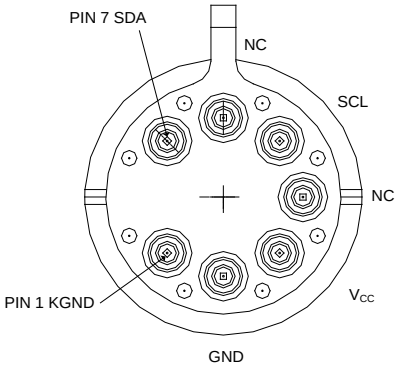


Figure 28: KSD Receptacle Orthographic Drawing

KSD/ISP pin outs

Figure 29 shows a bottom view of a KSD receptacle diagram and a description of its pin outs when used with an ISP plug.



I²C	
Pin	**Description**
Pin 1	KGND*
Pin 2	Ground
Pin 3	Power (VCC)
Pin 4	NC
Pin 5	Serial Clock (SCL)
Pin 6	NC
Pin 7	Serial Data/Address (SDA)

***Note:** KGND is connected to GND inside the plug. When a plug is inserted into the receptacle, KGND is pulled "low". The host monitors the KGND signal to determine the presence of the Plug..

Figure 29: KSD/ISP Receptacle Pin-Out Positions and Description

Electrical interface

Electrical characteristics

Table 3 shows absolute maximum values and temperatures for serial I²C keys/tokens; Tables 4 and 5 show DC and AC characteristics for I²C keys/tokens.

Caution

The conditions shown in Table 3 are stress ratings only. Stressing I²C keys/tokens beyond the limits specified in Tables 4 and 5 could compromise performance or cause permanent damage to keys or tokens.

Table 3: Absolute Maximum Values and Temperatures

Symbols	Parameters	Min/Max	Units
V_{CC}	Supply voltage	6.25	V
$V_{IN/OUT}$	All pins w.r.t. Ground	-0.5 to 6.5	V
T_{STG}	Storage temperature	-65 to 150	°C
T_{BIAS}	Operating temperature	-40 to 85	°C

Notes:

Keys/tokens manufactured before 2004 have an operating temperature range of 0 to 70°C.

Design Recommendation: It is recommended that all new key/token implementations be designed to operate with power supplies in the range of 2.7 to 3.6 volts. Although there is no immediate or certain future difficulties in the procurement of memory devices that operate with V_{cc} in the 4.5 to 5.5 volt range, it is possible the future availability of such memories may be impacted as semiconductor manufacturers continue to shrink their die geometries. Please contact the factory if you have any questions pertaining to this with your current or legacy design.

Continued on next page

Electrical interface, *continued*

Table 4: DC Electrical Characteristics

Symbols	Parameters	Min	Max	Units	Conditions
V_{CC}	Supply voltage	2.7	5.5	V	
V_{IH}	High-level voltage input	0.7xVcc	Vcc + 0.5	V	
V_{IL}	Low-level voltage input	-0.6	0.3xVcc	V	
V_{OL}	Low-level voltage output		0.4	V	I_{OL} = 2.1 mA, V_{CC} = 2.5 V
I_{LI}	Input-leakage current		± 10.0	µA	V_{IN} = 0.1V to V_{CC}
I_{LO}	Output-leakage current		± 10.0	µA	V_{OUT} = 0.1V to V_{CC}
C_{IN}	Input-pin capacitance		10.0	pF	V_{CC} = 5.0, T_a = 25°C, F_{SCL} = 1 MHz
C_{OUT}	Output-pin capacitance		10.0	pF	V_{CC} = 5.0, T_a = 25°C, F_{SCL} = 1 MHz
$I_{CC\ read}$	Supply-current, read		2.0	mA	V_{CC} = 5.5, SCL = 400 kHz
$I_{CC\ write}$	Supply-current, write		5.0	mA	V_{CC} = 5.5, SCL = 400 kHz
I_{CCS}	Supply-current standby		6.0	µA	SDA = SCL = V_{CC}

Continued on next page

Electrical interface, *continued*

Table 5: AC Electrical Characteristics

Symbols	Parameters	Min	Max	Units
F_{SCL}	Clock Frequency		400	kHz
t_{HIGH}	Clock High time	600		ns
t_{LOW}	Clock Low time	1300		ns
t_R	SDA and SCL Rise time		300	ns
t_F	SDA and SCL Fall time		300	ns
$t_{HD:STA}$	Start Condition Hold time	600		ns
$t_{SU:STA}$	Start Condition Setup time	600		ns
$t_{HD:DAT}$	Data Input Hold time	0		ns
$t_{SU:DAT}$	Data Input Setup time	100		ns
$t_{SU:STO}$	Stop Condition Setup time	600		ns
t_{AA}	SCL Low to valid SDA out		900	ns
t_{DH}	Data out Hold time	50		ns
t_{BUF}	Bus Free time between transmissions	1300		ns
T_I	Noise Suppression time		50	ns
t_{WC}	Write Cycle time *(byte or page)*		10	ms
t_{PUR}	Power up to read operation time		1	ms
t_{PUW}	Power up to write operation time		1	ms

Timing diagrams

I²C Bus Timing

Figure 30 shows the timing for a microcontroller and devices on an I²C bus.

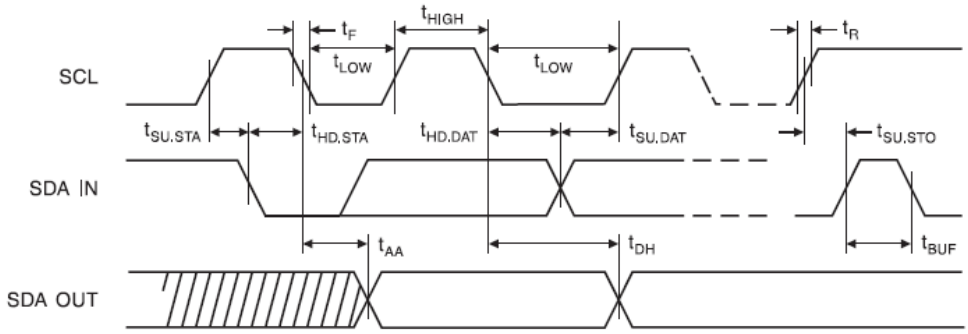


Figure 30: I²C Bus Timing

Electrostatic discharge (ESD)

Circuit component damage

A buildup of electrostatic charge gradients across the surface of a memory device can produce voltages that could damage circuit components. To prevent such damage, Datakey portable memory devices integrate materials, circuits, and mechanical barriers that help to ensure uniform voltage across the circuit.

Electrostatic-charge voltage levels

Any system that uses memory devices must also provide a means of dissipating electrostatic charges. By simply holding a portable memory device in your hand, it is possible to build up an electrostatic charge across the surface of the memory device, up to 20KV relative to ground. This would be equivalent to connecting a circuit of several hundred picofarads of capacitance with a low-series resistance to the memory device.

Electrostatic-charge dissipation

When inserting a portable memory device into a system that is grounded or at some other potential, the built-up charge from handling the memory device must be safely dissipated. This can be done by providing a path to ground for the charge. The path must be controlled to prevent large currents and high voltages from occurring on the memory device and in the receptacle. This can be done by putting a resistor in the circuit trace for the receptacle, and using over-voltage protection devices to direct the charge to system ground.

Memory device power and signal control

Poor contact concerns

When inserting a portable memory device into a receptacle, there can be poor contact between the memory device and the receptacle. There could be several possible causes:

- Dirty contact surfaces: To make enough electrical contact, the contact surfaces must be free of contaminants. This requires that the contacts be cleaned through a wiping action from the portable memory device.
- The contacts could bounce and require some time to settle. This could result in a series of random make and break conditions on any or all of the contacts.
- When a memory device is removed from a receptacle, the contacts do not always break evenly or cleanly.

Power concerns

When power is applied to a memory device when it is inserted or removed from a receptacle, random contact makes and breaks could cause significant problems for the memory device and the receptacle. Because the control and address signals are not controlled during those actions, undesirable logic combinations can occur, such as the following:

- Power and ground connections becoming unstable, causing further unpredictability.
- Fast power switching to a memory device can introduce noise into system power and ground distribution circuits, resulting in electrical damage to the memory device and subsequent data corruption.

Data-corruption prevention

The integrated circuits used in Datakey keys/tokens are designed to reduce the risk of data corruption during transient conditions. For example, keys/tokens require a Write Enable instruction before storing any data. Write instructions are also not permitted if the supply voltage is less than a prescribed value. As effective as these protection schemes might be, they do not always eliminate the potential problems with noise that can occur when power is applied to a circuit via a bouncing contact.

To avoid these problems, it is important to control the key/token's power and signal connections. This can be done by using detection circuits, which are discussed next.

Continued on next page

Memory device power and signal control, *continued*

Memory device detection circuit

The memory-device detection circuit detects when a memory device is present. Datakey receptacles use Last On/First Off (LOFO) contacts for this purpose. When inserting a memory device into the receptacle, the LOFO contacts make electrical connection only after all memory device contacts are closed. Similarly, the LOFO contacts break before any other contact is open.

Transistor switch circuit

When a receptacle detects a memory device for a certain minimum time, power can then be applied. This delay can be established by using a simple transistor switch circuit with the following characteristics:

- The power switch should have a low voltage drop when power is applied to the memory device. This will ensure that the voltage supplied to the key/token is within a safe and acceptable range.
- The circuit should apply power to any pull-up resistors connected to the key/token to prevent power from being applied unintentionally through the signal lines.
- The circuit should include a bleeder resistor to ensure that power is removed quickly when the switch is turned OFF.
- The switch should turn power ON fast enough to avoid causing problems in the key/token, and slow enough so that it does not introduce any significant noise into the system-reset circuit.

I²C read and write procedures

Procedures

Follow the procedures below when using a key/token in a receptacle with a power-switching circuit.

Read procedure

The procedure for reading data from a key/token is less critical than the sequence for writing data to that same key/token because the data are not subject to change. To read the data from a key/token, Datakey recommends the following procedure:

- Insert the key/token
- Detect the key/token using the LOFO contact
- Wait for contacts to settle *(verify presence of Key/token)*
- Apply power
- Wait for power to stabilize
- Test contact integrity
- Read data
- Remove power
- Remove the key/token

Continued on next page

I2C read and write procedures, *continued*

Write procedure

The write procedure must verify that the key/token is present throughout the write cycle, which will ensure that data is written to the key/token correctly. To write data to a key/token, Datakey recommends the following procedure:

- Insert the key/token
- Detect the key/token using the LOFO contact
- Wait for contacts to settle *(verify presence of Key/token)*
- Apply power
- Wait for power to stabilize
- Test contact integrity
- Write data
- Verify presence of key/token *(if not, indicate an error)*
- Verify the data written *(if applicable, indicate an error)*
- Remove power
- Remove the key/token

Long read/write operations

Large capacity memory systems should also be protected against key/token removal during long Read or Write operations. An activity light might be all that is needed for some applications. Other installations could require physical barriers or interlocks to ensure that the key/token being read or written to remains in place.

Addendum

IIK & IIT device interfacing

Introduction

The IIK and IIT devices provide 192 bytes of user programmable serial EEPROM memory. The IIK/IIT products also contain a unique embedded read-only serial number.

For customers that use the Keylink II or Slimlink II readers, version 2.04 firmware for these readers provides the correct access to the entire memory space and special features of the IIK and IIT products.

I²C bus communication

Communication between the microcontroller and devices on an I²C bus uses two signals: SCK and SDA. See Table 1

RST signal functionality

In the IIK and IIT devices, the RST signal replaces the size pin. This is normally the signal used to determine the memory size in our standard I²C products. However, the IIK and IIT products only come with one addressing scheme. Since there is only one method to address these keys and tokens, it is not necessary to read this pin to determine addressing requirements.

RST signal state for IIK or IIT device normal operation

Hold the RST signal low for normal operation of the IIK or IIT device. An internal pull-down resistor will hold this signal at a low level so it is acceptable to leave this signal unconnected on target boards.

The firmware modifications made in version 2.04 of the Keylink II and Slimlink II ensure that this signal remains low during normal read/write operations with these products.

Note: The RST signal may not be available on IIK and IIT products in future versions. Contact the factory for additional information.

Continued on next page

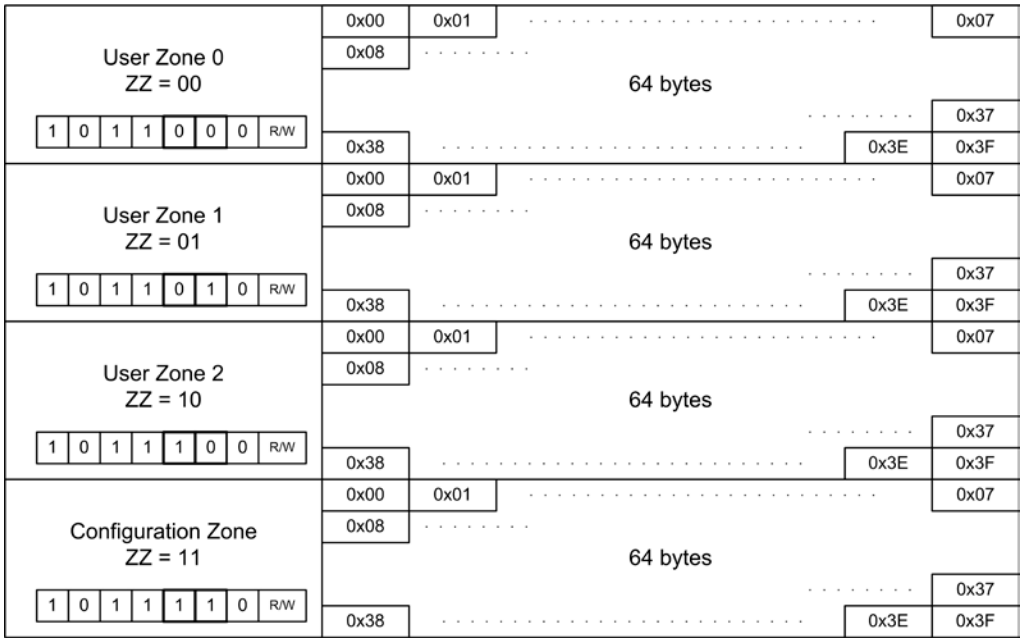
IIK & IIT device interfacing, *continued*

Memory Organization

The memory size in the IIK and IIT products is 2,048 bits organized as four zones of 64 bytes each:

- Zones 0, 1, and 2 are general purpose, read/write memory areas for application use.
- Zone 3 is a configuration zone that contains the DKE-specific fab code *(0xAE63)*, the CMC *code (0x0000 for general-purpose IIK and IIT devices)*, and the six byte serial number programmed into the key or token.

Table 5: Memory Map



Continued on next page

IIK & IIT device interfacing, *continued*

Configuration Zone

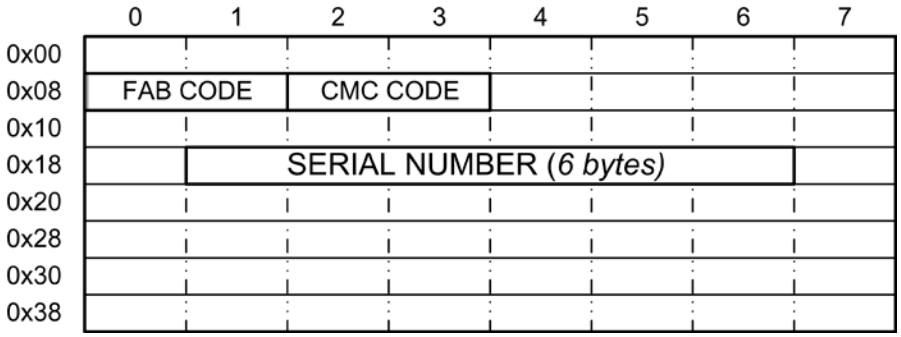
The configuration zone is only accessed under normal operation to read the fab code, CMC code, and serial number. The serial number is 6 bytes *(address 0x19 – 0x1E).*

The sequence to read the serial number is as follows:

Stage	Description
1.	Send the I2C the required start bit.
2.	Send the command byte for reading from zone 3 – 1 0 1 1 1 1 0 1 *(the last bit specifies a read operation).*
3.	Clock in an ACK from the device.
4.	Send the address byte for the address within the 64 byte zone – 0 0 0 1 1 0 0 1 *(serial number begins at 0x19).*
5.	Clock in an ACK from the device.
6.	Clock in the first byte of the serial number *(most significant byte).*
7.	Send an ACK to the device.
8.	Clock the next byte and continue to ACK for bytes 2, 3, 4, 5.
9.	Clock the last byte of the serial number *(address 0x1E)* then send a NACK and Stop to device.

Similarly, the fab code and CMC code may be read by initiating a Read command of four bytes as outlined above with an address of 0x08. Again the zone bits in the command byte will be 1 1.

Table 6: Configuration Code Structures



Continued on next page

***IIK & IIT device interfacing,** continued*

Read Command

Addresses within the IIK or IIT device are specified in the command *(or control)* byte and the following address byte.

- The general format for the read command byte for the IIK/IIT is 1 0 1 1 z z 0 1. The upper nibble *(1011)* is the device address.
- The next two bits (zz) specify the 64-byte zone that is to be addressed:
  - 00 = zone 0
  - 01 = zone 1
  - 10 = zone 2
  - 11 = zone 3
- Zone 3 is the configuration zone

Note: The Read Commands for an IIK or IIT device are not the same as standard I²C read commands that require a "dummy" write command to establish an address. Read commands specify the address within the command.

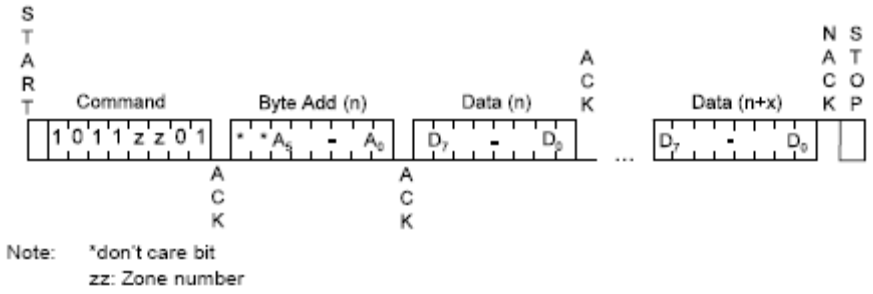


Figure 31: **IIK/IIT Read Command Structure**

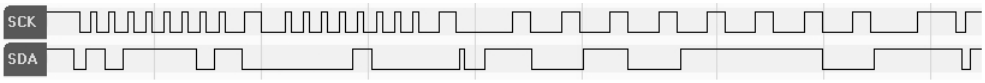


Figure 32: IIK/IIT Read Command *(1st Byte of FAB code = 0xAE)*

Continued on next page

IIK & IIT device interfacing, *continued*

Write Command

Addresses within the IIK or IIT product are specified in the command *(or control)* byte and the following address byte. The general format for the Write command byte for the IIK/IIT is 1 0 1 1 z z 0 0.

- The upper nibble (1011) is the device address.
- The next two bits (zz) specify the 64 byte zone that is to be addressed:
  - 00 = zone 0
  - 01 = zone 1
  - 10 = zone 2
  - 11 = zone 3
- Zone 3 is the configuration zone.

Write commands should be restricted to the first three zones of memory. The address within the 64-byte zone follows the command byte and the data to be written follows the address byte. There is an 8-byte page buffer for writing. All writes must be 8 bytes or less in length.

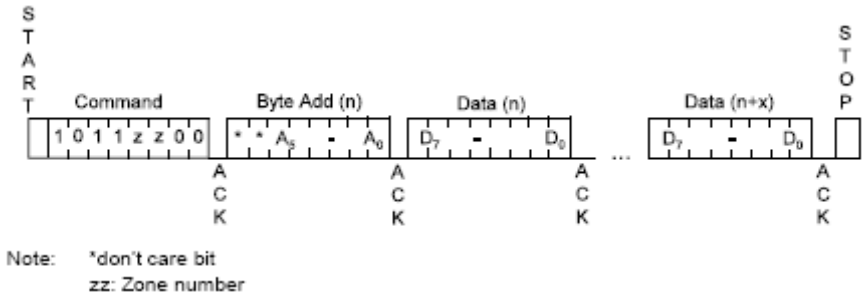


Figure 33: **IIK and IIT Devices Write Command**

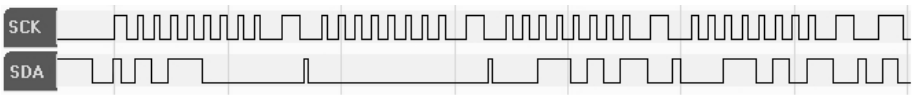


Figure 34: IIK/IIT Write Command *(Write 0x3535 to address 0x00)*

Acknowledgement

Atmel Corporation Timing diagrams are courtesy of the Atmel Corporation.

Revision History

Date	Revision	Description
12/08/05	A	The initial issue of the I²C interface specification.
3/15/07	B	Add Addendum (IIK and IIT Interfacing) and note regarding Power Supply Design Recommendations.
9/21/07	C	Add warning about not relying on page wrap-around, removed dimension info, updated pin out drawings, updated table 4 (I_{OL} and I_{CC}), and updated protection language, etc.
11/29/07	D	Add ISP Plug information and clarify SR4210PCB pin out to entire SR4000 receptacle family.
3/14/14	E	Updated Datakey logos.

The power of memory. Secured.

ATEK Access Technologies assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein.No licenses to patents or other intellectual property of ATEK Access Technologies are granted by the Company in connection with the sale of Datakey products, expressly or by implication.

223-0017-005 REV E 3/14