

Features

- Single 2.7V to 3.6V Supply
- RapidS Serial Interface: 66MHz Maximum Clock Frequency
 - SPI Compatible Modes 0 and 3
- User Configurable Page Size
 - 256-Bytes per Page
 - 264-Bytes per Page
 - Page Size Can Be Factory Pre-configured for 256-Bytes
- Page Program Operation
 - Intelligent Programming Operation
 - 1,024 Pages (256/264-Bytes/Page) Main Memory
- Flexible Erase Options
 - Page Erase (256-Bytes)
 - Block Erase (2-Kbytes)
 - Sector Erase (32-Kbytes)
 - Chip Erase (2-Mbits)
- One SRAM Data Buffer (256/264-Bytes)
- Continuous Read Capability through Entire Array
 - Ideal for Code Shadowing Applications
- Low-power Dissipation
 - 7mA Active Read Current Typical
 - 25µA Standby Current Typical
 - 15µA Deep Power-down Typical
- Hardware and Software Data Protection Features
 - Individual Sector
- Sector Lockdown for Secure Code and Data Storage
 - Individual Sector
- Security: 128-byte Security Register
 - 64-byte User Programmable Space
 - Unique 64-byte Device Identifier
- JEDEC Standard Manufacturer and Device ID Read
- 100,000 Program/Erase Cycles Per Page Minimum
- Data Retention – 20 Years
- Industrial Temperature Range
- Green (Pb/Halide-free/RoHS Compliant) Packaging Options

Description

The Atmel® AT45DB021D is a 2.7V, serial-interface Flash memory ideally suited for a wide variety of digital voice-, image-, program code- and data-storage applications. The Atmel AT45DB021D supports Atmel RapidS™ serial interface for applications requiring very high speed operations. RapidS serial interface is SPI compatible for frequencies up to 66MHz. Its 2-,162-,688-bits of memory are organized as 1,024-pages of 256-bytes or 264-bytes each. In addition to the main memory, the AT45DB021D also contains one SRAM buffer of 256-/264-bytes. EEPROM emulation (bit or byte alterability) is easily handled with a self-contained three step read-modify-write operation. Unlike conventional Flash memories that are accessed randomly with multiple address lines and a parallel interface, the Atmel DataFlash® uses a RapidS serial interface to sequentially access its data. The simple sequential access dramatically reduces active pin count, facilitates hardware layout, increases system reliability, minimizes switching noise, and reduces package size.



**2-megabit
2.7V Minimum
DataFlash**

Atmel AT45DB021D

Atmel RapidS

3638J–DFLASH–5/10



The device is optimized for use in many commercial and industrial applications where high-density, low-pin count, low-voltage and low-power are essential.

To allow for simple in-system reprogrammability, the Atmel® AT45DB021D does not require high input voltages for programming. The device operates from a single power supply, 2.7V to 3.6V, for both the program and read operations. The AT45DB021D is enabled through the chip select pin (\overline{CS}) and accessed via a three-wire interface consisting of the Serial Input (SI), Serial Output (SO), and the Serial Clock (SCK).

All programming and erase cycles are self-timed.

1. Pin Configurations and Pinouts

Table 1-1. Pin Configurations

Symbol	Name and Function	Asserted State	Type
\overline{CS}	<p>Chip Select: Asserting the \overline{CS} pin selects the device. When the \overline{CS} pin is deasserted, the device will be deselected and normally be placed in the standby mode (not Deep Power-Down mode), and the output pin (SO) will be in a high-impedance state. When the device is deselected, data will not be accepted on the input pin (SI).</p> <p>A high-to-low transition on the \overline{CS} pin is required to start an operation, and a low-to-high transition is required to end an operation. When ending an internally self-timed operation such as a program or erase cycle, the device will not enter the standby mode until the completion of the operation.</p>	Low	Input
SCK	<p>Serial Clock: This pin is used to provide a clock to the device and is used to control the flow of data to and from the device. Command, address, and input data present on the SI pin is always latched on the rising edge of SCK, while output data on the SO pin is always clocked out on the falling edge of SCK.</p>	–	Input
SI	<p>Serial Input: The SI pin is used to shift data into the device. The SI pin is used for all data input including command and address sequences. Data on the SI pin is always latched on the rising edge of SCK.</p>	–	Input
SO	<p>Serial Output: The SO pin is used to shift data out from the device. Data on the SO pin is always clocked out on the falling edge of SCK.</p>	–	Output
\overline{WP}	<p>Write Protect: When the \overline{WP} pin is asserted, all sectors specified for protection by the Sector Protection Register will be protected against program and erase operations regardless of whether the Enable Sector Protection command has been issued or not. The \overline{WP} pin functions independently of the software controlled protection method. After the \overline{WP} pin goes low, the content of the Sector Protection Register cannot be modified.</p> <p>If a program or erase command is issued to the device while the \overline{WP} pin is asserted, the device will simply ignore the command and perform no operation. The device will return to the idle state once the \overline{CS} pin has been deasserted. The Enable Sector Protection command and Sector Lockdown command, however, will be recognized by the device when the \overline{WP} pin is asserted.</p> <p>The \overline{WP} pin is internally pulled-high and may be left floating if hardware controlled protection will not be used. However, it is recommended that the \overline{WP} pin also be externally connected to V_{CC} whenever possible.</p>	Low	Input
RESET	<p>Reset: A low state on the reset pin (\overline{RESET}) will terminate the operation in progress and reset the internal state machine to an idle state. The device will remain in the reset condition as long as a low level is present on the \overline{RESET} pin. Normal operation can resume once the \overline{RESET} pin is brought back to a high level.</p> <p>The device incorporates an internal power-on reset circuit, so there are no restrictions on the \overline{RESET} pin during power-on sequences. If this pin and feature are not utilized it is recommended that the \overline{RESET} pin be driven high externally.</p>	Low	Input
V_{CC}	<p>Device Power Supply: The V_{CC} pin is used to supply the source voltage to the device. Operations at invalid V_{CC} voltages may produce spurious results and should not be attempted.</p>	–	Power
GND	<p>Ground: The ground reference for the power supply. GND should be connected to the system ground.</p>	–	Ground

Figure 1-1. SOIC Top View

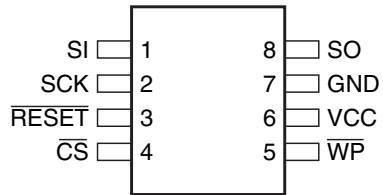
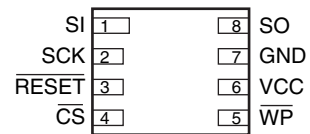
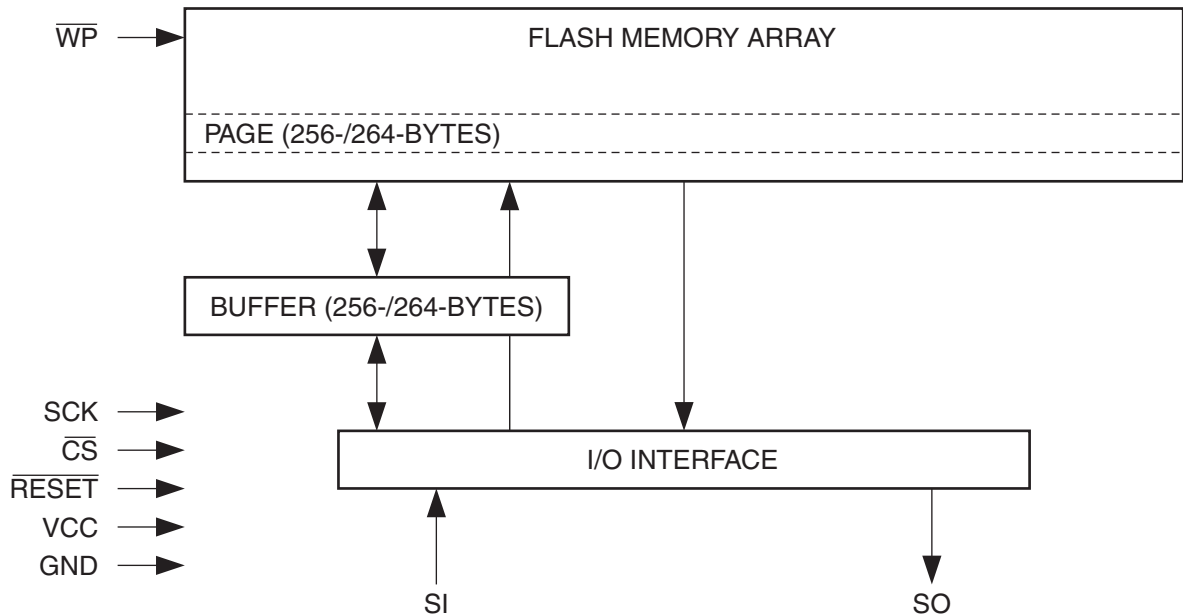


Figure 1-2. UDFN Top View⁽¹⁾



Notes: 1. The metal pad on the bottom of the UDFN package is floating. This pad can be a “No Connect” or connected to GND

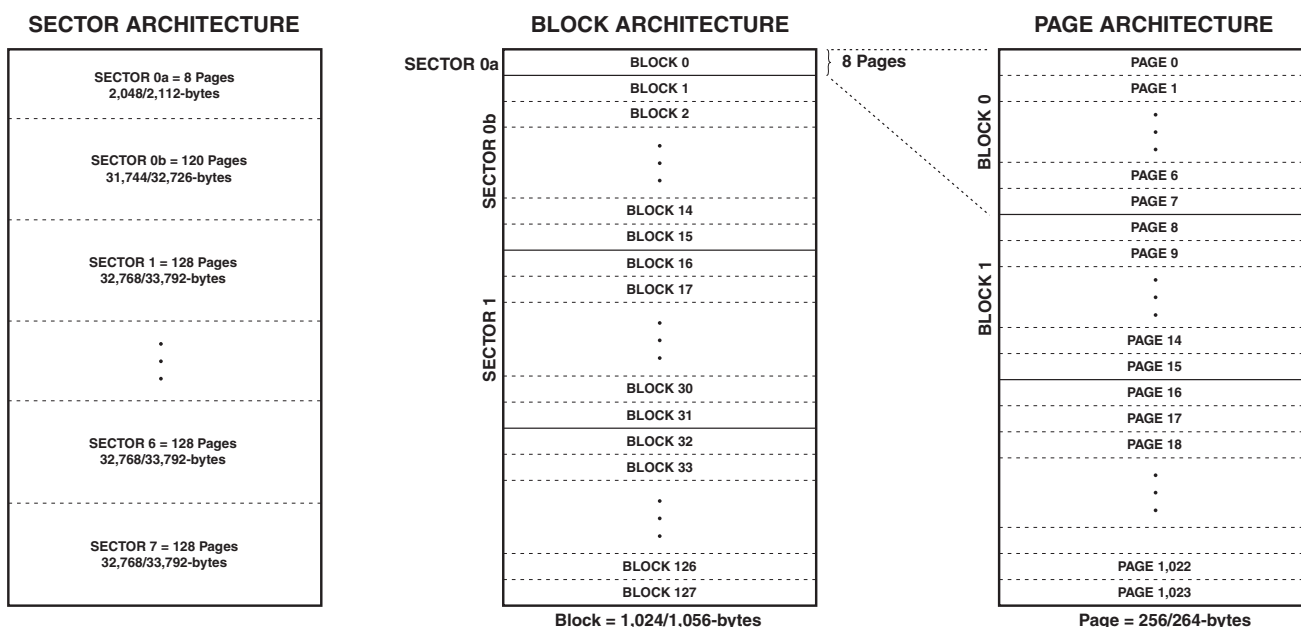
Figure 1-3. Block Diagram



2. Memory Array

To provide optimal flexibility, the memory array of the Atmel® AT45DB021D is divided into three levels of granularity comprising of sectors, blocks, and pages. The “Memory Architecture Diagram” illustrates the breakdown of each level and details the number of pages per sector and block. All program operations to the Atmel DataFlash® occur on a page-by-page basis. The erase operations can be performed at the chip, sector, block or page level.

Figure 2-1. Memory Architecture Diagram



3. Device Operation

The device operation is controlled by instructions from the host processor. The list of instructions and their associated opcodes are contained in [Tables 13-1 through 13-7](#). A valid instruction starts with the falling edge of \overline{CS} followed by the appropriate 8-bit opcode and the desired buffer or main memory address location. While the \overline{CS} pin is low, toggling the SCK pin controls the loading of the opcode and the desired buffer or main memory address location through the SI (serial input) pin. All instructions, addresses, and data are transferred with the most significant bit (MSB) first.

Buffer addressing for the DataFlash standard page size (264-bytes) is referenced in the datasheet using the terminology BFA8 - BFA0 to denote the nine address bits required to designate a byte address within a buffer. Main memory addressing is referenced using the terminology PA9 - PA0 and BA8 - BA0, where PA9 - PA0 denotes the 10-address bits required to designate a page address and BA8 - BA0 denotes the nine address bits required to designate a byte address within the page.

For the “Power of 2” binary page size (256-bytes), the Buffer addressing is referenced in the datasheet using the conventional terminology BFA7 - BFA0 to denote the eight address bits required to designate a byte address within a buffer. Main memory addressing is referenced using the terminology A17 - A0, where A17 - A8 denotes the 10-address bits required to designate a page address and A7 - A0 denotes the eight address bits required to designate a byte address within a page.

4. Read Commands

By specifying the appropriate opcode, data can be read from the main memory or from the SRAM data buffer. The Atmel DataFlash® supports Atmel RapidS™ protocols for Mode 0 and Mode 3. Please refer to the “Detailed Bit-level Read Timing” diagrams in this datasheet for details on the clock cycle sequences for each mode.

4.1 Continuous Array Read (Legacy Command – E8H): Up to 66MHz

By supplying an initial starting address for the main memory array, the Continuous Array Read command can be utilized to sequentially read a continuous stream of data from the device by simply providing a clock signal; no additional addressing information or control signals need to be provided. The DataFlash incorporates an internal address counter that will automatically increment on every clock cycle, allowing one continuous read operation without the need of additional address sequences. To perform a continuous read from the DataFlash standard page size (264-bytes), an opcode of E8H must be clocked into the device followed by three address bytes (which comprise the 24-bit page and byte address sequence) and four don't care bytes. The first 10-bits (PA9 - PA0) of the 19-bit address sequence specify which page of the main memory array to read, and the last 9-bits (BA8 - BA0) of the 19-bit address sequence specify the starting byte address within the page. To perform a continuous read from the binary page size (256-bytes), the opcode (E8H) must be clocked into the device followed by three address bytes and four don't care bytes. The first 10-bits (A17 - A8) of the 18-bits sequence specify which page of the main memory array to read, and the last 8-bits (A7 - A0) of the 18-bits address sequence specify the starting byte address within the page. The don't care bytes that follow the address bytes are needed to initialize the read operation. Following the don't care bytes, additional clock pulses on the SCK pin will result in data being output on the SO (serial output) pin.

The $\overline{\text{CS}}$ pin must remain low during the loading of the opcode, the address bytes, the don't care bytes, and the reading of data. When the end of a page in main memory is reached during a Continuous Array Read, the device will continue reading at the beginning of the next page with no delays incurred during the page boundary crossover (the crossover from the end of one page to the beginning of the next page). When the last bit in the main memory array has been read, the device will continue reading back at the beginning of the first page of memory. As with crossing over page boundaries, no delays will be incurred when wrapping around from the end of the array to the beginning of the array.

A low-to-high transition on the $\overline{\text{CS}}$ pin will terminate the read operation and tri-state the output pin (SO). The maximum SCK frequency allowable for the Continuous Array Read is defined by the f_{CAR1} specification. The Continuous Array Read bypasses the data buffer and leaves the contents of the buffer unchanged.

4.2 Continuous Array Read (High Frequency Mode – 0BH): Up to 66MHz

This command can be used with the serial interface to read the main memory array sequentially in high speed mode for any clock frequency up to the maximum specified by f_{CAR1} . To perform a continuous read array with the page size set to 264-bytes, the $\overline{\text{CS}}$ must first be asserted then an opcode 0BH must be clocked into the device followed by three address bytes and a dummy byte. The first 10-bits (PA9 - PA0) of the 19-bit address sequence specify which page of the main memory array to read, and the last nine bits (BA8 - BA0) of the 19-bit address sequence specify the starting byte address within the page. To perform a continuous read with the page size set to 256-bytes, the opcode, 0BH, must be clocked into the device followed by three address bytes (A17 - A0) and a dummy byte. Following the dummy byte, additional clock pulses on the SCK pin will result in data being output on the SO (serial output) pin.

The CS pin must remain low during the loading of the opcode, the address bytes, and the reading of data. When the end of a page in the main memory is reached during a Continuous Array Read, the device will continue reading at the beginning of the next page with no delays incurred during the page boundary crossover (the crossover from the end of one page to the beginning of the next page). When the last bit in the main memory array has been read, the device will continue reading back at the beginning of the first page of memory. As with crossing over page

boundaries, no delays will be incurred when wrapping around from the end of the array to the beginning of the array. A low-to-high transition on the CS pin will terminate the read operation and tri-state the output pin (SO). The maximum SCK frequency allowable for the Continuous Array Read is defined by the f_{CAR1} specification. The Continuous Array Read bypasses the data buffer and leaves the contents of the buffer unchanged.

4.3 Continuous Array Read (Low Frequency Mode: 03H): Up to 33MHz

This command can be used with the serial interface to read the main memory array sequentially without a dummy byte up to maximum frequencies specified by f_{CAR2} . To perform a continuous read array with the page size set to 264-bytes, the \overline{CS} must first be asserted then an opcode, 03H, must be clocked into the device followed by three address bytes (which comprise the 24-bit page and byte address sequence). The first 10-bits (PA9 - PA0) of the 19-bit address sequence specify which page of the main memory array to read, and the last nine bits (BA8 - BA0) of the 19-bit address sequence specify the starting byte address within the page. To perform a continuous read with the page size set to 256-bytes, the opcode, 03H, must be clocked into the device followed by three address bytes (A17 - A0). Following the address bytes, additional clock pulses on the SCK pin will result in data being output on the SO (serial output) pin.

The CS pin must remain low during the loading of the opcode, the address bytes, and the reading of data. When the end of a page in the main memory is reached during a Continuous Array Read, the device will continue reading at the beginning of the next page with no delays incurred during the page boundary crossover (the crossover from the end of one page to the beginning of the next page). When the last bit in the main memory array has been read, the device will continue reading back at the beginning of the first page of memory. As with crossing over page boundaries, no delays will be incurred when wrapping around from the end of the array to the beginning of the array. A low-to-high transition on the CS pin will terminate the read operation and tri-state the output pin (SO). The Continuous Array Read bypasses the data buffer and leaves the contents of the buffer unchanged.

4.4 Main Memory Page Read

A main memory page read allows the user to read data directly from any one of the 2,048-pages in the main memory, bypassing the data buffer and leaving the contents of the buffer unchanged. To start a page read from the Atmel® DataFlash® standard page size (264-bytes), an opcode of D2H must be clocked into the device followed by three address bytes (which comprise the 24-bit page and byte address sequence) and four don't care bytes. The first 10-bits (PA9 - PA0) of the 19-bit address sequence specify the page in main memory to be read, and the last 9-bits (BA8 - BA0) of the 19-bit address sequence specify the starting byte address within that page. To start a page read from the binary page size (256-bytes), the opcode D2H must be clocked into the device followed by three address bytes and four don't care bytes. The first 10-bits (A17 - A8) of the 18-bit sequence specify which page of the main memory array to read, and the last 8-bits (A7 - A0) of the 18-bit address sequence specify the starting byte address within the page. The don't care bytes that follow the address bytes are sent to initialize the read operation. Following the don't care bytes, additional pulses on SCK result in data being output on the SO (serial output) pin. The \overline{CS} pin must remain low during the loading of the opcode, the address bytes, the don't care bytes, and the reading of data. When the end of a page in main memory is reached, the device will continue reading back at the beginning of the same page. A low-to-high transition on the \overline{CS} pin will terminate the read operation and tri-state the output pin (SO). The maximum SCK frequency allowable for the Main Memory Page Read is defined by the f_{SCK} specification. The Main Memory Page Read bypasses the data buffer and leaves the contents of the buffer unchanged.

4.5 Buffer Read

The SRAM data buffer can be accessed independently from the main memory array, and utilizing the Buffer Read Command allows data to be sequentially read directly from the buffer. Two opcodes, D4H or D1H, can be used for the Buffer Read Command. The use of each opcode depends on the maximum SCK frequency that will be used to

read data from the buffer. The D4H opcode can be used at any SCK frequency up to the maximum specified by f_{CAR1} . The D1H opcode can be used for lower frequency read operations up to the maximum specified by f_{CAR2} .

To perform a buffer read from the Atmel® DataFlash® standard buffer (264-bytes), the opcode must be clocked into the device followed by three address bytes comprised of 15 don't care bits and 9 buffer address bits (BFA8 - BFA0). To perform a buffer read from the binary buffer (256-bytes), the opcode must be clocked into the device followed by three address bytes comprised of 16 don't care bits and eight buffer address bits (BFA7 - BFA0). Following the address bytes, one don't care byte must be clocked in to initialize the read operation. The \overline{CS} pin must remain low during the loading of the opcode, the address bytes, the don't care bytes, and the reading of data. When the end of a buffer is reached, the device will continue reading back at the beginning of the buffer. A low-to-high transition on the \overline{CS} pin will terminate the read operation and tri-state the output pin (SO).

5. Program and Erase Commands

5.1 Buffer Write

Data can be clocked in from the input pin (SI) into the buffer. To load data into the DataFlash standard buffer (264-bytes), a 1-byte opcode, 84H, must be clocked into the device followed by three address bytes comprised of 15 don't care bits and nine buffer address bits (BFA8 - BFA0). The nine buffer address bits specify the first byte in the buffer to be written. To load data into the binary buffers (256-bytes each), a 1-byte opcode, 84H, must be clocked into the device followed by three address bytes comprised of 16 don't care bits and eight buffer address bits (BFA7 - BFA0). The eight buffer address bits specify the first byte in the buffer to be written. After the last address byte has been clocked into the device, data can then be clocked in on subsequent clock cycles. If the end of the data buffer is reached, the device will wrap around back to the beginning of the buffer. Data will continue to be loaded into the buffer until a low-to-high transition is detected on the \overline{CS} pin.

5.2 Buffer to Main Memory Page Program with Built-in Erase

Data written into the buffer can be programmed into the main memory. A 1-byte opcode, 83H, must be clocked into the device. For the DataFlash standard page size (264-bytes), the opcode must be followed by three address bytes consist of five don't care bits, 10 page address bits (PA9 - PA0) that specify the page in the main memory to be written and nine don't care bits. To perform a buffer to main memory page program with built-in erase for the binary page size (256-bytes), the opcode 83H must be clocked into the device followed by three address bytes consisting of six don't care bits, 10 page address bits (A17 - A8) that specify the page in the main memory to be written and eight don't care bits. When a low-to-high transition occurs on the \overline{CS} pin, the part will first erase the selected page in main memory (the erased state is a logic 1) and then program the data stored in the buffer into the specified page in main memory. Both the erase and the programming of the page are internally self-timed and should take place in a maximum time of t_{EP} . During this time, the status register will indicate that the part is busy.

5.3 Buffer to Main Memory Page Program without Built-in Erase

A previously-erased page within main memory can be programmed with the contents of the buffer. A 1-byte opcode, 88H, must be clocked into the device. For the DataFlash standard page size (264-bytes), the opcode must be followed by three address bytes consist of five don't care bits, 10 page address bits (PA9 - PA0) that specify the page in the main memory to be written and nine don't care bits. To perform a buffer to main memory page program without built-in erase for the binary page size (256-bytes), the opcode 88H must be clocked into the device followed by three address bytes consisting of six don't care bits, 10 page address bits (A17 - A8) that specify the page in the main memory to be written and eight don't care bits. When a low-to-high transition occurs on the \overline{CS} pin, the part will program the data stored in the buffer into the specified page in the main memory. It is necessary that the page in main memory that is being programmed has been previously erased using one of the erase commands (Page Erase or Block Erase). The programming of the page is internally self-timed and should take place in a maximum time of t_p . During this time, the status register will indicate that the part is busy.

5.4 Page Erase

The Page Erase command can be used to individually erase any page in the main memory array allowing the Buffer to Main Memory Page Program to be utilized at a later time. To perform a page erase in the Atmel® DataFlash® standard page size (264-bytes), an opcode of 81H must be loaded into the device, followed by three address bytes comprised of five don't care bits, 10 page address bits (PA9 - PA0) that specify the page in the main memory to be erased and nine don't care bits. To perform a page erase in the binary page size (256-bytes), the opcode 81H must be loaded into the device, followed by three address bytes consist of six don't care bits, 10 page address bits (A17 - A8) that specify the page in the main memory to be erased and eight don't care bits. When a low-to-high transition occurs on the \overline{CS} pin, the part will erase the selected page (the erased state is a logical 1). The erase operation is internally self-timed and should take place in a maximum time of t_{PE} . During this time, the status register will indicate that the part is busy.

5.5 Block Erase

A block of eight pages can be erased at one time. This command is useful when large amounts of data has to be written into the device. This will avoid using multiple Page Erase Commands. To perform a block erase for the DataFlash standard page size (264-bytes), an opcode of 50H must be loaded into the device, followed by three address bytes comprised of five don't care bits, seven page address bits (PA9 - PA3) and 12 don't care bits. The seven page address bits are used to specify which block of eight pages is to be erased. To perform a block erase for the binary page size (256-bytes), the opcode 50H must be loaded into the device, followed by three address bytes consisting of six don't care bits, seven page address bits (A17 - A11) and 11 don't care bits. The 9-page address bits are used to specify which block of eight pages is to be erased. When a low-to-high transition occurs on the \overline{CS} pin, the part will erase the selected block of eight pages. The erase operation is internally self-timed and should take place in a maximum time of t_{BE} . During this time, the status register will indicate that the part is busy.

Table 5-1. Block Erase Addressing

PA9/ A17	PA8/ A16	PA7/ A15	PA6/ A14	PA5/ A13	PA4/ A12	PA3/ A11	PA2/ A10	PA1/ A9	PA0/ A8	Block
0	0	0	0	0	0	0	X	X	X	0
0	0	0	0	0	0	1	X	X	X	1
0	0	0	0	0	1	0	X	X	X	2
0	0	0	0	0	1	1	X	X	X	3
•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•
1	1	1	1	1	0	0	X	X	X	124
1	1	1	1	1	0	1	X	X	X	125
1	1	1	1	1	1	0	X	X	X	126
1	1	1	1	1	1	1	X	X	X	127

5.6 Sector Erase

The Sector Erase command can be used to individually erase any sector in the main memory. There are four sectors and only one sector can be erased at one time. To perform sector 0a or sector 0b erase for the DataFlash standard page size (264-bytes), an opcode of 7CH must be loaded into the device, followed by three address bytes comprised of five don't care bits, seven page address bits (PA9 - PA3) and 12 don't care bits. To perform a sector 1-7 erase, the opcode 7CH must be loaded into the device, followed by three address bytes comprised of five don't care bits, three page address bits (PA9 - PA7) and 16 don't care bits. To perform sector 0a or sector 0b erase for

the binary page size (25-bytes), an opcode of 7CH must be loaded into the device, followed by three address bytes comprised of six don't care bits and seven page address bits (A17 - A11) and 11 don't care bits. To perform a sector 1-seven erase, the opcode 7CH must be loaded into the device, followed by three address bytes comprised of six don't care bit and three page address bits (A17 - A15) and 16 don't care bits. The page address bits are used to specify any valid address location within the sector which is to be erased. When a low-to-high transition occurs on the \overline{CS} pin, the part will erase the selected sector. The erase operation is internally self-timed and should take place in a maximum time of t_{SE} . During this time, the status register will indicate that the part is busy.

Table 5-2. Sector Erase Addressing

PA9/ A17	PA8/ A16	PA7/ A15	PA6/ A14	PA5/ A13	PA4/ A12	PA3/ A11	PA2/ A10	PA1/ A9	PA0/ A8	Sector
0	0	0	0	0	0	0	X	X	X	0a
0	0	0	0	0	0	1	X	X	X	0b
•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•
0	1	1	X	X	X	X	X	X	X	5
1	0	0	X	X	X	X	X	X	X	6
1	1	1	X	X	X	X	X	X	X	7

5.7 Chip Erase

The entire main memory can be erased at one time by using the Chip Erase command.

To execute the Chip Erase command, a 4-byte command sequence C7H, 94H, 80H and 9AH must be clocked into the device. Since the entire memory array is to be erased, no address bytes need to be clocked into the device, and any data clocked in after the opcode will be ignored. After the last bit of the opcode sequence has been clocked in, the \overline{CS} pin can be deasserted to start the erase process. The erase operation is internally self-timed and should take place in a time of t_{CE} . During this time, the Status Register will indicate that the device is busy.

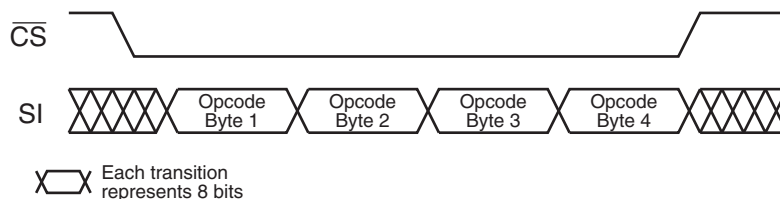
The Chip Erase command will not affect sectors that are protected or locked down; the contents of those sectors will remain unchanged. Only those sectors that are not protected or locked down will be erased.

The \overline{WP} pin can be asserted while the device is erasing, but protection will not be activated until the internal erase cycle completes.

Table 5-3. Chip Erase Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Chip Erase	C7H	94H	80H	9AH

Figure 5-1. Chip Erase



5.8 Main Memory Page Program Through Buffer

This operation is a combination of the Buffer Write and Buffer to Main Memory Page Program with Built-in Erase operations. Data is first clocked into the buffer from the input pin (SI) and then programmed into a specified page in the main memory. To perform a main memory page program through buffer for the Atmel® DataFlash® standard page size (264-bytes), a 1-byte opcode, 82H, must first be clocked into the device, followed by three address bytes. The address bytes are comprised of five don't care bits, 10 page address bits, (PA9 - PA0) that select the page in the main memory where data is to be written, and nine buffer address bits (BFA8 - BFA0) that select the first byte in the buffer to be written. To perform a main memory page program through buffer for the binary page size (256-bytes), the opcode 82H must be clocked into the device followed by three address bytes consisting of six don't care bits, 10 page address bits (A17 - A8) that specify the page in the main memory to be written, and eight buffer address bits (BFA7 - BFA0) that selects the first byte in the buffer to be written. After all address bytes are clocked in, the part will take data from the input pins and store it in the specified data buffer. If the end of the buffer is reached, the device will wrap around back to the beginning of the buffer. When there is a low-to-high transition on the \overline{CS} pin, the part will first erase the selected page in main memory to all ones and then program the data stored in the buffer into that memory page. Both the erase and the programming of the page are internally self-timed and should take place in a maximum time of t_{EP} . During this time, the status register will indicate that the part is busy.

6. Sector Protection

Two protection methods, hardware and software controlled, are provided for protection against inadvertent or erroneous program and erase cycles. The software controlled method relies on the use of software commands to enable and disable sector protection while the hardware controlled method employs the use of the Write Protect (\overline{WP}) pin. The selection of which sectors that are to be protected or unprotected against program and erase operations is specified in the nonvolatile Sector Protection Register. The status of whether or not sector protection has been enabled or disabled by either the software or the hardware controlled methods can be determined by checking the Status Register.

6.1 Software Sector Protection

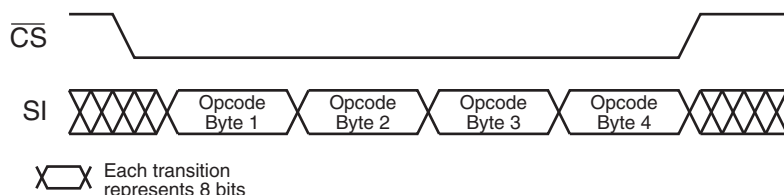
6.1.1 Enable Sector Protection Command

Sectors specified for protection in the Sector Protection Register can be protected from program and erase operations by issuing the Enable Sector Protection command. To enable the sector protection using the software controlled method, the \overline{CS} pin must first be asserted as it would be with any other command. Once the \overline{CS} pin has been asserted, the appropriate 4-byte command sequence must be clocked in via the input pin (SI). After the last bit of the command sequence has been clocked in, the \overline{CS} pin must be deasserted after which the sector protection will be enabled.

Table 6-1. Enable Sector Protection Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Enable Sector Protection	3DH	2AH	7FH	A9H

Figure 6-1. Enable Sector Protection



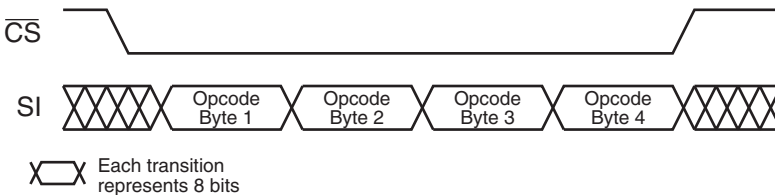
6.1.2 Disable Sector Protection Command

To disable the sector protection using the software controlled method, the \overline{CS} pin must first be asserted as it would be with any other command. Once the \overline{CS} pin has been asserted, the appropriate 4-byte sequence for the Disable Sector Protection command must be clocked in via the input pin (SI). After the last bit of the command sequence has been clocked in, the \overline{CS} pin must be deasserted after which the sector protection will be disabled. The \overline{WP} pin must be in the deasserted state; otherwise, the Disable Sector Protection command will be ignored.

Table 6-2. Disenable Sector Protection Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Disable Sector Protection	3DH	2AH	7FH	9AH

Figure 6-2. Disable Sector Protection



6.1.3 Various Aspects About Software Controlled Protection

Software controlled protection is useful in applications in which the \overline{WP} pin is not or cannot be controlled by a host processor. In such instances, the \overline{WP} pin may be left floating (the \overline{WP} pin is internally pulled high) and sector protection can be controlled using the Enable Sector Protection and Disable Sector Protection commands.

If the device is power cycled, then the software controlled protection will be disabled. Once the device is powered up, the Enable Sector Protection command should be reissued if sector protection is desired and if the \overline{WP} pin is not used.

7. Hardware Controlled Protection

Sectors specified for protection in the Sector Protection Register and the Sector Protection Register itself can be protected from program and erase operations by asserting the \overline{WP} pin and keeping the pin in its asserted state. The Sector Protection Register and any sector specified for protection cannot be erased or reprogrammed as long as the \overline{WP} pin is asserted. In order to modify the Sector Protection Register, the \overline{WP} pin must be deasserted. If the \overline{WP} pin is permanently connected to GND, then the content of the Sector Protection Register cannot be changed. If the \overline{WP} pin is deasserted, or permanently connected to V_{CC} , then the content of the Sector Protection Register can be modified.

The \overline{WP} pin will override the software controlled protection method but only for protecting the sectors. For example, if the sectors were not previously protected by the Enable Sector Protection command, then simply asserting the \overline{WP} pin would enable the sector protection within the maximum specified t_{WPE} time. When the \overline{WP} pin is deasserted; however, the sector protection would no longer be enabled (after the maximum specified t_{WPD} time) as long as the Enable Sector Protection command was not issued while the \overline{WP} pin was asserted. If the Enable Sector Protection command was issued before or while the \overline{WP} pin was asserted, then simply deasserting the \overline{WP} pin would not disable the sector protection. In this case, the Disable Sector Protection command would need to be issued while the \overline{WP} pin is deasserted to disable the sector protection. The Disable Sector Protection command is also ignored whenever the \overline{WP} pin is asserted.

A noise filter is incorporated to help protect against spurious noise that may inadvertently assert or deassert the \overline{WP} pin.

The table below details the sector protection status for various scenarios of the \overline{WP} pin, the Enable Sector Protection command, and the Disable Sector Protection command.

Figure 7-1. \overline{WP} Pin and Protection Status

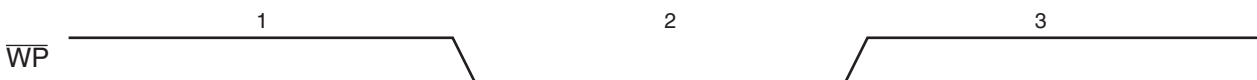


Table 7-1. \overline{WP} Pin and Protection Status

Time Period	\overline{WP} Pin	Enable Sector Protection Command	Disable Sector Protection Command	Sector Protection Status	Sector Protection Register
1	High	Command Not Issued Previously – Issue Command	X Issue Command –	Disabled Disabled Enabled	Read/Write Read/Write Read/Write
2	Low	X	X	Enabled	Read Only
3	High	Command Issued During Period 1 or 2 – Issue Command	Not Issued Yet Issue Command –	Enabled Disabled Enabled	Read/Write Read/Write Read/Write

7.1 Sector Protection Register

The nonvolatile Sector Protection Register specifies which sectors are to be protected or unprotected with either the software or hardware controlled protection methods. The Sector Protection Register contains 8-bytes of data, of which byte locations zero through seven contain values that specify whether sectors zero through seven will be protected or unprotected. The Sector Protection Register is user modifiable and must first be erased before it can be reprogrammed. [Table 7-3](#) illustrates the format of the Sector Protection Register

Table 7-2. Sector Protection Register.

Sector Number	0 (0a, 0b)	1 to 7
Protected	See Table 7-3	FFH
Unprotected		00H

Table 7-3. Sector 0 (0a, 0b)

	0a	0b	Bit 3, 2	Bit 1, 0	Data Value
	(Page 0-7)	(Page 8-127)			
	Bit 7, 6	Bit 5, 4			
Sectors 0a, 0b Unprotected	00	00	xx	xx	0xH
Protect Sector 0a	11	00	xx	xx	CxH
Protect Sector 0b (Page 8-127)	00	11	xx	xx	3xH
Protect Sectors 0a (Page 0-7), 0b (Page 8-127) ⁽¹⁾	11	11	xx	xx	FxH

Note: 1. The default value for bytes 0 through 7 when shipped from Atmel is 00H
x = don't care

7.1.1 Erase Sector Protection Register Command

In order to modify and change the values of the Sector Protection Register, it must first be erased using the Erase Sector Protection Register command.

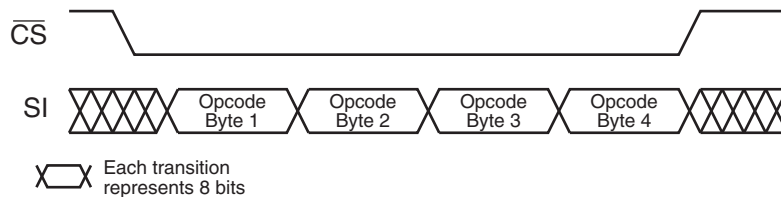
To erase the Sector Protection Register, the \overline{CS} pin must first be asserted as it would be with any other command. Once the \overline{CS} pin has been asserted, the appropriate 4-byte opcode sequence must be clocked into the device via the SI pin. The 4-byte opcode sequence must start with 3DH and be followed by 2AH, 7FH, and CFH. After the last bit of the opcode sequence has been clocked in, the \overline{CS} pin must be deasserted to initiate the internally self-timed erase cycle. The erasing of the Sector Protection Register should take place in a time of t_{PE} , during which time the Status Register will indicate that the device is busy. If the device is powered-down before the completion of the erase cycle, then the contents of the Sector Protection Register cannot be guaranteed.

The Sector Protection Register can be erased with the sector protection enabled or disabled. Since the erased state (FFH) of each byte in the Sector Protection Register is used to indicate that a sector is specified for protection, leaving the sector protection enabled during the erasing of the register allows the protection scheme to be more effective in the prevention of accidental programming or erasing of the device. If for some reason an erroneous program or erase command is sent to the device immediately after erasing the Sector Protection Register and before the register can be reprogrammed, then the erroneous program or erase command will not be processed because all sectors would be protected.

Table 7-4. Erase Sector Protection Register Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Erase Sector Protection Register	3DH	2AH	7FH	CFH

Figure 7-2. Erase Sector Protection Register



7.1.2 Program Sector Protection Register Command

Once the Sector Protection Register has been erased, it can be reprogrammed using the Program Sector Protection Register command.

To program the Sector Protection Register, the \overline{CS} pin must first be asserted and the appropriate 4-byte opcode sequence must be clocked into the device via the SI pin. The 4-byte opcode sequence must start with 3DH and be followed by 2AH, 7FH, and FCH. After the last bit of the opcode sequence has been clocked into the device, the data for the contents of the Sector Protection Register must be clocked in. As described in [Section 7.1](#), the Sector Protection Register contains 4-bytes of data, so 4-bytes must be clocked into the device. The first byte of data corresponds to sector zero, the second byte corresponds to sector one, the third byte corresponds to sector two, and the last byte of data corresponding to sector three.

After the last data byte has been clocked in, the \overline{CS} pin must be deasserted to initiate the internally self-timed program cycle. The programming of the Sector Protection Register should take place in a time of t_P , during which time the Status Register will indicate that the device is busy. If the device is powered-down during the program cycle, then the contents of the Sector Protection Register cannot be guaranteed.

If the proper number of data bytes is not clocked in before the \overline{CS} pin is deasserted, then the protection status of the sectors corresponding to the bytes not clocked in can not be guaranteed. For example, if only the first two bytes are clocked in instead of the complete 8-bytes, then the protection status of the last six sectors cannot be

guaranteed. Furthermore, if more than 8-bytes of data is clocked into the device, then the data will wrap back around to the beginning of the register. For instance, if 9-bytes of data are clocked in, then the ninth byte will be stored at byte location zero of the Sector Protection Register.

If a value other than 00H or FFH is clocked into a byte location of the Sector Protection Register, then the protection status of the sector corresponding to that byte location cannot be guaranteed. For example, if a value of 17H is clocked into byte location two of the Sector Protection Register, then the protection status of sector two cannot be guaranteed.

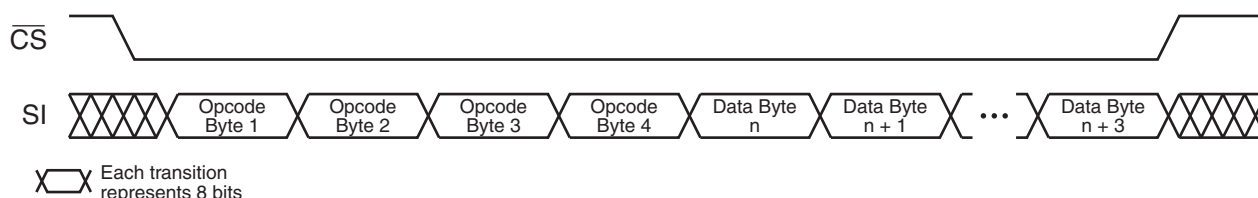
The Sector Protection Register can be reprogrammed while the sector protection enabled or disabled. Being able to reprogram the Sector Protection Register with the sector protection enabled allows the user to temporarily disable the sector protection to an individual sector rather than disabling sector protection completely.

The Program Sector Protection Register command utilizes the internal SRAM buffer for processing. Therefore, the contents of the buffer will be altered from its previous state when this command is issued.

Table 7-5. Program Sector Protection Register Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Program Sector Protection Register	3DH	2AH	7FH	FCH

Figure 7-3. Program Sector Protection Register



7.1.3 Read Sector Protection Register Command

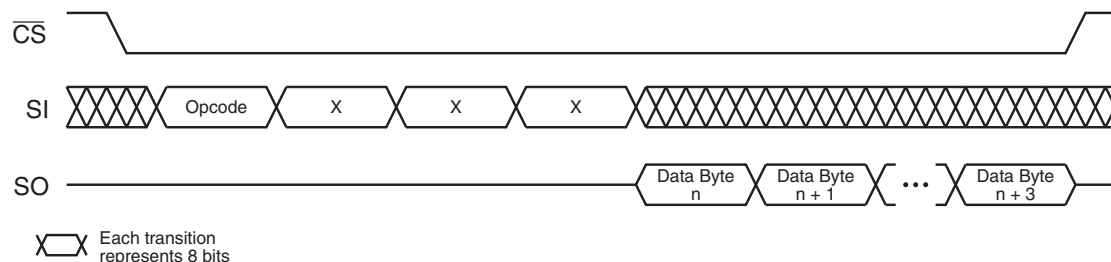
To read the Sector Protection Register, the \overline{CS} pin must first be asserted. Once the \overline{CS} pin has been asserted, an opcode of 32H and three dummy bytes must be clocked in via the SI pin. After the last bit of the opcode and dummy bytes have been clocked in, any additional clock pulses on the SCK pins will result in data for the content of the Sector Protection Register being output on the SO pin. The first byte corresponds to sector 0 (0a, 0b), the second byte corresponds to sector one, the third byte corresponds to sector two, and the last byte (byte four) corresponds to sector three. Once the last byte of the Sector Protection Register has been clocked out, any additional clock pulses will result in undefined data being output on the SO pin. The \overline{CS} must be deasserted to terminate the Read Sector Protection Register operation and put the output into a high-impedance state.

Table 7-6. Read Sector Protection Register Command

Command	Byte 1	Byte 2	Byte 3	Byte 4
Read Sector Protection Register	32H	xxH	xxH	xxH

Note: xx = Dummy Byte

Figure 7-4. Read Sector Protection Register



7.1.4 Various Aspects About the Sector Protection Register

The Sector Protection Register is subject to a limit of 10,000 erase/program cycles. Users are encouraged to carefully evaluate the number of times the Sector Protection Register will be modified during the course of the applications' life cycle. If the application requires that the Sector Protection Register be modified more than the specified limit of 10,000 cycles because the application needs to temporarily unprotect individual sectors (sector protection remains enabled while the Sector Protection Register is reprogrammed), then the application will need to limit this practice. Instead, a combination of temporarily unprotecting individual sectors along with disabling sector protection completely will need to be implemented by the application to ensure that the limit of 10,000 cycles is not exceeded.

8. Security Features

8.1 Sector Lockdown

The device incorporates a Sector Lockdown mechanism that allows each individual sector to be permanently locked so that it becomes read only. This is useful for applications that require the ability to permanently protect a number of sectors against malicious attempts at altering program code or security information. **Once a sector is locked down, it can never be erased or programmed, and it can never be unlocked.**

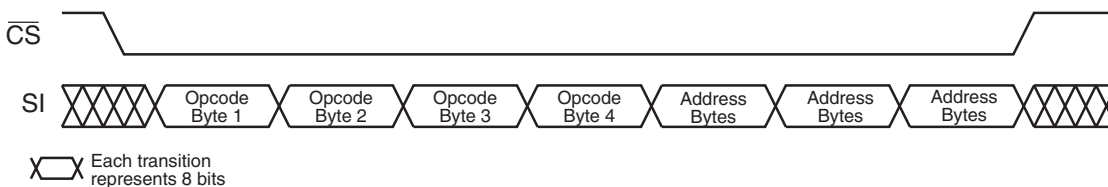
To issue the Sector Lockdown command, the \overline{CS} pin must first be asserted as it would be for any other command. Once the \overline{CS} pin has been asserted, the appropriate 4-byte opcode sequence must be clocked into the device in the correct order. The 4-byte opcode sequence must start with 3DH and be followed by 2AH, 7FH, and 30H. After the last byte of the command sequence has been clocked in, then three address bytes specifying any address within the sector to be locked down must be clocked into the device. After the last address bit has been clocked in, the \overline{CS} pin must then be deasserted to initiate the internally self-timed lockdown sequence.

The lockdown sequence should take place in a maximum time of t_p , during which time the Status Register will indicate that the device is busy. If the device is powered-down before the completion of the lockdown sequence, then the lockdown status of the sector cannot be guaranteed. In this case, it is recommended that the user read the Sector Lockdown Register to determine the status of the appropriate sector lockdown bits or bytes and reissue the Sector Lockdown command if necessary.

Table 8-1. Sector Lockdown

Command	Byte 1	Byte 2	Byte 3	Byte 4
Sector Lockdown	3DH	2AH	7FH	30H

Figure 8-1. Sector Lockdown



8.1.1 Sector Lockdown Register

Sector Lockdown Register is a nonvolatile register that contains 8-bytes of data, as shown below:

Table 8-2. Sector Lockdown Register

Sector Number	0 (0a, 0b)	1 to 7
Locked	See Below	FFH
Unlocked		00H

Table 8-3. Sector 0 (0a, 0b)

	0a	0b	Bit 3, 2	Bit 1, 0	Data Value
	(Page 0-7)	(Page 8-127)			
	Bit 7, 6	Bit 5, 4			
Sectors 0a, 0b Unlocked	00	00	00	00	00H
Sector 0a Locked (Page 0-7)	11	00	00	00	C0H
Sector 0b Locked (Page 8-127)	00	11	00	00	30H
Sectors 0a, 0b Locked (Page 0-127)	11	11	00	00	F0H

8.1.2 Reading the Sector Lockdown Register

The Sector Lockdown Register can be read to determine which sectors in the memory array are permanently locked down. To read the Sector Lockdown Register, the \overline{CS} pin must first be asserted. Once the \overline{CS} pin has been asserted, an opcode of 35H and 3 dummy bytes must be clocked into the device via the SI pin. After the last bit of the opcode and dummy bytes have been clocked in, the data for the contents of the Sector Lockdown Register will be clocked out on the SO pin. The first byte corresponds to sector 0 (0a, 0b) the second byte corresponds to sector one and the last byte (byte eight) corresponds to sector seven. After the last byte of the Sector Lockdown Register has been read, additional pulses on the SCK pin will simply result in undefined data being output on the SO pin.

Deasserting the \overline{CS} pin will terminate the Read Sector Lockdown Register operation and put the SO pin into a high-impedance state.

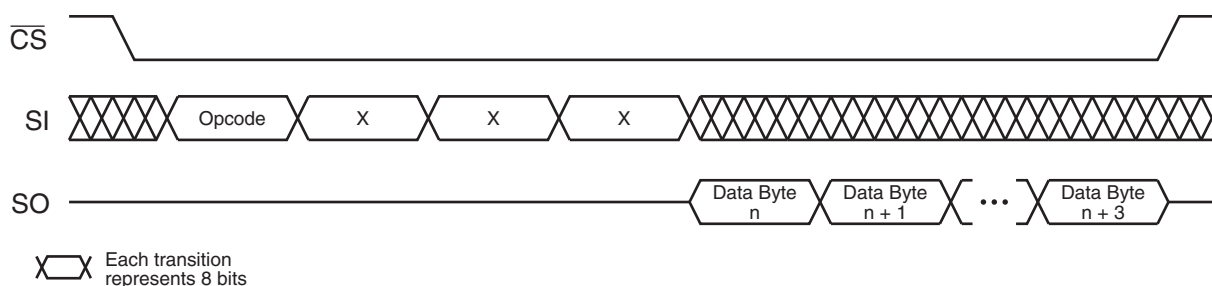
Table 8-4 details the values read from the Sector Lockdown Register.

Table 8-4. Sector Lockdown Register

Command	Byte 1	Byte 2	Byte 3	Byte 4
Read Sector Lockdown Register	35H	xxH	xxH	xxH

Note: xx = Dummy Byte

Figure 8-2. Read Sector Lockdown Register



8.2 Security Register

The device contains a specialized Security Register that can be used for purposes such as unique device serialization or locked key storage. The register is comprised of a total of 128-bytes that is divided into two portions. The first 64-bytes (byte locations 0 through 63) of the Security Register are allocated as a one-time user programmable space. Once these 64-bytes have been programmed, they cannot be reprogrammed. The remaining 64-bytes of the register (byte locations 64 through 127) are factory programmed by Atmel® and will contain a unique value for each device. The factory programmed data is fixed and cannot be changed.

Table 8-5. Security Register

	Security Register Byte Number									
	0	1	...	62	63	64	65	...	126	127
Data Type	One-time User Programmable					Factory Programmed By Atmel				

8.2.1 Programming the Security Register

The user programmable portion of the Security Register does not need to be erased before it is programmed.

To program the Security Register, the \overline{CS} pin must first be asserted and the appropriate 4-byte opcode sequence must be clocked into the device in the correct order. The 4-byte opcode sequence must start with 9BH and be followed by 00H, 00H, and 00H. After the last bit of the opcode sequence has been clocked into the device, the data for the contents of the 64-byte user programmable portion of the Security Register must be clocked in.

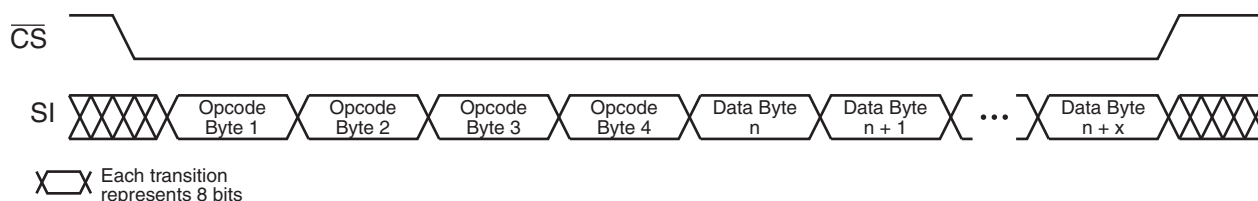
After the last data byte has been clocked in, the \overline{CS} pin must be deasserted to initiate the internally self-timed program cycle. The programming of the Security Register should take place in a time of t_p , during which time the Status Register will indicate that the device is busy. If the device is powered-down during the program cycle, then the contents of the 64-byte user programmable portion of the Security Register cannot be guaranteed.

If the full 64-bytes of data is not clocked in before the \overline{CS} pin is deasserted, then the values of the byte locations not clocked in cannot be guaranteed. For example, if only the first two bytes are clocked in instead of the complete 64 bytes, then the remaining 62-bytes of the user programmable portion of the Security Register cannot be guaranteed. Furthermore, if more than 64-bytes of data is clocked into the device, then the data will wrap back around to the beginning of the register. For instance, if 65-bytes of data are clocked in, then the 65th byte will be stored at byte location 0 of the Security Register.

The user programmable portion of the Security Register can only be programmed one time. Therefore, it is not possible to only program the first two bytes of the register and then program the remaining 62-bytes at a later time.

The Program Security Register command utilizes the internal SRAM buffer for processing. Therefore, the contents of the buffer will be altered from its previous state when this command is issued.

Figure 8-3. Program Security Register

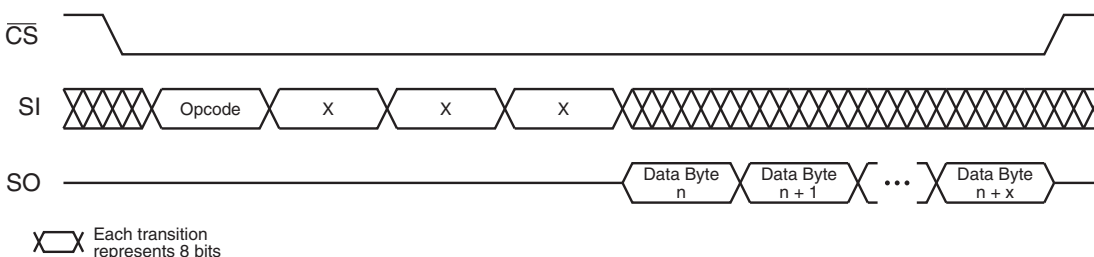


8.2.2 Reading the Security Register

The Security Register can be read by first asserting the \overline{CS} pin and then clocking in an opcode of 77H followed by three dummy bytes. After the last don't care bit has been clocked in, the content of the Security Register can be clocked out on the SO pins. After the last byte of the Security Register has been read, additional pulses on the SCK pin will simply result in undefined data being output on the SO pins.

Deasserting the \overline{CS} pin will terminate the Read Security Register operation and put the SO pins into a high-impedance state.

Figure 8-4. Read Security Register



9. Additional Commands

9.1 Main Memory Page to Buffer Transfer

A page of data can be transferred from the main memory to the buffer. To start the operation for the Atmel® DataFlash® standard page size (264-bytes), a 1-byte opcode, 53H, must be clocked into the device, followed by three address bytes comprised of five don't care bits, 10 page address bits (PA9 - PA0), which specify the page in main memory that is to be transferred, and 9 don't care bits. To perform a main memory page to buffer transfer for the binary page size (256-bytes), the opcode 53H must be clocked into the device followed by three address bytes consisting of six don't care bits, 10 page address bits (A17 - A8) which specify the page in the main memory that is to be transferred, and eight don't care bits. The \overline{CS} pin must be low while toggling the SCK pin to load the opcode and the address bytes from the input pin (SI). The transfer of the page of data from the main memory to the buffer will begin when the \overline{CS} pin transitions from a low to a high state. During the transfer of a page of data (t_{XFR}), the status register can be monitored to determine whether the transfer has been completed.

9.2 Main Memory Page to Buffer Compare

A page of data in main memory can be compared to the data in the buffer. To initiate the operation for the DataFlash standard page size, a 1-byte opcode, 60H, must be clocked into the device, followed by three address bytes consisting of five don't care bits, 10 page address bits (PA9 - PA0) that specify the page in the main memory that is to be compared to the buffer, and nine don't care bits. To start a main memory page to buffer compare for a binary page size, the opcode 60H must be clocked into the device followed by three address bytes consisting of six don't care bits, ten page address bits (A17 - A8) that specify the page in the main memory that is to be compared to the buffer, and eight don't care bits. The \overline{CS} pin must be low while toggling the SCK pin to load the opcode and the address bytes from the input pin (SI). On the low-to-high transition of the \overline{CS} pin, the data bytes in the selected main memory page will be compared with the data bytes in the buffer. During this time (t_{COMP}), the status register will indicate that the part is busy. On completion of the compare operation, bit 6 of the status register is updated with the result of the compare.

9.3 Auto Page Rewrite

This mode is only needed if multiple bytes within a page or multiple pages of data are modified in a random fashion within a sector. This mode is a combination of two operations: Main Memory Page to Buffer Transfer and Buffer to

Main Memory Page Program with Built-in Erase. A page of data is first transferred from the main memory to the buffer and then the same data (from the buffer) is programmed back into its original page of main memory. To start the rewrite operation for the Atmel® DataFlash® standard page size (264-bytes), a 1-byte opcode, 58H, must be clocked into the device, followed by three address bytes comprised of five don't care bits, 10 page address bits (PA9-PA0) that specify the page in main memory to be rewritten and nine don't care bits. To initiate an auto page rewrite for a binary page size (256-bytes), the opcode 58H must be clocked into the device followed by three address bytes consisting of six don't care bits, 10 page address bits (A17 - A8) that specify the page in the main memory that is to be written and eight don't care bits. When a low-to-high transition occurs on the \overline{CS} pin, the part will first transfer data from the page in main memory to a buffer and then program the data from the buffer back into same page of main memory. The operation is internally self-timed and should take place in a maximum time of t_{EP} . During this time, the status register indicate that the part is busy.

If a sector is programmed or reprogrammed sequentially page by page, then the programming algorithm shown in [Figure 20-1 \(page 41\)](#) is recommended. Otherwise, if multiple bytes in a page or several pages are programmed randomly in a sector, then the programming algorithm shown in [Figure 20-2 \(page 42\)](#) is recommended. Each page within a sector must be updated/rewritten at least once within every 20,000 cumulative page erase/program operations in that sector. Please contact Atmel for availability of devices that are specified to exceed the 20K cycle cumulative limit.

9.4 Status Register Read

The status register can be used to determine the device's ready/busy status, page size, a Main Memory Page to Buffer Compare operation result, the Sector Protection status or the device density. The Status Register can be read at any time, including during an internally self-timed program or erase operation. To read the status register, the \overline{CS} pin must be asserted and the opcode of D7H must be loaded into the device. After the opcode is clocked in, the 1-byte status register will be clocked out on the output pin (SO), starting with the next clock cycle. The data in the status register, starting with the MSB (bit seven), will be clocked out on the SO pin during the next eight clock cycles. After the one byte of the status register has been clocked out, the sequence will repeat itself (as long as \overline{CS} remains low and SCK is being toggled). The data in the status register is constantly updated, so each repeating sequence will output new data.

Ready/busy status is indicated using bit seven of the status register. If bit seven is a one, then the device is not busy and is ready to accept the next command. If bit seven is a zero, then the device is in a busy state. Since the data in the status register is constantly updated, the user must toggle SCK pin to check the ready/busy status. There are several operations that can cause the device to be in a busy state: Main Memory Page to Buffer Transfer, Main Memory Page to Buffer Compare, Buffer to Main Memory Page Program, Main Memory Page Program through Buffer, Page Erase, Block Erase, Sector Erase, Chip Erase and Auto Page Rewrite.

The result of the most recent Main Memory Page to Buffer Compare operation is indicated using bit six of the status register. If bit six is a zero, then the data in the main memory page matches the data in the buffer. If bit six is a one, then at least one bit of the data in the main memory page does not match the data in the buffer.

Bit one in the Status Register is used to provide information to the user whether or not the sector protection has been enabled or disabled, either by software-controlled method or hardware-controlled method. A logic 1 indicates that sector protection has been enabled and logic 0 indicates that sector protection has been disabled.

Bit zero in the Status Register indicates whether the page size of the main memory array is configured for "power of 2" binary page size (256-bytes) or the DataFlash standard page size (264-bytes). If bit zero is a one, then the page size is set to 256-bytes. If bit zero is a zero, then the page size is set to 264-bytes.

The device density is indicated using bits five, four, three, and two of the status register. For Atmel AT45DB021D, the four bits are 0101. The decimal value of these four binary bits does not equate to the device density; the four bits represent a combinational code relating to differing densities of DataFlash devices. The device density is not the same as the density code indicated in the JEDEC device ID information. The device density is provided only for backward compatibility.



Table 9-1. Status Register Format

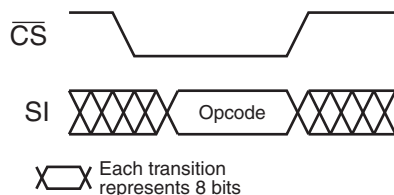
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RDY/BUSY	COMP	0	1	0	1	PROTECT	PAGE SIZE

10. Deep Power-down

After initial power-up, the device will default in standby mode. The Deep Power-down command allows the device to enter into the lowest power consumption mode. To enter the Deep Power-down mode, the \overline{CS} pin must first be asserted. Once the \overline{CS} pin has been asserted, an opcode of B9H command must be clocked in via input pin (SI). After the last bit of the command has been clocked in, the \overline{CS} pin must be de-asserted to initiate the Deep Power-down operation. After the \overline{CS} pin is de-asserted, the device will enter the Deep Power-down mode within the maximum t_{EDPD} time. Once the device has entered the Deep Power-down mode, all instructions are ignored except for the Resume from Deep Power-down command.

Table 10-1. Deep Power-down

Command	Opcode
Deep Power-down	B9H

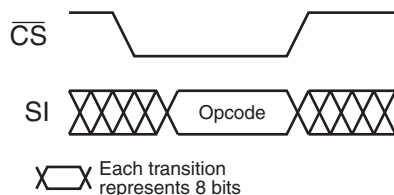
Figure 10-1. Deep Power-down


10.1 Resume from Deep Power-down

The Resume from Deep Power-down command takes the device out of the Deep Power-down mode and returns it to the normal standby mode. To Resume from Deep Power-down mode, the \overline{CS} pin must first be asserted and an opcode of ABH command must be clocked in via input pin (SI). After the last bit of the command has been clocked in, the \overline{CS} pin must be de-asserted to terminate the Deep Power-down mode. After the \overline{CS} pin is de-asserted, the device will return to the normal standby mode within the maximum t_{RDPD} time. The \overline{CS} pin must remain high during the t_{RDPD} time before the device can receive any commands. After resuming from Deep Power-down, the device will return to the normal standby mode.

Table 10-2. Resume from Deep Power-down

Command	Opcode
Resume from Deep Power-down	ABH

Figure 10-2. Resume from Deep Power-Down


11. “Power of 2” Binary Page Size Option

“Power of 2” binary page size Configuration Register is a user-programmable nonvolatile register that allows the page size of the main memory to be configured for binary page size (256-bytes) or the Atmel® DataFlash® standard page size (264-bytes). **The “power of 2” page size is a One-time Programmable (OTP) register and once the device is configured for “power of 2” page size, it cannot be reconfigured again.** The devices are initially shipped with the page size set to 264-bytes. The user has the option of ordering binary page size (256-bytes) devices from the factory. For details, please refer to [Section 21. “Ordering Information” on page 43.](#)

For the binary “power of 2” page size to become effective, the following steps must be followed:

1. Program the one-time programmable configuration register using opcode sequence 3DH, 2AH, 80H and A6H (please see [Section 11.1](#)).
2. Power cycle the device (i.e. power down and power up again).
3. The page for the binary page size can now be programmed.

If the above steps are not followed to set the page size prior to page programming, incorrect data during a read operation may be encountered.

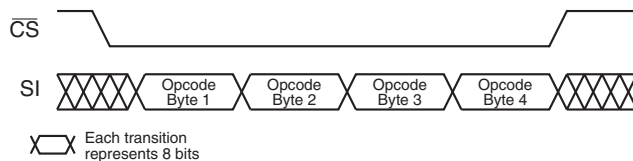
11.1 Programming the Configuration Register

To program the Configuration Register for “power of 2” binary page size, the \overline{CS} pin must first be asserted as it would be with any other command. Once the \overline{CS} pin has been asserted, the appropriate 4-byte opcode sequence must be clocked into the device in the correct order. The 4-byte opcode sequence must start with 3DH and be followed by 2AH, 80H, and A6H. After the last bit of the opcode sequence has been clocked in, the \overline{CS} pin must be deasserted to initiate the internally self-timed program cycle. The programming of the Configuration Register should take place in a time of t_p , during which time the Status Register will indicate that the device is busy. The device must be power-cycled after the completion of the program cycle to set the “power of 2” page size. If the device is powered-down before the completion of the program cycle, then setting the Configuration Register cannot be guaranteed. However, the user should check bit zero of the status register to see whether the page size was configured for binary page size. If not, the command can be re-issued again.

Table 11-1. Programming the Configuration Register

Command	Byte 1	Byte 2	Byte 3	Byte 4
Power of Two Page Size	3DH	2AH	80H	A6H

Figure 11-1. Erase Sector Protection Register



12. Manufacturer and Device ID Read

Identification information can be read from the device to enable systems to electronically query and identify the device while it is in system. The identification method and the command opcode comply with the JEDEC standard for “Manufacturer and Device ID Read Methodology for SPI Compatible Serial Interface Memory Devices”. The type of information that can be read from the device includes the JEDEC defined Manufacturer ID, the vendor specific Device ID, and the vendor specific Extended Device Information.

To read the identification information, the \overline{CS} pin must first be asserted and the opcode of 9FH must be clocked into the device. After the opcode has been clocked in, the device will begin outputting the identification data on the SO pin during the subsequent clock cycles. The first byte that will be output will be the Manufacturer ID followed by

two bytes of Device ID information. The fourth byte output will be the Extended Device Information String Length, which will be 00H indicating that no Extended Device Information follows. As indicated in the JEDEC standard, reading the Extended Device Information String Length and any subsequent data is optional.

Deasserting the \overline{CS} pin will terminate the Manufacturer and Device ID Read operation and put the SO pin into a high-impedance state. The \overline{CS} pin can be deasserted at any time and does not require that a full byte of data be read.

12.1 Manufacturer and Device ID Information

Table 12-1. Byte 1 – Manufacturer ID

Hex Value	JEDEC Assigned Code							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1FH	0	0	0	1	1	1	1	1

Manufacturer ID	1FH = Atmel
-----------------	-------------

Table 12-2. Byte 2 – Device ID (Part 1)

Hex Value	Family Code			Density Code				
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
23H	0	0	1	0	0	0	1	1

Family Code	001 = DataFlash
Density Code	00011 = 2-Mbit

Table 12-3. Byte 3 – Device ID (Part 2)

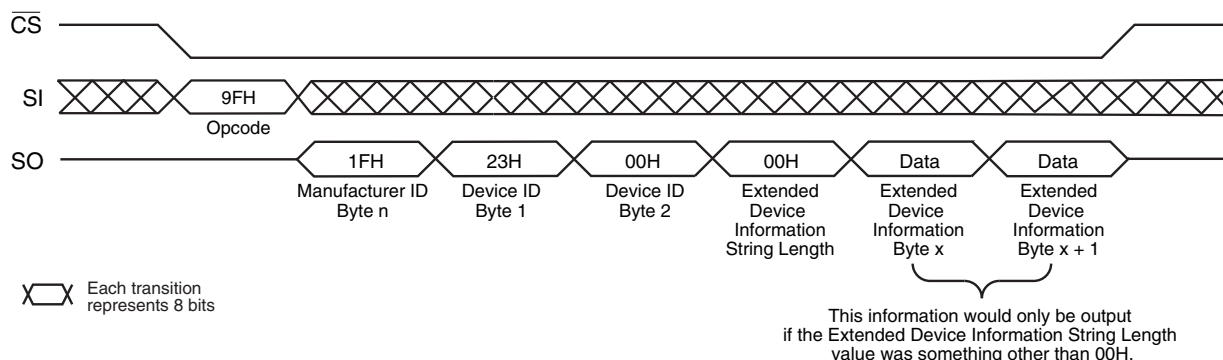
Hex Value	MLC Code			Product Version Code				
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	0	0	0	0	0	0	0	0

MLC Code	000 = 1-bit/Cell Technology
Product Version	00000 = Initial Version

Table 12-4. Byte 4 – Extended Device Information String Length

Hex Value	Byte Count							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	0	0	0	0	0	0	0	0

Byte Count	00H = 0 Bytes of Information
------------	------------------------------



Note: Based on JEDEC publication 106 (JEP106), Manufacturer ID data can be comprised of any number of bytes. Some manufacturers may have Manufacturer ID codes that are two, three or even four bytes long with the first byte(s) in the sequence being 7FH. A system should detect code 7FH as a “Continuation Code” and continue to read Manufacturer ID bytes. The first non-7FH byte would signify the last byte of Manufacturer ID data. For Atmel (and some other manufacturers), the Manufacturer ID data is comprised of only one byte.

12.2 Operation Mode Summary

The commands described previously can be grouped into four different categories to better describe which commands can be executed at what times.

Group A commands consist of:

1. Main Memory Page Read
2. Continuous Array Read
3. Read Sector Protection Register
4. Read Sector Lockdown Register
5. Read Security Register

Group B commands consist of:

1. Page Erase
2. Block Erase
3. Sector Erase
4. Chip Erase
5. Main Memory Page to Buffer Transfer
6. Main Memory Page to Buffer Compare
7. Buffer to Main Memory Page Program with Built-in Erase
8. Buffer to Main Memory Page Program without Built-in Erase
9. Main Memory Page Program through Buffer
10. Auto Page Rewrite

Group C commands consist of:

1. Buffer Read
2. Buffer Write
3. Status Register Read
4. Manufacturer and Device ID Read

Group D commands consist of:

1. Erase Sector Protection Register
2. Program Sector Protection Register
3. Sector Lockdown
4. Program Security Register

If a Group A command is in progress (not fully completed), then another command in Group A, B, C, or D should not be started. However, during the internally self-timed portion of Group B commands one through four, any command in Group C can be executed. During the internally self-timed portion of Group B commands five through ten, only Group C commands three and four can be executed. Finally, during the internally self-timed portion of a Group D command, only the Status Register Read command should be executed.

13. Command Tables

Table 13-1. Read Commands

Command	Opcode
Main Memory Page Read	D2H
Continuous Array Read (Legacy Command)	E8H
Continuous Array Read (Low Frequency)	03H
Continuous Array Read (High Frequency)	0BH
Buffer Read (Low Frequency)	D1H
Buffer Read	D4H

Table 13-2. Program and Erase Commands

Command	Opcode
Buffer Write	84H
Buffer to Main Memory Page Program with Built-in Erase	83H
Buffer to Main Memory Page Program without Built-in Erase	88H
Page Erase	81H
Block Erase	50H
Sector Erase	7CH
Chip Erase	7CH, 94H, 80H, 9AH
Main Memory Page Program through Buffer	82H

Table 13-3. Protection and Security Commands

Command	Opcode
Enable Sector Protection	3DH + 2AH + 7FH + A9H
Disable Sector Protection	3DH + 2AH + 7FH + 9AH
Erase Sector Protection Register	3DH + 2AH + 7FH + CFH
Program Sector Protection Register	3DH + 2AH + 7FH + FCH
Read Sector Protection Register	32H
Sector Lockdown	3DH + 2AH + 7FH + 30H
Read Sector Lockdown Register	35H
Program Security Register	9BH + 00H + 00H + 00H
Read Security Register	77H

Table 13-4. Additional Commands

Command	Opcode
Main Memory Page to Buffer Transfer	53H
Main Memory Page to Buffer Compare	60H
Auto Page Rewrite through Buffer	58H
Deep Power-down	B9H
Resume from Deep Power-down	ABH
Status Register Read	D7H
Manufacturer and Device ID Read	9FH

Table 13-5. Legacy Commands⁽¹⁾

Command	Opcode
Buffer Read	54H
Main Memory Page Read	52H
Continuous Array Read	68H
Status Register Read	57H

Note: 1. These legacy commands are not recommended for new designs

Table 13-6. Detailed Bit-level Addressing Sequence for Binary Page Size (256-Bytes)

Page Size = 256-bytes		Address Byte								Address Byte								Address Byte								Additional Don't Care Bytes
Opco de	Opcode	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	A17	A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
03h	0 0 0 0 0 0 1 1	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	N/A
0Bh	0 0 0 0 1 0 1 1	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	1
50h	0 1 0 1 0 0 0 0	x	x	x	x	x	x	A	A	A	A	A	A	x	x	x	x	x	x	x	x	x	x	x	x	N/A
53h	0 1 0 1 0 0 1 1	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
58h	0 1 0 1 1 0 0 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
60h	0 1 1 0 0 0 0 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
77h	0 1 1 1 0 1 1 1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	N/A
7Ch	0 1 1 1 1 1 0 0	x	x	x	x	x	x	A	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	N/A
81h	1 0 0 0 0 0 0 1	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
82h	1 0 0 0 0 0 1 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	N/A
83h	1 0 0 0 0 0 1 1	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
84h	1 0 0 0 0 1 0 0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	A	A	A	A	A	A	A	A	N/A
88h	1 0 0 0 1 0 0 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	x	x	x	x	x	x	x	x	N/A
9Fh	1 0 0 1 1 1 1 1	N/A								N/A								N/A								N/A
B9h	1 0 1 1 1 0 0 1	N/A								N/A								N/A								N/A
ABh	1 0 1 0 1 0 1 1	N/A								N/A								N/A								N/A
D1h	1 1 0 1 0 0 0 1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	A	A	A	A	A	A	A	A	N/A
D2h	1 1 0 1 0 0 1 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	4
D4h	1 1 0 1 0 1 0 0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	A	A	A	A	A	A	A	A	1
D7h	1 1 0 1 0 1 1 1	N/A								N/A								N/A								N/A
E8h	1 1 1 0 1 0 0 0	x	x	x	x	x	x	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	4

Note: x = Don't Care

Table 13-7. Detailed Bit-level Addressing Sequence for Standard Atmel DataFlash Page Size (264-Bytes)

Page Size = 264-bytes								Address Byte				Address Byte								Address Byte								Additional Don't Care Bytes					
Opcode	Opcode							Reserved	Reserved	Reserved	Reserved	Reserved	PA9	PA8	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	BA8	BA7	BA6	BA5	BA4		BA3	BA2	BA1	BA0	
03h	0	0	0	0	0	0	1	1	x	x	x	x	x	P	P	P	P	P	P	P	P	P	P	B	B	B	B	B	B	B	B	B	N/A
0Bh	0	0	0	0	1	0	1	1	x	x	x	x	x	P	P	P	P	P	P	P	P	P	B	B	B	B	B	B	B	B	B	1	
50h	0	1	0	1	0	0	0	0	x	x	x	x	x	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	x	x	x	N/A	
53h	0	1	0	1	0	0	1	1	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
58h	0	1	0	1	1	0	0	0	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
60h	0	1	1	0	0	0	0	0	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
77h	0	1	1	1	0	1	1	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	N/A	
7Ch	0	1	1	1	1	1	0	0	x	x	x	x	x	P	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	N/A	
81h	1	0	0	0	0	0	0	1	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
82h	1	0	0	0	0	0	1	0	x	x	x	x	x	P	P	P	P	P	P	P	P	B	B	B	B	B	B	B	B	B	B	N/A	
83h	1	0	0	0	0	0	1	1	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
84h	1	0	0	0	0	1	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	B	B	B	B	B	B	B	B	B	B	N/A	
88h	1	0	0	0	1	0	0	0	x	x	x	x	x	P	P	P	P	P	P	P	P	x	x	x	x	x	x	x	x	x	x	N/A	
9Fh	1	0	0	1	1	1	1	1	N/A				N/A				N/A				N/A				N/A								
B9h	1	0	1	1	1	0	0	1	N/A				N/A				N/A				N/A				N/A								
ABh	1	0	1	0	1	0	1	1	N/A				N/A				N/A				N/A				N/A								
D1h	1	1	0	1	0	0	0	1	x	x	x	x	x	x	x	x	x	x	x	x	B	B	B	B	B	B	B	B	B	B	N/A		
D2h	1	1	0	1	0	0	1	0	x	x	x	x	x	P	P	P	P	P	P	P	P	B	B	B	B	B	B	B	B	B	4		
D4h	1	1	0	1	0	1	0	0	x	x	x	x	x	x	x	x	x	x	x	B	B	B	B	B	B	B	B	B	B	1			
D7h	1	1	0	1	0	1	1	1	N/A				N/A				N/A				N/A				N/A								
E8h	1	1	1	0	1	0	0	0	x	x	x	x	x	P	P	P	P	P	P	P	P	B	B	B	B	B	B	B	B	B	4		

Note: P = Page Address Bit
 B = Byte/Buffer Address Bit
 x = Don't Care

14. Power-on/Reset State

When power is first applied to the device, or when recovering from a reset condition, the device will default to Mode 3. In addition, the output pin (SO) will be in a high impedance state, and a high-to-low transition on the \overline{CS} pin will be required to start a valid instruction. The mode (Mode 3 or Mode 0) will be automatically selected on every falling edge of \overline{CS} by sampling the inactive clock state.

14.1 Initial Power-up/Reset Timing Restrictions

At power up, the device must not be selected until the supply voltage reaches the V_{CC} (min.) and further delay of t_{VCSL} . During power-up, the internal Power-on Reset circuitry keeps the device in reset mode until the V_{CC} rises above the Power-on Reset threshold value (V_{POR}). At this time, all operations are disabled and the device does not respond to any commands. After power up is applied and the V_{CC} is at the minimum operating voltage V_{CC} (min.), the t_{VCSL} delay is required before the device can be selected in order to perform a read operation.

Similarly, the t_{PUW} delay is required after the V_{CC} rises above the Power-on Reset threshold value (V_{POR}) before the device can perform a write (Program or Erase) operation. After initial power-up, the device will default in Standby mode.

Table 14-1. Initial Power-up/Reset Timing Restrictions

Symbol	Parameter	Min	Typ	Max	Units
t_{VCSL}	V_{CC} (min.) to Chip Select low	1			ms
t_{PUW}	Power-Up Device Delay before Write Allowed			20	ms
V_{POR}	Power-On Reset Voltage	1.5		2.5	V

15. System Considerations

The serial interface is controlled by the clock SCK, serial input SI and chip select \overline{CS} pins. These signals must rise and fall monotonically and be free from noise. Excessive noise or ringing on these pins can be misinterpreted as multiple edges and cause improper operation of the device. The PC board traces must be kept to a minimum distance or appropriately terminated to ensure proper operation. If necessary, decoupling capacitors can be added on these pins to provide filtering against noise glitches.

As system complexity continues to increase, voltage regulation is becoming more important. A key element of any voltage regulation scheme is its current sourcing capability. Like all Flash memories, the peak current for Atmel® DataFlash® occur during the programming and erase operation. The regulator needs to supply this peak current requirement. An under specified regulator can cause current starvation. Besides increasing system noise, current starvation during programming or erase can lead to improper operation and possible data corruption.

In an effort to continue our goal of maintaining world-class quality leadership, Atmel has been performing extensive testing on the Atmel AT45DB021D that would not normally be done with a Serial Flash device. The testing that has been performed on the AT45DB021D involved extensive, non-stop reading of the memory array on pre-conditioned devices. The pre-conditioning of the devices, which entailed erasing and programming the entire memory array 10,000 times, was done to simulate a customer environment and to exercise the memory cells to a certain degree. The non-stop reading of the devices was done in three levels of granularity, with the first level involving a continuous, looped read of 256-bytes (a single page) of memory, the second level involving a continuous, looped-read of a 4-Kbyte (16-pages) portion of memory, and the third level entailing non-stop reading of the entire memory array. Read operations were performed at both +25°C and +125°C and with a supply voltage of 3.7V, which exceeds the specified datasheet operating voltage range. The results of all of the extensive tests indicate that the contents of a portion of memory being read continuously could be altered after 800,000,000 read operations only if that portion of the memory was not erased or reprogrammed at all during the 800,000,000 read operations. If that portion of memory was reprogrammed at some point, then it would take another 800,000,000

read operations after reprogramming before the contents could potentially be altered. For example, if the Serial Flash is being used for boot code storage, then it would take 800,000,000 boot operations before that boot code may become altered, provided that the boot code was not updated or reprogrammed. If an application was to read the entire memory array non-stop at a clock frequency of 10MHz, it would take over five years to reach 800,000,000 read operations. Atmel® firmly believes that this extended testing result should not be a cause for concern. We also believe that most, if not all, applications will never read the same portion of memory 800,000,000 times throughout the life of the application without ever updating that portion of memory.

16. Electrical Specifications

Temperature under Bias.....	-55°C to +125°C
Storage Temperature	-65°C to +150°C
All Input Voltages (except V_{CC} but including NC pins) with Respect to Ground.....	-0.6V to +6.25V
All Output Voltages with Respect to Ground.....	-0.6V to $V_{CC} + 0.6V$

***NOTICE:** Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. The "Absolute Maximum Ratings" are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Voltage Extremes referenced in the "Absolute Maximum Ratings" are intended to accommodate short duration undershoot/overshoot conditions and does not imply or guarantee functional device operation at these levels for any extended period of time

Table 16-1. DC and AC Operating Range

		Atmel AT45DB021D
Operating Temperature (Case)	Ind.	-40°C to 85°C
V_{CC} Power Supply		2.7V to 3.6V

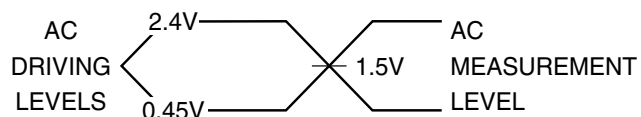
Table 16-2. DC Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
I_{DP}	Deep Power-down Current	\overline{CS} , \overline{RESET} , $\overline{WP} = V_{IH}$, all inputs at CMOS levels		15	25	μA
I_{SB}	Standby Current	\overline{CS} , \overline{RESET} , $\overline{WP} = V_{IH}$, all inputs at CMOS levels		25	50	μA
$I_{CC1}^{(1)}$	Active Current, Read Operation	$f = 20MHz$; $I_{OUT} = 0mA$; $V_{CC} = 3.6V$		7	10	mA
		$f = 33MHz$; $I_{OUT} = 0mA$; $V_{CC} = 3.6V$		8	12	mA
		$f = 50MHz$; $I_{OUT} = 0mA$; $V_{CC} = 3.6V$		10	14	mA
		$f = 66MHz$; $I_{OUT} = 0mA$; $V_{CC} = 3.6V$		11	15	mA
I_{CC2}	Active Current, Program/Erase Operation	$V_{CC} = 3.6V$		12	17	mA
I_{LI}	Input Load Current	$V_{IN} = \text{CMOS levels}$			1	μA
I_{LO}	Output Leakage Current	$V_{IO} = \text{CMOS levels}$			1	μA
V_{IL}	Input Low Voltage				$V_{CC} \times 0.3$	V
V_{IH}	Input High Voltage		$V_{CC} \times 0.7$			V
V_{OL}	Output Low Voltage	$I_{OL} = 1.6mA$; $V_{CC} = 2.7V$			0.4	V
V_{OH}	Output High Voltage	$I_{OH} = -100\mu A$	$V_{CC} - 0.2V$			V

- Notes: 1. I_{CC1} during a buffer read is 20mA maximum @ 20MHz
2. All inputs (SI, SCK, CS#, WP#, and RESET#) are guaranteed by design to be 5V tolerant

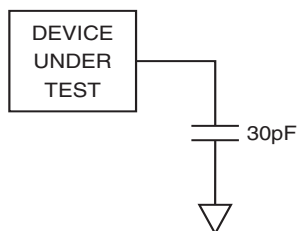
Table 16-3. AC Characteristics – Atmel RapidS/Serial Interface

Symbol	Parameter	Min	Typ	Max	Units
f_{SCK}	SCK Frequency			66	MHz
f_{CAR1}	SCK Frequency for Continuous Array Read			66	MHz
f_{CAR2}	SCK Frequency for Continuous Array Read (Low Frequency)			33	MHz
t_{WH}	SCK High Time	6.8			ns
t_{WL}	SCK Low Time	6.8			ns
$t_{SCKR}^{(1)}$	SCK Rise Time, Peak-to-Peak (Slew Rate)	0.1			V/ns
$t_{SCKF}^{(1)}$	SCK Fall Time, Peak-to-Peak (Slew Rate)	0.1			V/ns
t_{CS}	Minimum \overline{CS} High Time	50			ns
t_{CSS}	\overline{CS} Setup Time	5			ns
t_{CSH}	\overline{CS} Hold Time	5			ns
t_{SU}	Data In Setup Time	2			ns
t_H	Data In Hold Time	3			ns
t_{HO}	Output Hold Time	0			ns
t_{DIS}	Output Disable Time		27	35	ns
t_V	Output Valid			6	ns
t_{WPE}	\overline{WP} Low to Protection Enabled			1	μ s
t_{WPD}	\overline{WP} High to Protection Disabled			1	μ s
t_{EDPD}	\overline{CS} High to Deep Power-down Mode			3	μ s
t_{RDPD}	\overline{CS} High to Standby Mode			35	μ s
t_{XFR}	Page to Buffer Transfer Time			200	μ s
t_{comp}	Page to Buffer Compare Time			200	μ s
t_{EP}	Page Erase and Programming Time (256-/264-bytes)		14	35	ms
t_P	Page Programming Time (256-/264-bytes)		2	4	ms
t_{PE}	Page Erase Time (256-/264-bytes)		13	32	ms
t_{BE}	Block Erase Time (2,048-2,112-bytes)		15	35	ms
t_{SE}	Sector Erase Time (32,768-/33,792-bytes)		400	700	ms
t_{CE}	Chip Erase Time		3.6	6	s
t_{RST}	RESET Pulse Width	10			μ s
t_{REC}	RESET Recovery Time			1	μ s

Figure 16-1. Input Test Waveforms and Measurement Levels

$t_R, t_F < 2ns$ (10% to 90%)

Figure 16-2. Output Test Load



17. AC Waveforms

Six different timing waveforms are shown on [page 32](#). Waveform 1 shows the SCK signal being low when \overline{CS} makes a high-to-low transition, and waveform 2 shows the SCK signal being high when \overline{CS} makes a high-to-low transition. In both cases, output SO becomes valid while the SCK signal is still low (SCK low time is specified as t_{WL}). Timing waveforms 1 and 2 conform to Atmel® RapidS™ serial interface but for frequencies up to 66MHz. Waveforms 1 and 2 are compatible with SPI Mode 0 and SPI Mode 3, respectively.

Waveform 3 and waveform 4 illustrate general timing diagram for RapidS serial interface. These are similar to waveform 1 and waveform 2, except that output SO is not restricted to become valid during the t_{WL} period. These timing waveforms are valid over the full frequency range (maximum frequency = 66MHz) of the RapidS serial case.

Figure 17-1. Waveform 1 – SPI Mode 0 Compatible (for Frequencies up to 66MHz)

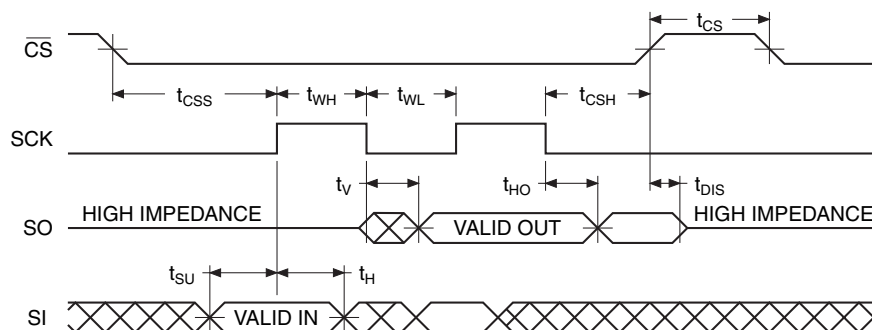


Figure 17-2. Waveform 2 – SPI Mode 3 Compatible (for Frequencies up to 66MHz)

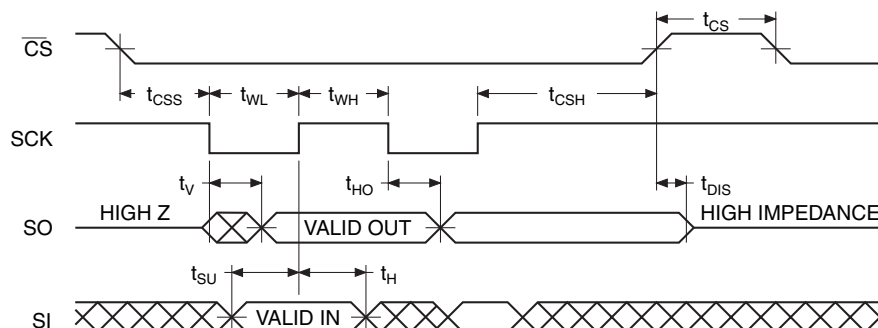


Figure 17-3. Waveform 3 – Atmel RapidS Mode 0 ($F_{MAX} = 66\text{MHz}$)

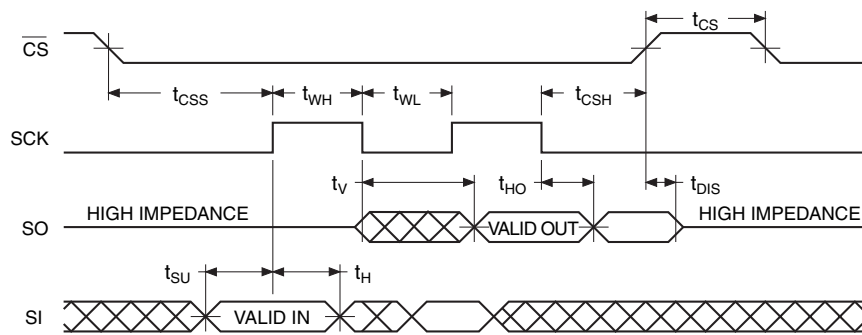
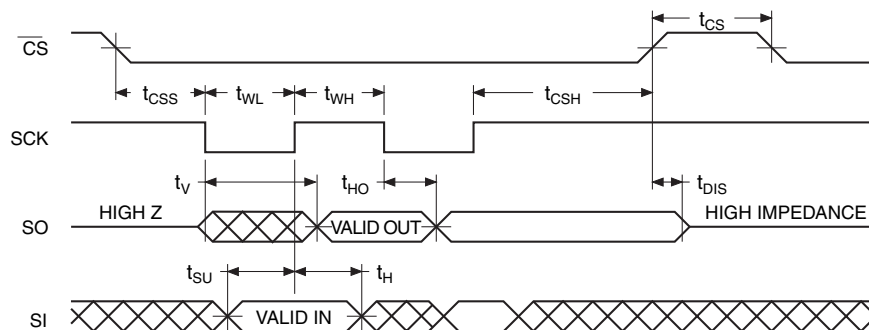


Figure 17-4. Waveform 4 – Atmel RapidS Mode 3 ($F_{MAX} = 66\text{MHz}$)

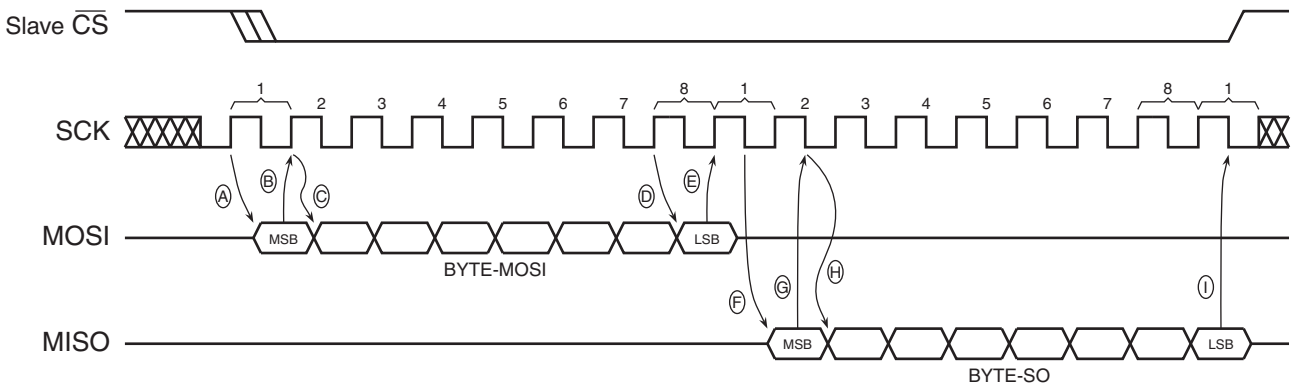


17.1 Utilizing the Atmel RapidS Function

To take advantage of the Atmel® RapidS™ function's ability to operate at higher clock frequencies, a full clock cycle must be used to transmit data back and forth across the serial bus. The Atmel DataFlash® is designed to always clock its data out on the falling edge of the SCK signal and clock data in on the rising edge of SCK.

For full clock cycle operation to be achieved, when the DataFlash is clocking data out on the falling edge of SCK, the host controller should wait until the next falling edge of SCK to latch the data in. Similarly, the host controller should clock its data out on the rising edge of SCK in order to give the DataFlash a full clock cycle to latch the incoming data in on the next rising edge of SCK.

Figure 17-5. Atmel RapidS Mode



MOSI = Master Out, Slave In

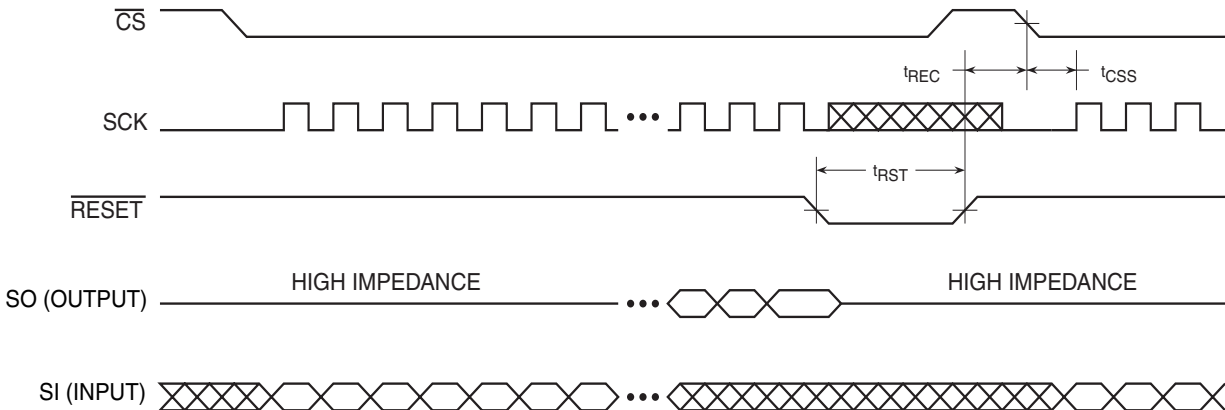
MISO = Master In, Slave Out

The Master is the host controller and the Slave is the DataFlash

The Master always clocks data out on the rising edge of SCK and always clocks data in on the falling edge of SCK.
The Slave always clocks data out on the falling edge of SCK and always clocks data in on the rising edge of SCK.

- Master clocks out first bit of BYTE-MOSI on the rising edge of SCK
- Slave clocks in first bit of BYTE-MOSI on the next rising edge of SCK
- Master clocks out second bit of BYTE-MOSI on the same rising edge of SCK
- Last bit of BYTE-MOSI is clocked out from the Master
- Last bit of BYTE-MOSI is clocked into the slave
- Slave clocks out first bit of BYTE-SO
- Master clocks in first bit of BYTE-SO
- Slave clocks out second bit of BYTE-SO
- Master clocks in last bit of BYTE-SO

Figure 17-6. Reset Timing



Note: The \overline{CS} signal should be in the high state before the \overline{RESET} signal is deasserted

Figure 17-7. Command Sequence for Read/Write Operations for Page Size 256-Bytes (Except Status Register Read, Manufacturer and Device ID Read)

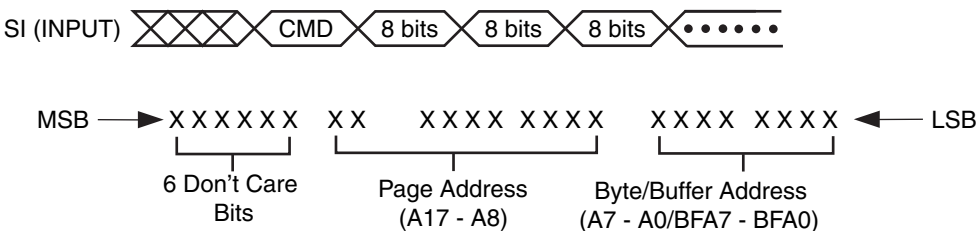


Figure 17-8. Command Sequence for Read/Write Operations for Page Size 264-Bytes (Except Status Register Read, Manufacturer and Device ID Read)

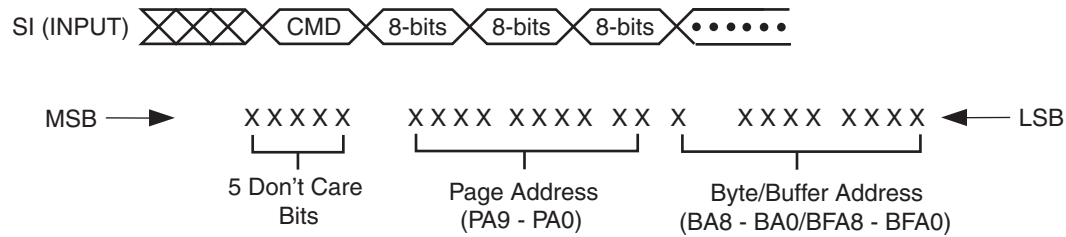


Figure 17-9. Write Operations

The following block diagram and waveforms illustrate the various write sequences available

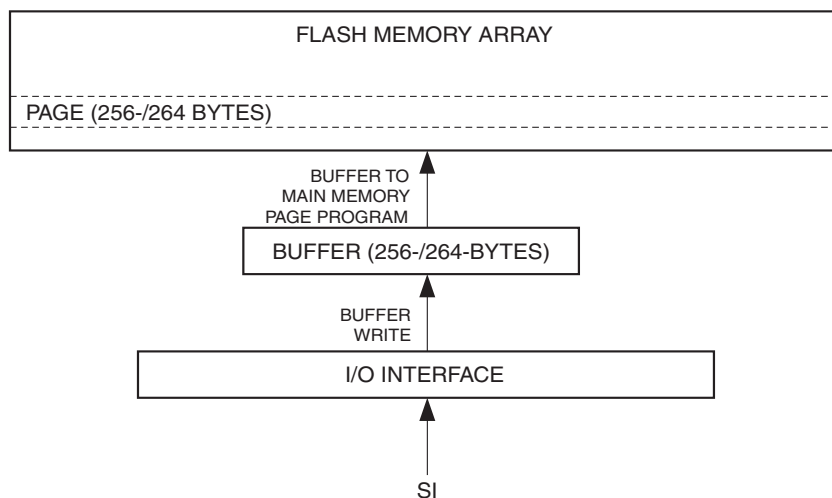


Figure 17-10. Buffer Write

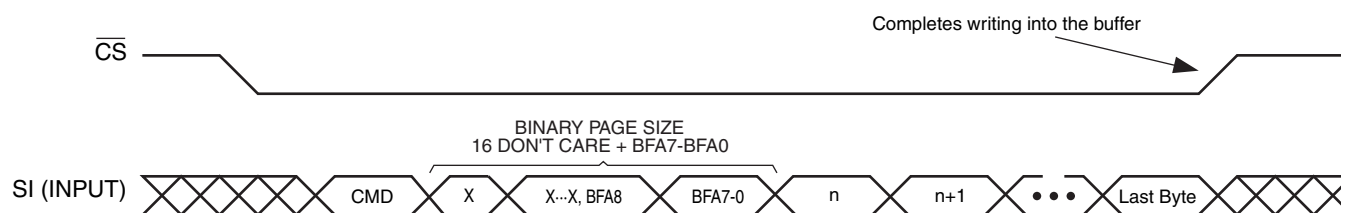
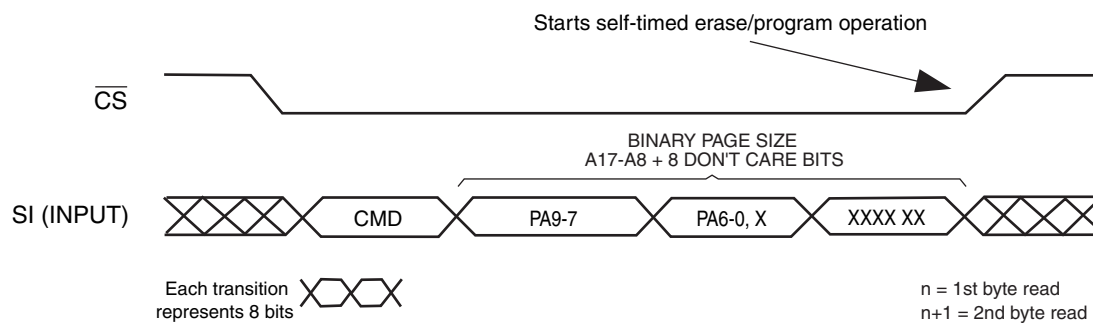


Figure 17-11. Buffer to Main Memory Page Program (Data from Buffer Programmed into Flash Page)



18. Read Operations

The following block diagram and waveforms illustrate the various read sequences available.

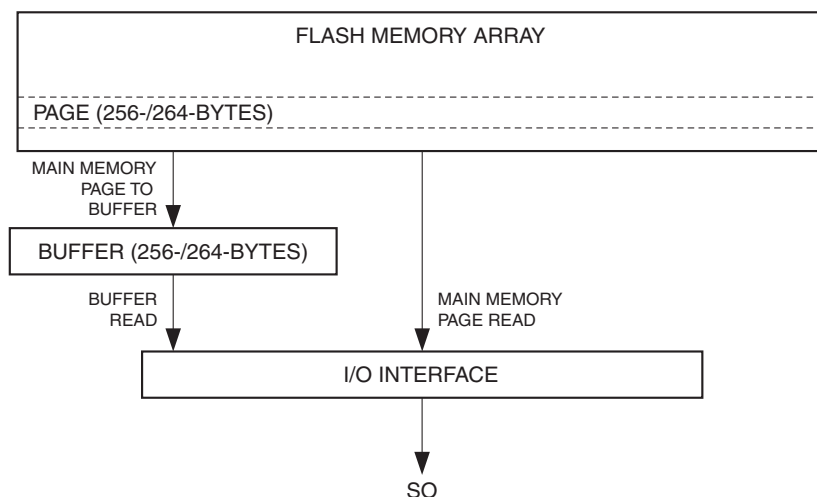
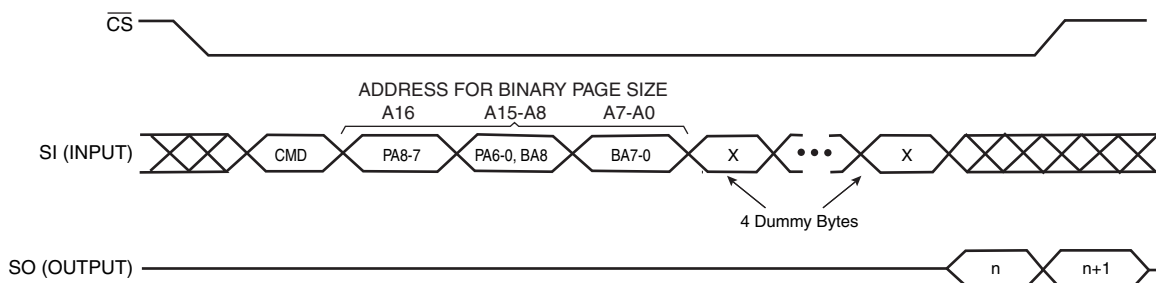


Figure 18-1. Main Memory Page Read



Timing diagram for the read command sequence:

- CS (Chip Select):** Active-low signal. It transitions from high to low at the start of the command sequence and returns to high at the end. An arrow points to the rising edge with the text "Starts reading page data into buffer".
- SI (INPUT):** Shows the command sequence:
 - 4-bit address field: X, X, X, X
 - Command: CMD
 - 6-bit address field: X...X, PA9-7
 - 16-bit address field: PA6-0, X
 - 16-bit data field: XXXX XXXX (labeled "BINARY PAGE SIZE" and "6 DON'T CARE BITS + A17-A8 + 8 DON'T CARE BITS")
- SO (OUTPUT):** Shown as a horizontal line, indicating it is not active during this sequence.

The diagram illustrates the timing of the SPI interface signals:

- CS (Chip Select):** A signal that transitions from high to low to initiate a transfer and returns to high at the end.
- SI (INPUT):** The serial input signal. It shows a sequence of bytes: CMD, X, X..X, BFA8, BFA7-0, X, and a 1 Dummy Byte. The address field (X..X, BFA8, BFA7-0) is labeled as "BINARY PAGE SIZE 16 DON'T CARE + BFA7-BFA0".
- SO (OUTPUT):** The serial output signal. It shows data output starting at address n and n+1.

Each transition represents 8 bits.

Figure 19-1. Continuous Array Read (Legacy Opcode E8H)

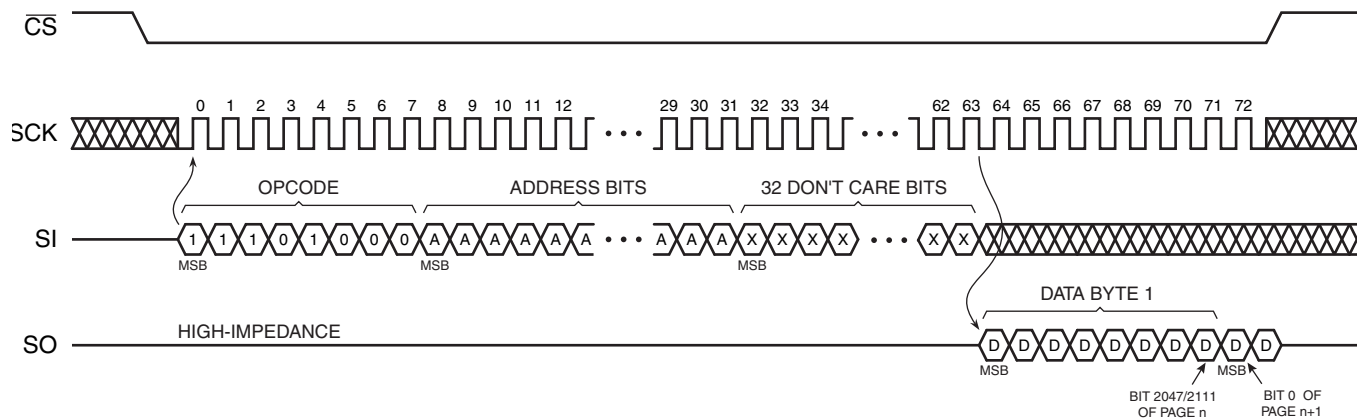


Figure 19-2. Continuous Array Read (Opcode 0BH)

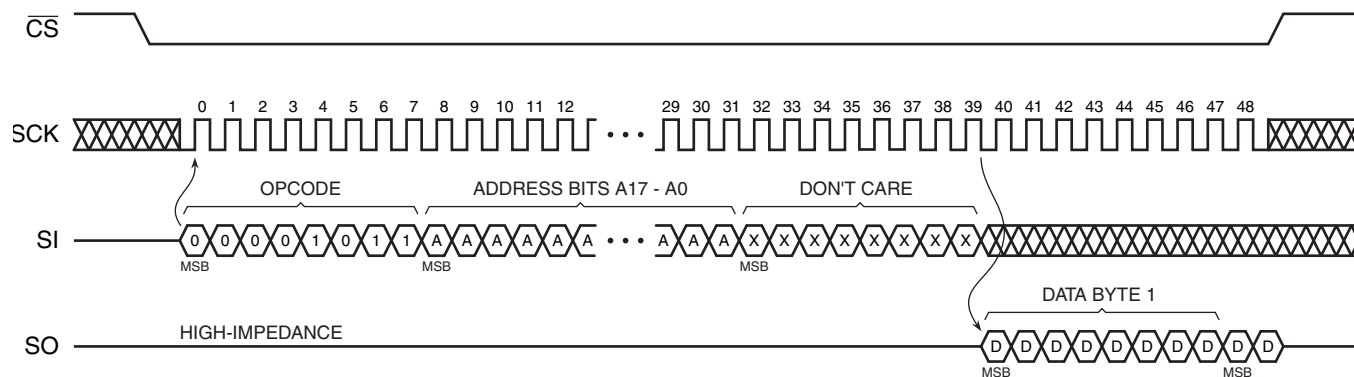


Figure 19-3. Continuous Array Read (Low Frequency: Opcode 03H)

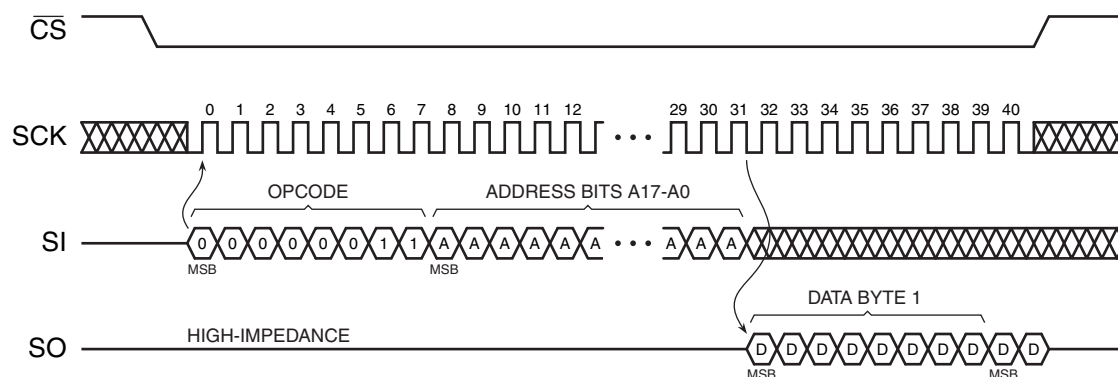


Figure 19-4. Main Memory Page Read (Opcode: D2H)

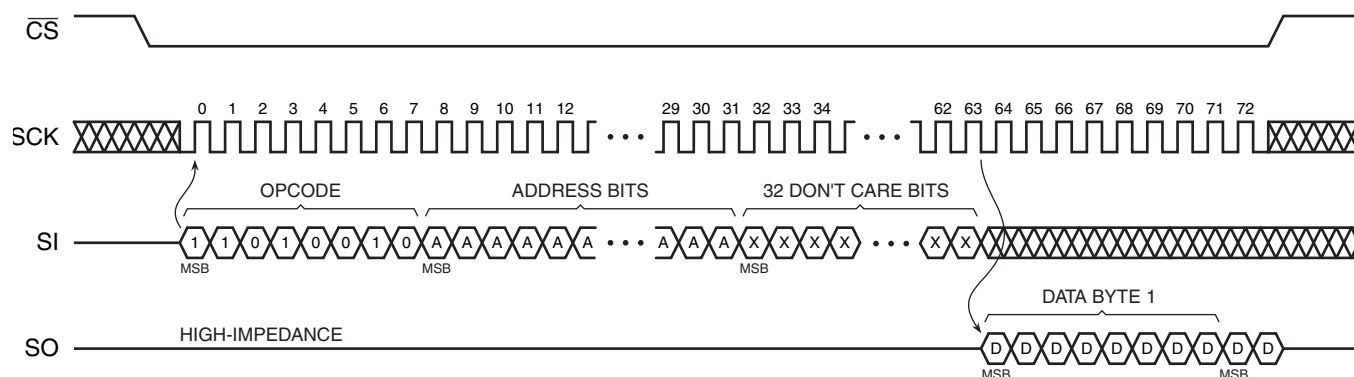


Figure 19-5. Buffer Read (Opcode D4H)

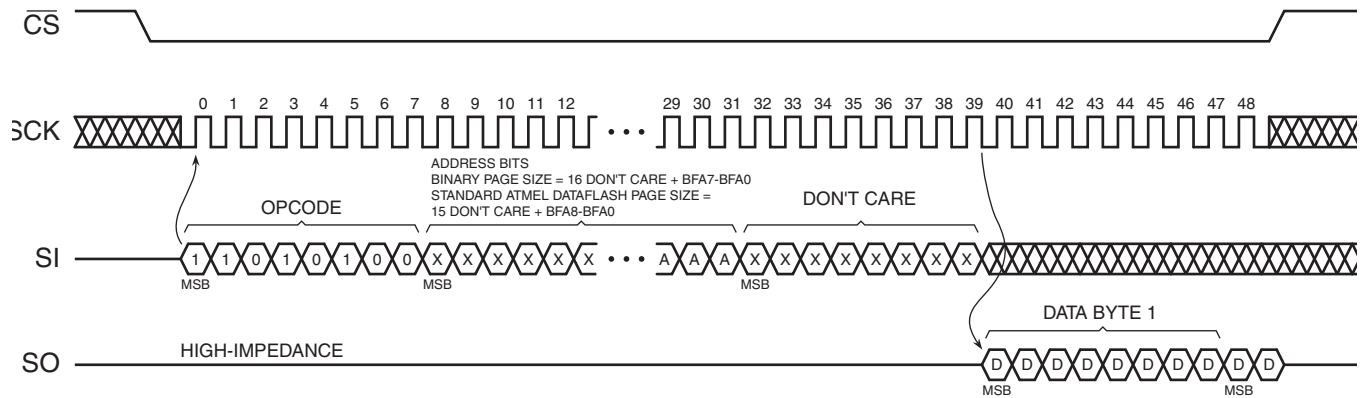


Figure 19-6. Buffer Read (Low Frequency: Opcode D1H)

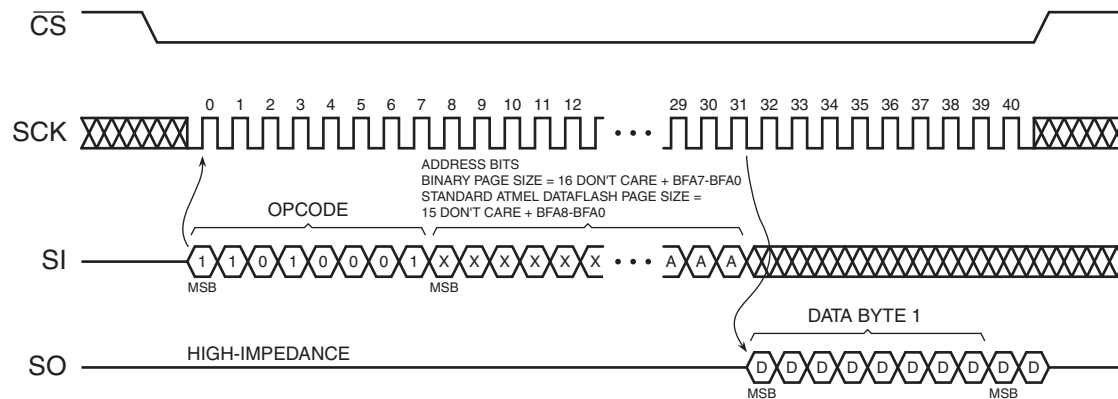
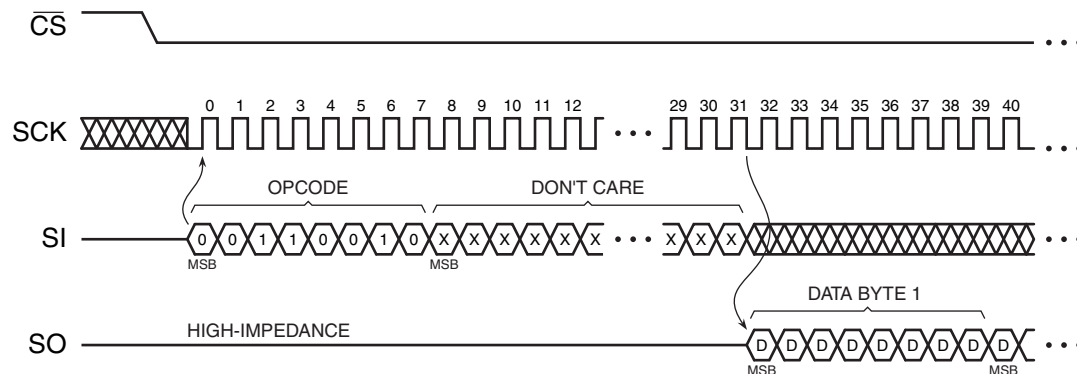


Figure 19-7. Read Sector Protection Register (Opcode 32H)



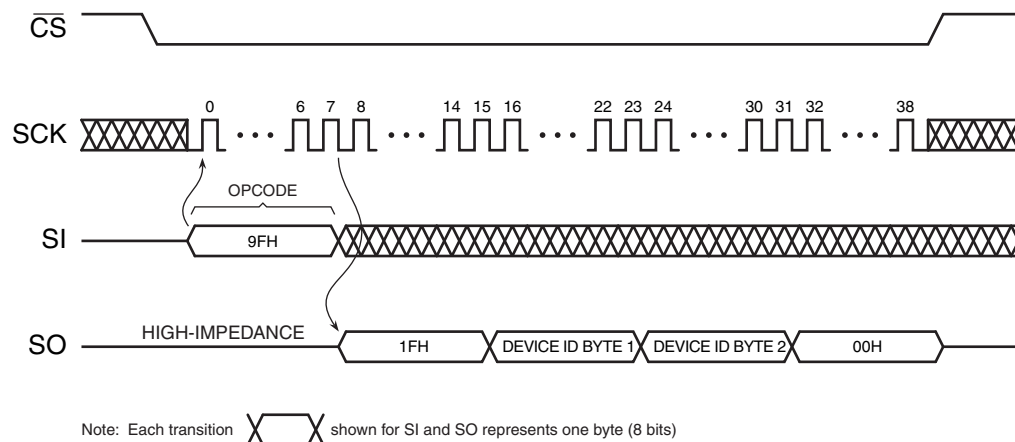
The diagram illustrates the timing of an SPI interface. The **CS** (Chip Select) signal is active-low, indicated by a bar over the label. The **SCK** (Serial Clock) signal is a periodic clock. The **SI** (Serial Input) signal shows the data being sent to the device. The first 8 bits of the SI signal are labeled **OPCODE**, and the following bits are labeled **DON'T CARE**. The **SO** (Serial Output) signal is shown in a high-impedance state until the first data byte is received. The first data byte is labeled **DATA BYTE 1**, and its MSB (Most Significant Bit) is shown on the SO signal. The diagram also shows the MSB (Most Significant Bit) of the opcode and the first data byte on the SI signal.

The diagram illustrates the timing of an SPI transaction. The **CS** (Chip Select) signal is active-low, indicated by a bar over the label. It transitions from high to low at the start of the transaction and returns to high at the end. The **SCK** (Serial Clock) signal is a periodic square wave. The **SI** (Serial Input) signal shows the data being sent to the device. The first 8 bits are the **OPCODE** (00001111), followed by several **DON'T CARE** bytes (represented by X's). The **SO** (Serial Output) signal is in a **HIGH-IMPEDANCE** state until the first data byte is received. This **DATA BYTE 1** is shown as **DDDDDDDD** (represented by D's). The diagram also shows subsequent **DON'T CARE** bytes on the SI line and the continuation of the SCK signal.

The diagram illustrates the timing of the SPI interface signals:

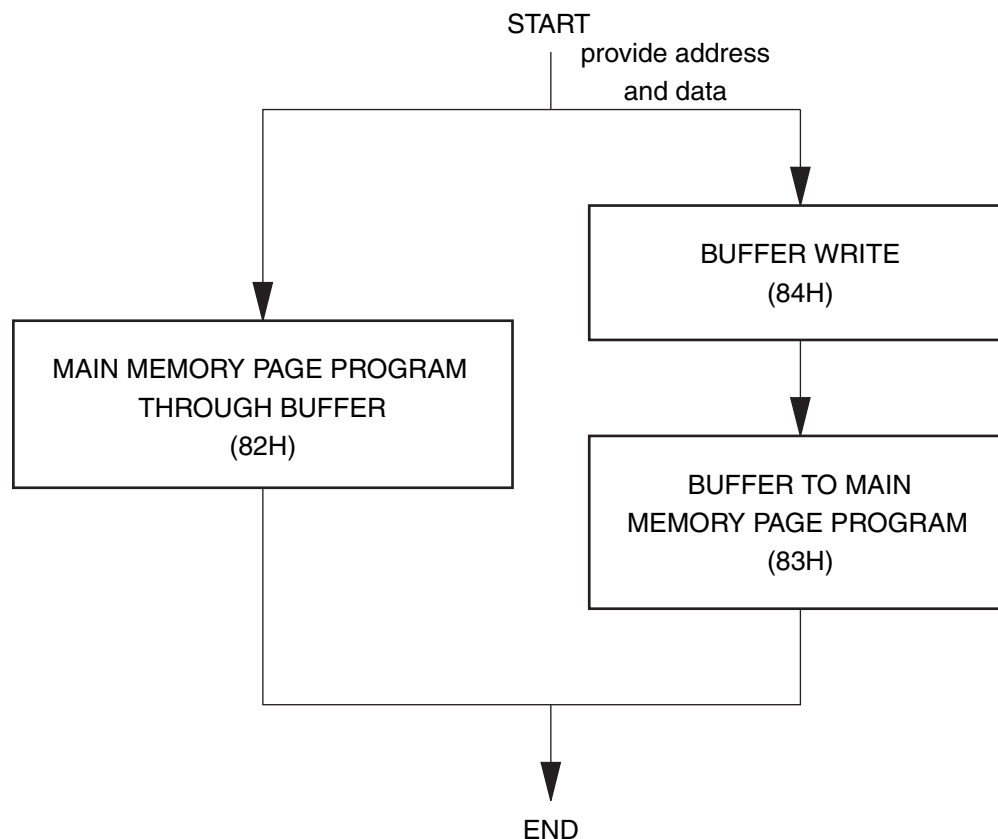
- CS (Chip Select):** Active-low signal, shown as a pulse that goes low during the data transfer.
- SCK (Serial Clock):** A periodic clock signal used for data transfer.
- SI (Serial Input):** The input data stream. It starts with an **OPCODE** (1101011) and is followed by **STATUS REGISTER DATA**. The first bit is labeled **MSB**.
- SO (Serial Output):** The output data stream. It is in a **HIGH-IMPEDANCE** state during the opcode phase and then outputs **STATUS REGISTER DATA**. The first bit is labeled **MSB**.

Figure 19-11. Manufacturer and Device Read (Opcode 9FH)



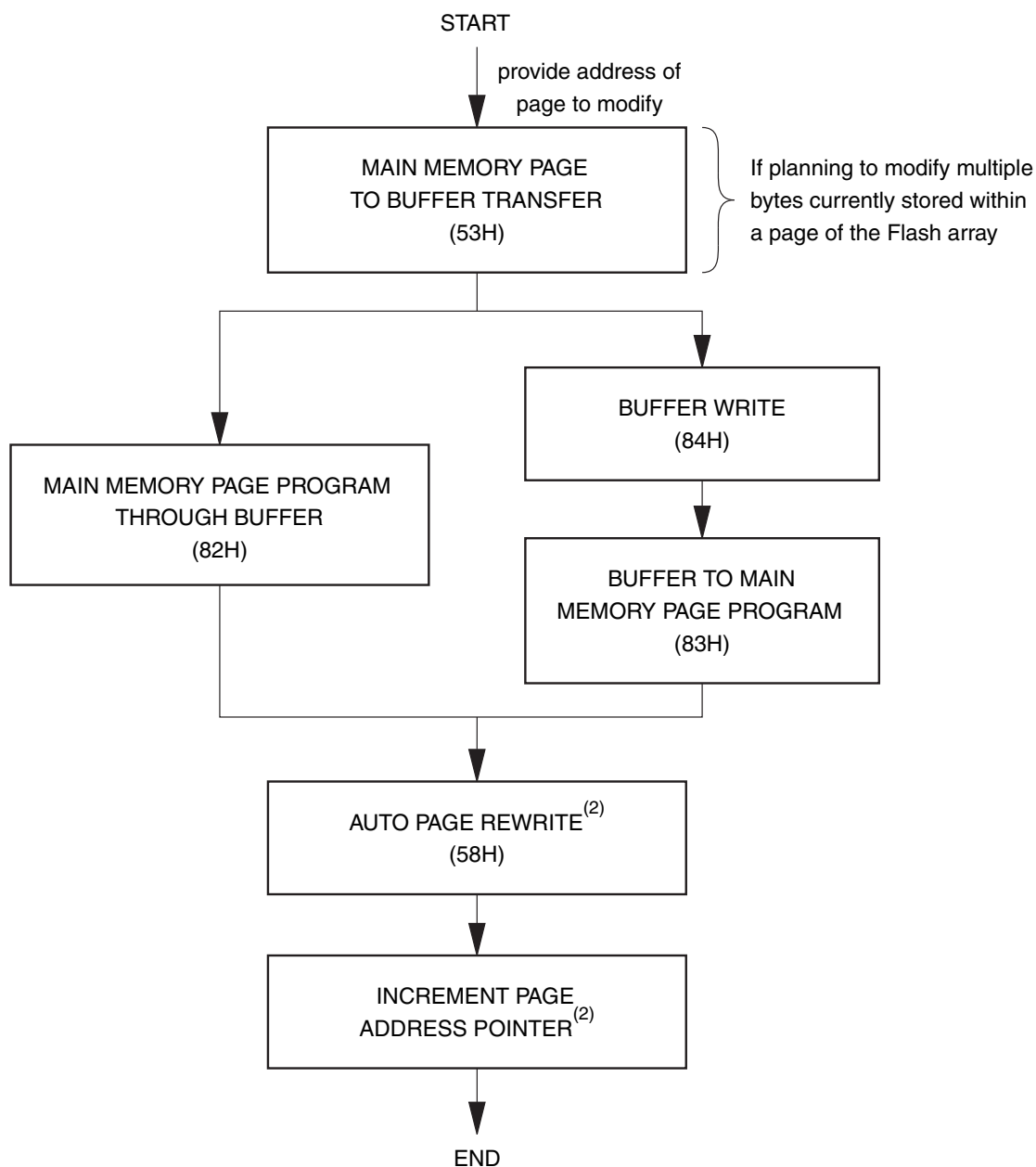
20. Auto Page Rewrite Flowchart

Figure 20-1. Algorithm for Programming or Reprogramming of the Entire Array Sequentially



- Notes:
1. This type of algorithm is used for applications in which the entire array is programmed sequentially, filling the array page-by-page
 2. A page can be written using either a Main Memory Page Program operation or a Buffer Write operation followed by a Buffer to Main Memory Page Program operation
 3. The algorithm above shows the programming of a single page. The algorithm will be repeated sequentially for each page within the entire array

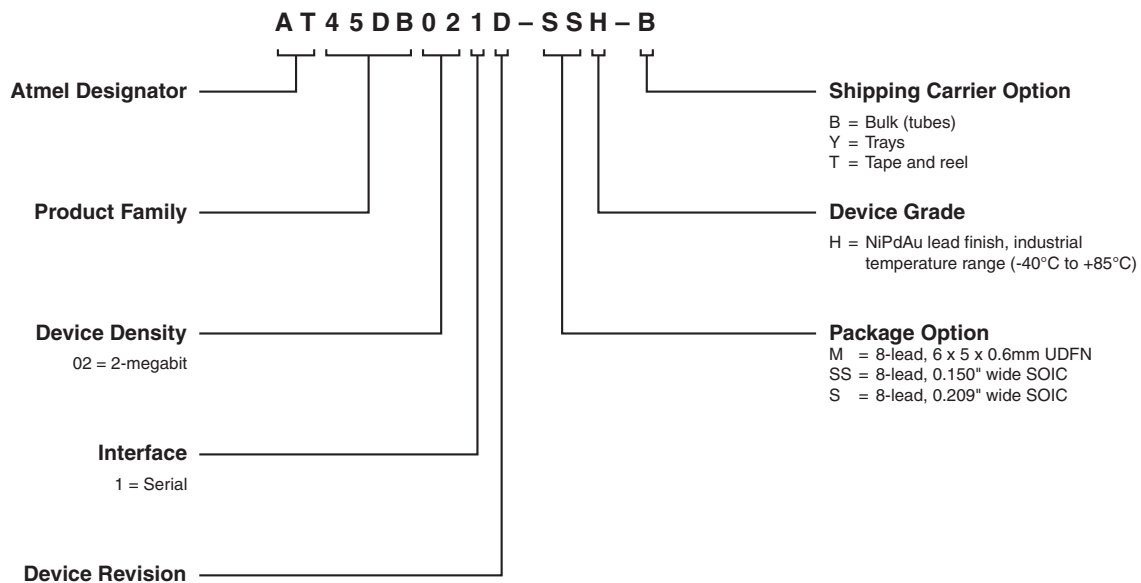
Figure 20-2. Algorithm for Randomly Modifying Data



- Notes:
1. To preserve data integrity, each page of an Atmel DataFlash sector must be updated/rewritten at least once within every 10,000 cumulative page erase and program operations
 2. A Page Address Pointer must be maintained to indicate which page is to be rewritten. The Auto Page Rewrite command must use the address specified by the Page Address Pointer
 3. Other algorithms can be used to rewrite portions of the Flash array. Low-power applications may choose to wait until 10,000 cumulative page erase and program operations have accumulated before rewriting all pages of the sector. See application note AN-4 ("Using Atmel's Serial DataFlash") for more details

21. Ordering Information

21.1 Ordering Code Detail



21.2 Green Package Options (Pb/Halide-free/RoHS Compliant)

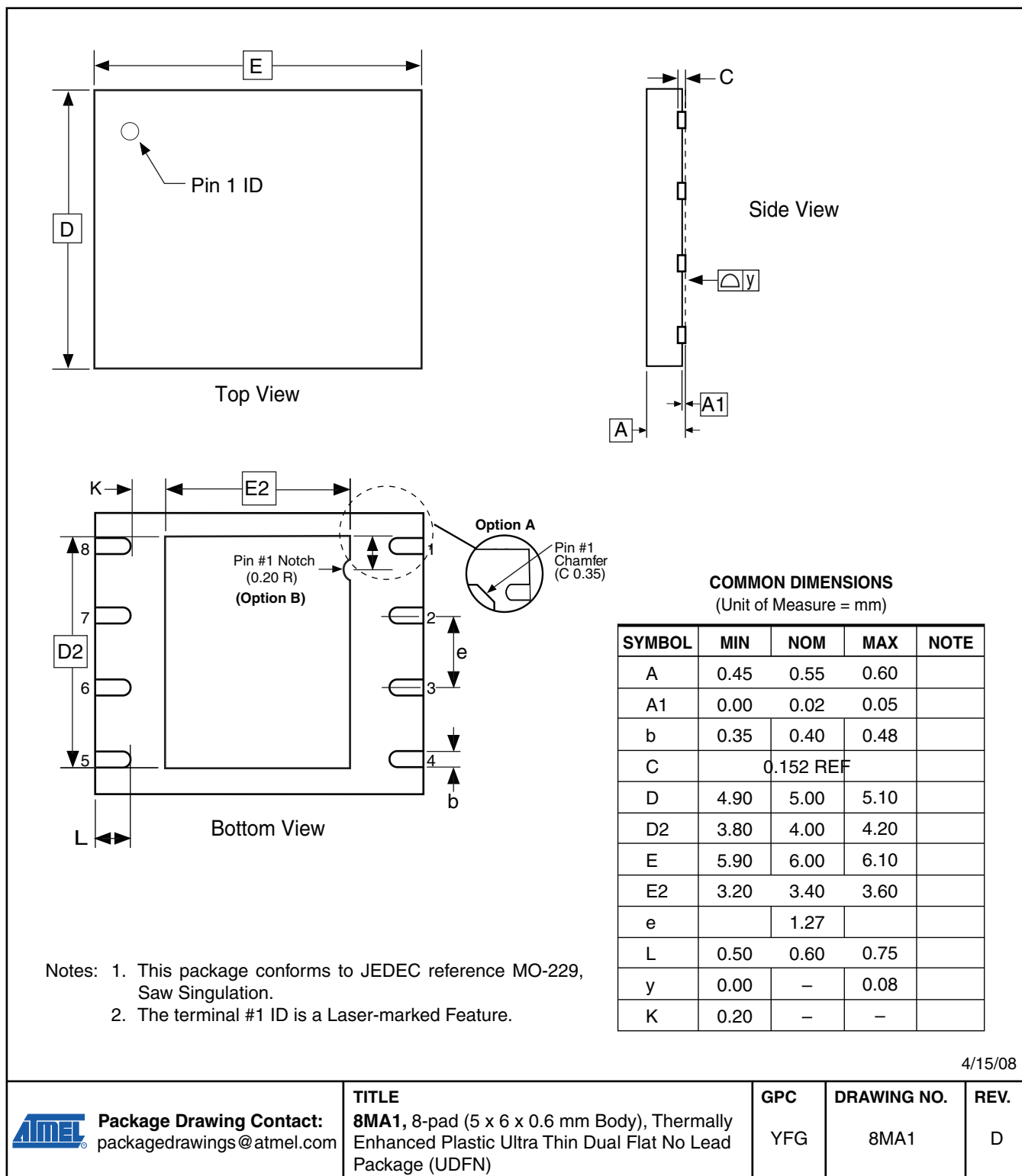
Ordering Code ⁽¹⁾⁽²⁾	Package	Lead Finish	Operating Voltage	f _{SK} (MHz)	Operation Range
AT45DB021D-MH-Y AT45DB021D-MH-T AT45DB021D-MH-SL954 ⁽³⁾ AT45DB021D-MH-SL955 ⁽⁴⁾	8MA1	NiPdAu	2.7V to 3.6V	66	Industrial (-40°C to +85°C)
AT45DB021D-SSH-B AT45DB021D-SSH-T AT45DB021D-SSH-SL954 ⁽³⁾ AT45DB021D-SSH-SL955 ⁽⁴⁾	8S1				
AT45DB021D-SH-B AT45DB021D-SH-T AT45DB021D-SH-SL954 ⁽³⁾ AT45DB021D-SH-SL955 ⁽⁴⁾	8S2				

- Notes:
- The shipping carrier option is not marked on the devices
 - Standard parts are shipped with the page size set to 264-bytes. The user is able to configure these parts to a 256-byte page size if desired
 - Parts ordered with suffix SL954 are shipped in bulk with the page size set to 256-bytes. Parts will have a 954 or SL954 marked on them
 - Parts ordered with suffix SL955 are shipped in tape and reel with the page size set to 256-bytes. Parts will have a 954 or SL954 marked on them

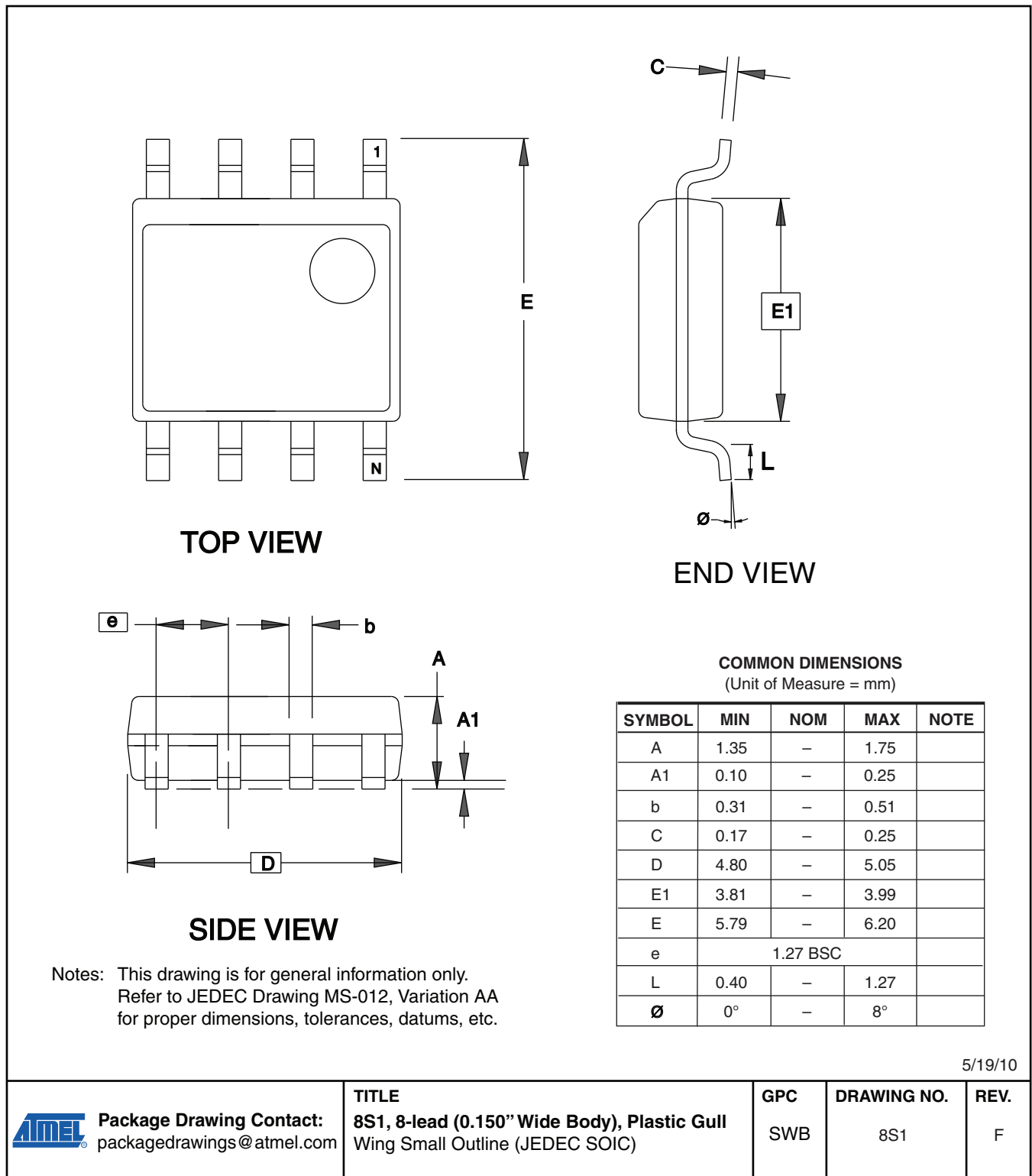
Package Type	
8MA1	8-lead (6 x 5 x 0.6mm Body), Thermally Enhanced Plastic Ultra Thin Dual Flat No Lead Package (UDFN)
8S1	8-lead, 0.150" Wide, Plastic Gull Wing Small Outline Package (JEDEC SOIC)
8S2	8-lead, 0.209" Wide, Plastic Gull Wing Small Outline Package (EIAJ SOIC)

22. Packaging Information

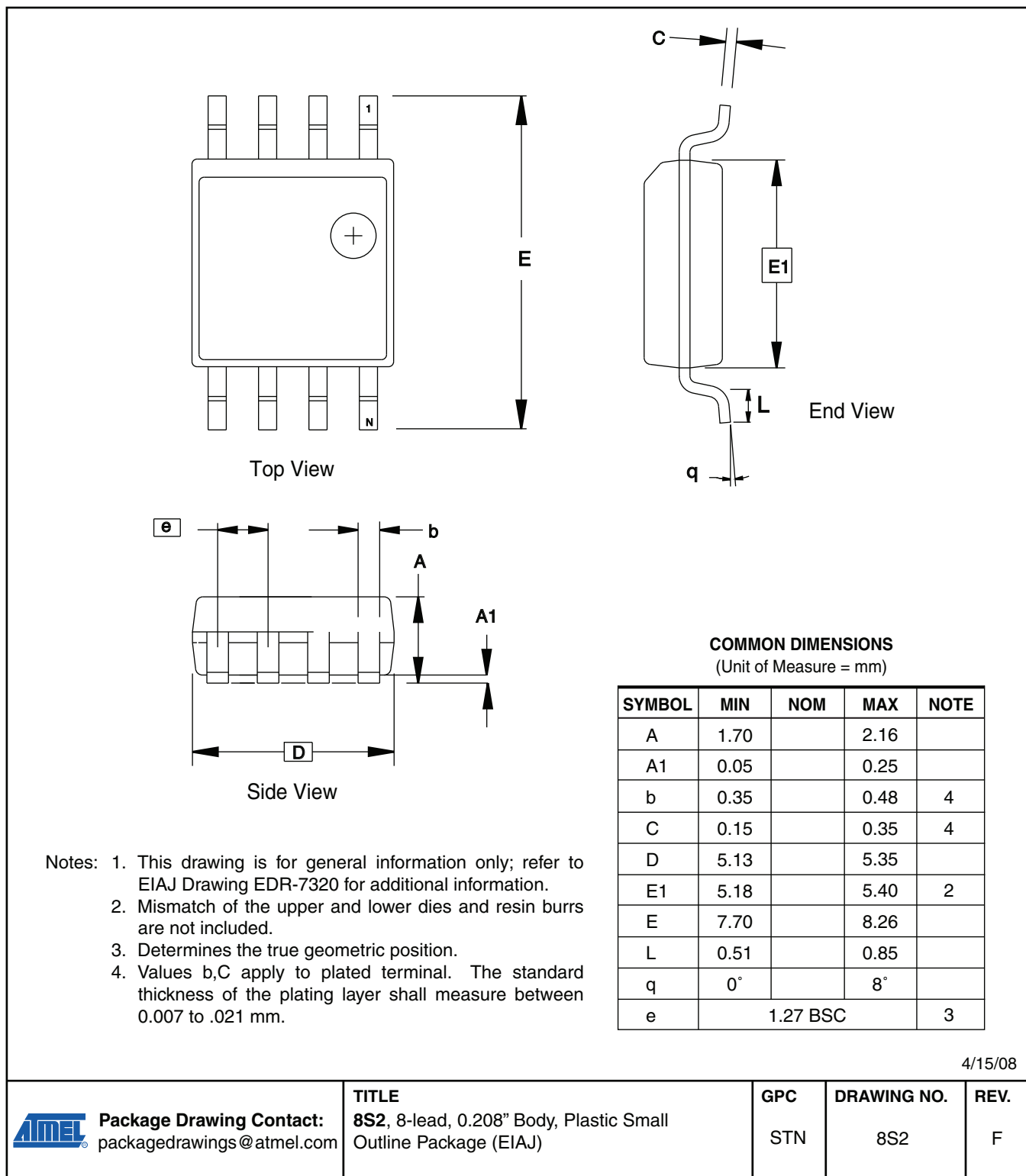
22.1 8MA1 – UDFN



22.2 8S1 – JEDEC SOIC



22.3 8S2 – EIAJ SOIC



23. Revision History

Doc. Rev.	Date	Comments
A	06/2006	Initial release.
B	02/2007	Removed RDY/BUS \overline{Y} pin references.
C	08/2007	Changed t_{VCSL} time to 1ms Changed I_{DP} (Max) to 15 μ A Added Chip Erase time Changed t_{RDPD} time to 35 μ s Fixed the typographical error in the Block Architecture diagram
D	11/2007	Changed the t_{XFR} and t_{COMP} times from 400 μ s to 200 μ s Changed part number ordering code to reflect NiPdAu lead finish <ul style="list-style-type: none"> - Changed AT45DB021D-MU to AT45DB021D-MH - Changed AT45DB021D-SSU to AT45DB021D-SSH - Changed AT45DB021D-SU to AT45DB021D-SH Added lead finish details to Ordering Information table Added Ordering Code Detail
E	02/2008	Fixed the typographical error, under Status Register Read, to indicate that bit 3 is a "0"
F	04/2008	Replaced 8M1-A MLF Package with 8MA1 UDFN Package Added part number ordering code details for suffixes SL954/955
G	02/2009	Changed t_{DIS} (Typ and Max) to 27ns and 35ns, respectively
H	03/2009	Changed Deep Power-Down Current values <ul style="list-style-type: none"> - Increased typical value from 5μA to 15μA - Increased maximum value from 15μA to 25μA
I	04/2009	Updated Absolute Maximum Ratings Updated System Specifications
J	05/2010	Updated template Changed number of bytes and sectors in Section 7.1.2 on page 13 Changed T_{SE} values in Table 16-3 on page 31 <ul style="list-style-type: none"> - Typ from 0.8 to 400, Max from 2.5 to 700 and Units from s to ms Changed BA0 to PA0 and x to P under PA3, row 50h in Table 13-7 on page 27 Changed A11 from x to P, row 50h in Table 13-6 on page 26 Changed from 10,000 to 20,000 cumulative page erase/program operations in Section 9.3 Added "Please contact Atmel for availability of devices that are specified to exceed the 20K cycle cumulative limit" in Section 9.3



Headquarters

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

International

Atmel Asia Limited
Unit 01-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parking 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
JAPAN
Tel: (+81) 3-3523-3551
Fax: (+81) 3-3523-7581

Product Contact

Technical Support
dataflash@atmel.com

Sales Contact
www.atmel.com/contacts

Literature Requests
www.atmel.com/literatur

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. **EXCEPT AS SET FORTH IN ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© 2010 Atmel Corporation. All rights reserved. Atmel®, Atmel logo and combinations thereof, Everywhere You Are®, DataFlash®, RapidS™ logo and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.