

### FEATURES

- **Compatible with Serial Peripheral Interface (SPI)**
- **Supports SPI Modes 0 and 3**
- **66MHz clock rate**
- **Block Write Protection**
- **Secure WRITE**
- **Secure READ**
- **2-Byte User Serial Number**
- **Hibernate Mode for low Standby Current**
- **Page and Block Rollover options**
- **Unlimited Read/Write Endurance**
- **Non-Volatile STORE under Instruction Control**
- **Automatic RECALL to SRAM on Power Up**
- **Unlimited RECALL Cycles**
- **100k STORE Cycles**
- **100-Year Non-volatile Data Retention**
- **3.0V to 3.6V Power Supply**
- **Commercial and Industrial Temperatures**
- **8-pin 150 mil SOIC Package**
- **RoHS-Compliant**

### DESCRIPTION

The Anvo-Systems Dresden ANV31A81A is a 256Kb serial SRAM with a non-volatile SONOS storage element included with each memory cell, organized as 32k words of 8 bits each. The devices are accessed by a high speed SPI-compatible bus. The ANV31A81A is enabled through the Chip Enable pin ( $\overline{E}$ ) and accessed via a 3-wire interface consisting of Serial Data Input (SI), Serial Data Output (SO) and Serial Clock (SCK). All programming cycles are self-timed, and no separate ERASE cycle is required before STORE.

The serial SRAM provides the fast access & cycle times, ease of use and unlimited read & write endurance of a normal SRAM. Dedicated safety features supporting high data accuracy.

With Secure WRITE operation the ANV31A81A accepts address and data only when the correct 2 Byte CRC, generated from the 15 bit address and 64 Byte data, is transmitted. Corrupt data cannot overwrite existing memory content and even valid data would not overwrite on a corrupted address. With status register bit 4 the success of the Secure WRITE operation can be monitored. In case of corrupt data bit 4 will be set volatile to high. With Secure READ operation the ANV31A81A calculates the correct 2-Byte CRC parallel to data transfer. The 2-Byte CRC is transmitted after 64 Bytes of data have been transmitted.

Both STORE and RECALL operations are available under instruction control.

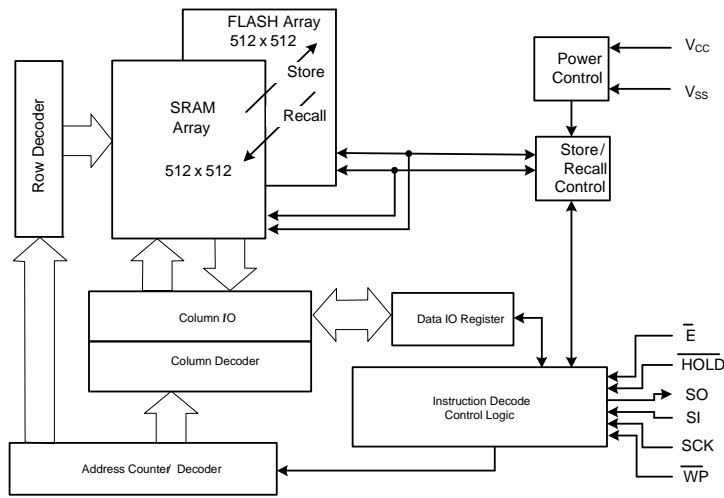
On power up, data is automatically restored to the SRAM (the Power Up Recall operation).

BLOCK WRITE Protection is enabled by programming the status register with one of four options to protect blocks.

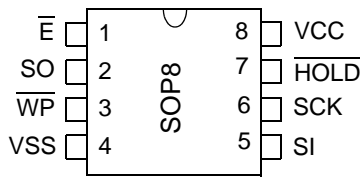
A 2-Byte non-volatile register supports the option of a 2-Byte user defined serial number. This register is under customer control only.

Status register bit 5 will control page and block roll over modes.

BLOCK DIAGRAM



PIN CONFIGURATION



Top View

8-pin SOP 150 mil

PIN DESCRIPTIONS

| Signal Name       | Signal Description           |
|-------------------|------------------------------|
| $\bar{E}$         | Chip Enable                  |
| SCK               | Serial Clock                 |
| SI                | Serial Input                 |
| SO                | Serial Output                |
| $\overline{HOLD}$ | Hold (Suspends Serial Input) |
| $\overline{WP}$   | Write Protect                |
| VCC               | Power Supply Voltage         |
| VSS               | Ground                       |

Serial Interface Description

**Master:** The device that generates the serial clock.

**Slave:** Because the Serial Clock pin (SCK) is always an input, the device always operates as a slave.

**Transmitter/Receiver:** The device has separate pins designated for data transmission (SO) and reception (SI).

**Serial Output:** The SO pin is used to transfer data serially out of the device. During a read cycle data is shifted out on this pin after the falling edge of the Serial Clock.

**Serial Input:** The SI pin is used to transfer data serially into the device. It receives instructions, addresses, and data. Data is latched on the rising edge of the Serial Clock.

**Serial Clock:** The SCK pin is used to synchronize the communication between a master and the device. Instructions, addresses, or data, present on the SI pin, are latched on the rising edge of the clock input, while data on the SO pin is changed after the falling edge of the clock input.

**MSB:** The Most Significant Bit (MSB) is the first bit transmitted and received.

**Serial Op-Code:** After the device is selected with  $\bar{E}$  going low, the first byte will be received. This byte contains the op-code that defines the operations to be performed.

**Invalid Op-Code:** If an invalid op-code is received, no data will be shifted into the device, and the serial output

pin (SO) will remain in a high impedance state until the falling edge of  $\overline{E}$  is detected. This will re-initialize the serial communication.

**Chip Enable:** The device is selected when the  $\overline{E}$  pin is low. When the device is not selected ( $\overline{E}$  pin is high), data will not be accepted via the SI pin, and the serial output pin (SO) will remain in a high impedance state. Unless an internal Write cycle is in progress the device will be in the Standby mode. Driving Chip Enable ( $\overline{E}$ ) Low enables the device, placing it in the active power mode. After Power-up a falling edge on Chip Enable ( $\overline{E}$ ) is required prior to the start of any instruction.

**Write Protect:** The main purpose of this input signal is to freeze the size of the area of memory that is protected against Write instructions (as specified by the values in the BP1 and BP0 bits of the Status Register) and the selected PowerStore mode. This pin must be driven either High or Low, and must be stable during all write operations. In case the Write Protect pin is not available the part cannot be hardware protected (internal high).

**Hold:** The HOLD pin is used in conjunction with the  $\overline{E}$  pin to select the device. When the device is selected and a serial sequence is underway, HOLD can be used to pause the serial communication with the master device without resetting the serial sequence.

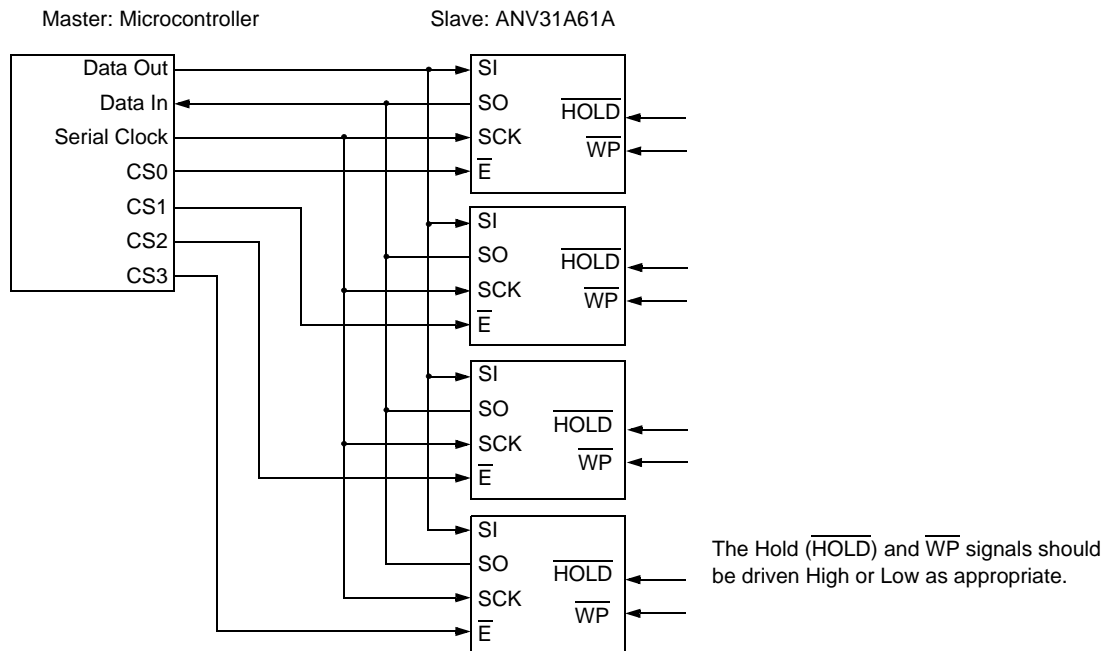
**Connecting to the SPI Bus**

These devices are fully compatible with the SPI protocol.

All instructions, addresses and input data bytes are shifted in to the device most significant bit first. The Serial Input (SI) is sampled on the first rising edge of the Serial Clock (SCK) after Chip Enable ( $\overline{E}$ ) goes Low. All output data bytes are shifted out after any read instruction, most significant bit first. The Serial Output (SO) is latched on the first falling edge of the Serial Clock (SCK) after the instruction (such as the Read from Memory Array, Secure Read and Read Status Register instructions) has been clocked into the device.

The Figure shows four devices, connected to an MCU, on an SPI bus. Only one device is selected at a time, so only one device drives the Serial Data Output (SO) line at a time, all the others being in high impedance.

**SPI BUS CONNECTION**



## SPI Modes

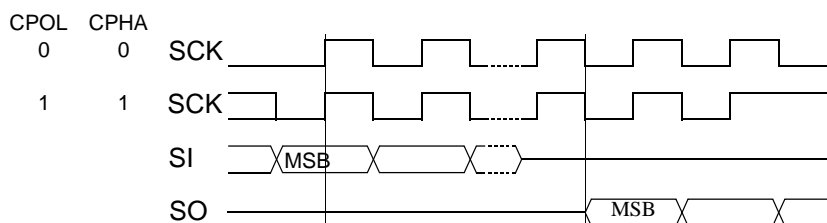
Each device can be driven by a microcontroller with its SPI peripheral running in either of the following two modes:

- Mode 0: CPOL=0, CPHA=0
- Mode 3: CPOL=1, CPHA=1

For these two modes, input data is latched in on the rising edge of Serial Clock (SCK), and output data is

available from the falling edge of Serial Clock (SCK). The difference between the two modes, as shown in the following figure, is the clock polarity when the bus master is in Stand-by mode and not transferring data:

- SCK remains at 0 for (CPOL=0, CPHA=0)
- SCK remains at 1 for (CPOL=1, CPHA=1)



## Operating Features

### Power up:

When the power supply is turned on from  $V_{SS}$ , Chip Enable ( $\bar{E}$ ) has to follow the  $V_{CC}$  voltage in accordance with the definition of  $V_{IH}$ . It must not be allowed to float, but could be connected via a suitable pull-up resistor to  $V_{CC}$ .

The Chip Enable signal ( $\bar{E}$ ) is edge as well as level sensitive. This ensures that the device becomes deselected after Power-down until  $\bar{E}$  reaches  $V_{CC}$  and a falling edge of  $\bar{E}$  from the  $V_{IH}$  level has been detected thereafter. This will start the first operation.

### Power On Reset:

In order to prevent data corruption and inadvertent Write operations during Power-up, all input signals will be ignored and Serial Data Output (SO) will be in high impedance state. Power On Reset is exited when  $V_{CC}$  reaches a stable  $V_{CCmin}$ . At this time logical signals can be applied.

### Power-down / Brown Out:

When  $V_{CC}$  drops during normal operation below  $V_{RESET}$  all external operations will be disabled, the device will ignore any input signals and Serial Data Output (SO) will be in high impedance state. Power-down during self timed Store Operation will corrupt data in the memory.

### Operating and Stand-by Modes:

When Chip Enable ( $\bar{E}$ ) is Low, the device is enabled. In Operating Mode it is consuming  $I_{CC(OP)}$ . In the other case, when Chip Enable ( $\bar{E}$ ) is High without prior Hibernate instruction, the device is in Standby Mode with the reduced Supply Current  $I_{CC(SB)}$  and with prior Hiber-

nate instruction the Supply Current will be with  $I_{CC(HM)}$  extreme low. To exit the Hibernate mode Chip Enable ( $\bar{E}$ ) has to go Low and after  $t_{RESTORE}$  operations can be executed.

### Hold Condition:

The Hold ( $\overline{HOLD}$ ) signal suspends any serial communication with the device without resetting the clock sequence.

Serial Data Output is in high impedance state during Hold condition,  $\overline{HOLD} = \text{Low}$ . The other SPI-inputs are disabled and Don't Care.

The device has to be active with Chip Enable ( $\bar{E}$ ) Low to enter the Hold condition. The device has to be selected for the duration of the Hold condition, for the selected operation to be continued after exiting the Hold condition. The Hold condition starts when Hold ( $\overline{HOLD}$ ) becomes Low, the device is active with Chip Enable ( $\bar{E}$ ) Low and Serial Clock (SCK) is already Low. The Hold conditions ends when Hold ( $\overline{HOLD}$ ) goes High, the device is still active with Chip Enable ( $\bar{E}$ ) Low and Serial Clock (SCK) is Low.

Chip Enable ( $\bar{E}$ ) has priority over Hold ( $\overline{HOLD}$ ). Driving Chip Enable ( $\bar{E}$ ) High during Hold condition will reset the device. With the next falling edge of Chip Select ( $\bar{E}$ ) a new instruction has to be submitted.

## Functional Description

The device utilizes an 8-bit instruction register. The list of instructions and their operation codes are contained in the following table. All instructions, addresses, and data are transferred with the MSB first and start with a high-to-low transition on the  $\bar{E}$ -pin. Each instruction starts with one of the single-byte codes below.

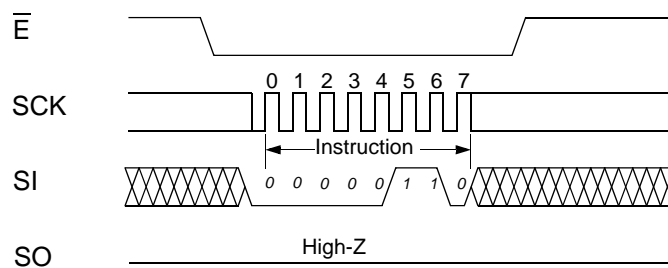
| Instruction Name | Instruction format | Operation                                   |
|------------------|--------------------|---|
| WREN             | 0000 0110          | Set Write Enable Latch                      |
| WRDI             | 0000 0100          | Reset Write Enable Latch                    |
| RDSR             | 0000 0101          | Read Status Register                        |
| WRSR             | 0000 0001          | Write Status Register                       |
| READ             | 0000 0011          | Read Data from Memory Array                 |
| SECURE READ      | 0001 0011          | Secure Read Data from Memory Array with CRC |
| WRITE            | 0000 0010          | Write Data to Memory Array                  |
| SECURE WRITE     | 0001 0010          | Secure WRITE Data to Memory Array with CRC  |
| STORE            | 0000 1000          | Store SRAM data non-volatile                |
| RECALL           | 0000 1001          | Recall non-volatile data to SRAM            |
| WRSNR            | 1100 0010          | WRITE User Serial Number                    |
| RDSNR            | 1100 0011          | READ User Serial Number                     |
| Hibernate        | 1011 1001          | Enter Hibernate Mode                        |

**Write Enable (WREN):**

The device will power-up in the write disable state when  $V_{CC}$  is applied. Before any WRITE instruction is accepted, the Write Enable Latch has to be set with the WREN command.

the device, Chip Enable ( $\bar{E}$ ) is driven Low, and the bits of the instruction byte are shifted in on Serial Data Input (SI). The device then enters a wait state, waiting for the device to be deselected, by Chip Enable ( $\bar{E}$ ) being driven High.

As shown in the figure below, to send this instruction to



**Write Disable (WRDI):**

To protect the device against inadvertent writes, the Write Disable instruction disables all WRITE modes. The WRDI instruction is independent of the status of the WP pin.

The Write Enable Latch (WEN) bit can be reset by any of the following events:

As shown in the figure below, to send this instruction to the device, Chip Enable ( $\bar{E}$ ) is driven Low and the bits of the instruction byte are shifted in, on Serial Data Input (SI). The device then enters a wait state, waiting for the device to be deselected, by Chip Enable ( $\bar{E}$ ) being driven High.

- Power-up
- WRDI instruction execution
- WRSR instruction completion
- WRSNR instruction completion
- WRITE instruction completion
- SECURE WRITE instruction completion



Status Register

| Bit 7 | Bit 6 | Bit 5 | Bit 4            | Bit 3 | Bit 2 | Bit 1 | Bit 0            |
|-------|-------|-------|------------------|-------|-------|-------|------------------|
| WPEN  |       | PRO   | $\overline{SWM}$ | BP1   | BP0   | WEN   | $\overline{RDY}$ |

| Bit                        | non-volatile | Definition   |
|----------------------------|--------------|--|
| Bit 0 ( $\overline{RDY}$ ) | no           | The Ready bit indicates whether the memory is busy with a STORE or RECALL cycle. When set to 1, such a cycle is in progress, when reset to 0 no such cycle is in progress. It is a read only bit   |
| Bit 1 (WEN)                | no           | The Write Enable Latch bit indicates the status of the internal Write Enable Latch. When set to 1 with a Write Enable (WREN) instruction the internal Write Enable Latch is set, when set to 0 with a Write Disable (WRDI) instruction the internal Write Enable Latch is reset and no Write, Secure Write, Write Serial Number or Write Status Register instruction are accepted.   |
| Bit 2 (BP0)                | yes          | The Block Protect (BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Write instructions. These bits are written with the Write Status Register (WRSR) instruction. When one or both of the Block Protect (BP1, BP0) bits are set to 1, the relevant memory area (see Write Status Register WRSR) becomes protected against all Write (WRITE, Secure WRITE) instructions to the Memory Array. The Block Protect (BP1, BP0), Write Protect Enable (WPEN) and Power Store Disable (PDIS) bits can be written provided that the Hardware Protected Mode has not been set. |
| Bit 3 (BP1)                | yes          |  |
| Bit 4 ( $\overline{SWM}$ ) | no           | Secure WRITE Monitoring bit indicates the success of the last Secure Write operation. With $\overline{SWM} = 0$ Secure Write was successful, with $\overline{SWM} = 1$ data and/or address were corrupt. Secure Write was ignored. It is a read only bit.  |
| Bit 5 ( $\overline{PRO}$ ) | yes          | Page RollOver bit is non-volatile. Set $\overline{PRO}=0$ will activate page roll over mode, with $\overline{PRO} = 1$ the block roll over mode is selected.   |
| Bit 6                      |              | don't care   |
| Bit 7 (WPEN)               | yes          | The Write Protect Enable bit is operated in conjunction with the Write Protect ( $\overline{WP}$ ) signal. The Status Register Write Protect Enable (WPEN) bit and Write Protect ( $\overline{WP}$ ) signal allow the device to be put in the Hardware Protected Mode (when the Status Register Write Protect Enable (WPEN) bit is set to 1, and Write Protect ( $\overline{WP}$ ) is driven Low). In this mode, the non-volatile bits of the Status Register become read-only bits and the Write Status Register (WRSR) instruction is no longer accepted for execution.  |

**Read Status Register (RDSR):**

The Read Status Register instruction provides access to the status register. The READY/BUSY and Write Enable status of the device can be detected by the RDSR instruction. Similarly, the Block Write Protection bits and WPEN indicate the extent of protection

employed. These bits, besides  $\overline{\text{RDY}}$ ,  $\overline{\text{SWM}}$  and WEN, are set by using the WRSR instruction. The  $\overline{\text{SWM}}$  bit will be automatically set as result of a corrupt volatile data transfer in Secure WRITE and can just be reset to zero by a successful SECURE WRITE.



**Write Status Register (WRSR):**

The WRSR instruction allows the user to select one of four levels of protection.

$\overline{\text{SWM}}$  is internally reset to 0 at the begin of each Secure WRITE operation.

WREN command has to be sent prior to WRSR. The properties for access the Status Register are the same as the memory array. The WRSR instruction is volatile.

The device is divided into several array segments. One quarter, one half, or all of the memory segments can be protected. Any data within any selected segment will therefore be READ only. The block write protection levels and corresponding status register control bits BP1 and BP2 are shown in the following table.

To make Bit2, Bit3, Bit5 and Bit7 non-volatile a STORE instruction has to follow the WRSR instruction. This STORE instruction is also valid for the memory array and all other non-volatile registers.

| Level<br>(Protected Block) | Status Register Bits |     | Array Addresses Protected |
|----------------------------|----------------------|-----|---------------------------|
|                            | BP1                  | BP0 | ANV31A81A                 |
| 0                          | 0                    | 0   | None                      |
| 1 (Upper quarter)          | 0                    | 1   | 6000 - 7FFF               |
| 2 (Upper half)             | 1                    | 0   | 4000 - 7FFF               |
| 3 (Whole memory)           | 1                    | 1   | 0000 - 7FFF               |

The WRSR instruction also allows the user to enable or disable the write protect (WP) pin through the use of the Write Protect Enable (WPEN) bit. Hardware Write Protection Mode (HPM) is enabled when the  $\overline{\text{WP}}$  pin is low and the WPEN bit is "1". Hardware Write Protection Mode is disabled when either the  $\overline{\text{WP}}$  pin is high or the WPEN bit is "0". When the device is hardware write

protected, writes to the Status Register, including the Block Protect bits and the WPEN bit, and the block-protected sections in the memory array are disabled. Writes are only allowed to sections of the memory which are not block-protected. When the WPEN bit is hardware write protected, it cannot be changed back to "0", as long as the  $\overline{\text{WP}}$  pin is held low .

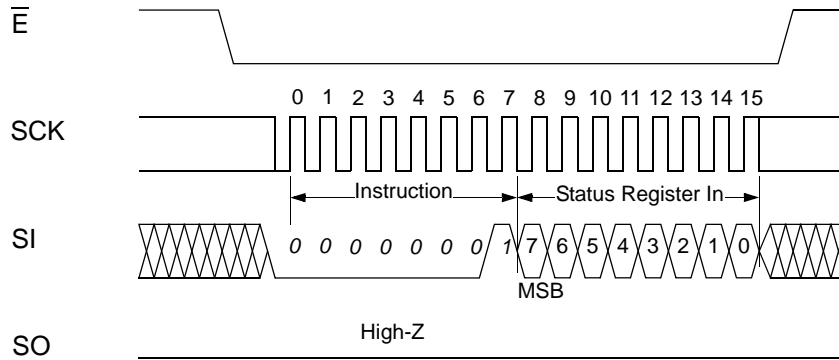
| WPEN | $\overline{\text{WP}}$ | Mode                   | Protected Blocks <sup>a</sup> | Unprotected Blocks <sup>a</sup> | Status Register  |
|------|------------------------|------------------------|-------------------------------|---------------------------------|--|
| 0    | H                      | Software               | Protected                     | Writeable                       | Writable, if WEN=1,<br>BP0, BP1 and PDIS are<br>writable |
| 0    | L                      | Protected<br>SPM       | Protected                     | Writeable                       |  |
| 1    | H                      |                        | Protected                     | Writeable                       |  |
| 1    | L                      | Hardware Protected HPM | Protected                     | Writeable                       | Protected, if WEN=1, no<br>access to BP0, BP1 and PDIS   |

<sup>a</sup> As defined by the values in the Block Protect (BP1, BP0) bits of the Status Register.

# ANV31A81A

$\overline{\text{PRO}}$  defines the roll over modes page or block, which are applicable only for WRITE operation. It will be ignored in case of Secure WRITE operation. The Write Status Register (WRSR) instruction has no effect on b4 (SWM), b1 (WEN) and b0 (RDY) of the

Status Register. Chip Enable (E) must be driven High after the eighth bit of the data byte has been clocked in. If not, the Write Status Register (WRSR) instruction is not executed.



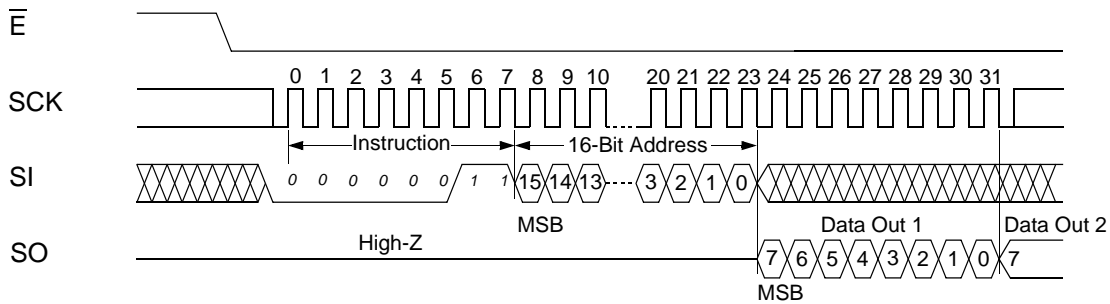
## Read from Memory Array (READ):

Reading the device via the SO (Serial Output) pin requires the following sequence. After the  $\overline{\text{E}}$  is pulled low to select a device, the READ operation code is transmitted via SI followed by the byte address to be read (A15 - A0, don't care bits is A15).

READ operation always in block roll over mode, so that the READ sequence can be continued, the byte address is automatically incremented and data will continue to be shifted out. When the highest address is reached, the address counter will roll over to the lowest address allowing the entire memory to be read in one continuous READ cycle.

Upon completion, any data on SI will be ignored. The data (D7 - D0) at the specified address is then shifted out onto SO. If only one byte is to be read,  $\overline{\text{E}}$  should be driven high after the data comes out. The device is for

The Read cycle can be started when a WRITE or STORE cycle is not in progress and terminated at any time driving the Chip Enable ( $\overline{\text{E}}$ ) to High.

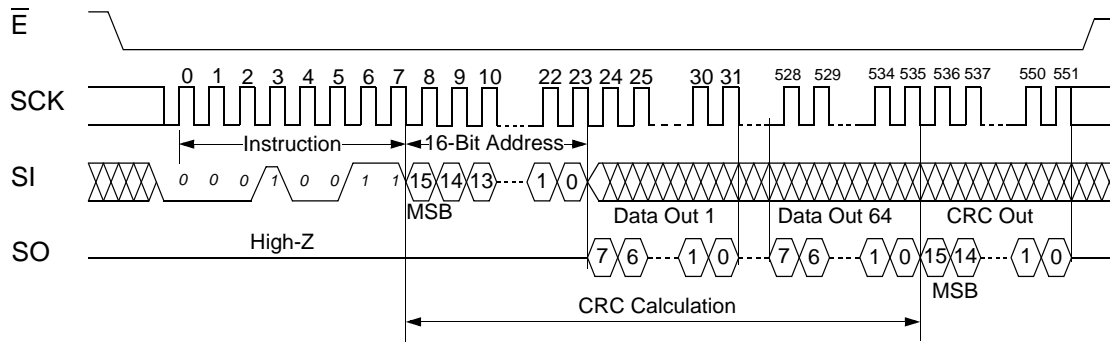


## Secure Read Memory Array (Secure READ):

The secure READ operation is a 64 Byte data read out of the memory array. In parallel with the data transfer to the external bus internally a CRC is calculated, including the start address and all 64 Byte data. After last byte is read the 16 bit CRC will be clocked out on SO. Only the actual 15 address bits are used for CRC

calculation. The unused address bit is not included in the CRC calculation. The CRC16-CCITT polynomial  $x^{16}+x^{12}+x^5+1$  is used for calculation. Page roll over is defined for Secure READ. The initial value is 0xFFFF. The checksum is transmitted with MSB first.





**Write to Memory Array (WRITE):**

In order to write the device, two separate instructions must be executed. First, the device must be write enabled via the Write Enable (WREN) Instruction. Then a Write (WRITE) Instruction may be executed. Also, the address of the memory location(s) to be written must be outside the protected address field location, selected by the Block Write Protection Level.

A Write Instruction requires the following sequence. After  $\overline{E}$  is pulled low to select the device, the WRITE operation code is transmitted via SI followed by the byte address (A15 - A0) and the data (D7 - D0) to be written. Write to the SRAM will start after the  $\overline{E}$  pin is brought high. The Low-to-High transition of the  $\overline{E}$  pin must occur during the SCK low-time immediately after clocking in the D0 (LSB) data bit, if not, the WRITE operation will not be executed.

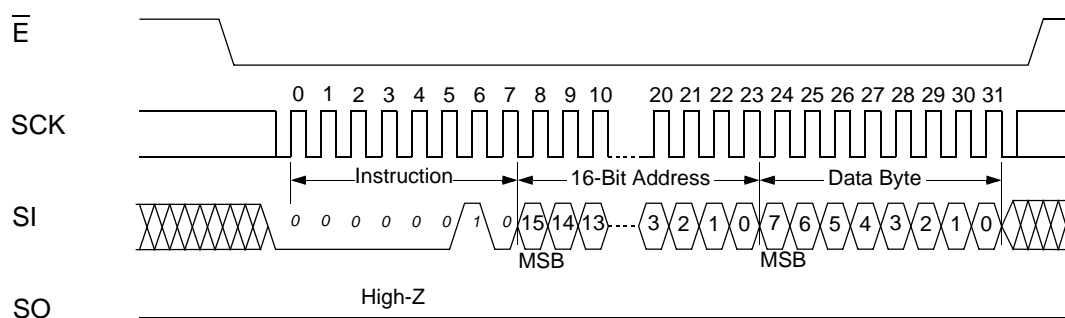
With status register setting  $\overline{PRO}=0$  the device is capable of a 64-byte PAGE WRITE operation. After each byte of data is received, the six low order address bits are internally incremented by one; the high order bits of the address will remain constant. If more than 64 Bytes of data are transmitted, the address counter will roll

over and the previously written data will be overwritten. At this point in time the written data are volatile.

With status register setting  $\overline{PRO}=1$  the block roll over mode will be applied and up to the whole memory array can be written with one command. After each byte of data is received, the address bits are internally incremented by one. Reaching the end of a page all previous written data on this page becomes valid and former SRAM data is overwritten. WRITE will be continued on next page and address bits will be internally incremented until page end. Reaching the highest address the counter will roll over. This process can be continued until all data are written.

With the rising edge of  $\overline{E}$  the previous SRAM data becomes invalid and the volatile Write operation of the data will occur. The device is automatically returned to the write disable state on completion of a WRITE cycle.

If the device is not Write enabled (WREN), the device will ignore the Write instruction and will return to the standby state, when  $\overline{E}$  is brought high. A new  $\overline{E}$  falling edge is required to re-initiate the serial communication.



**Secure WRITE Memory Array (Secure WRITE):**

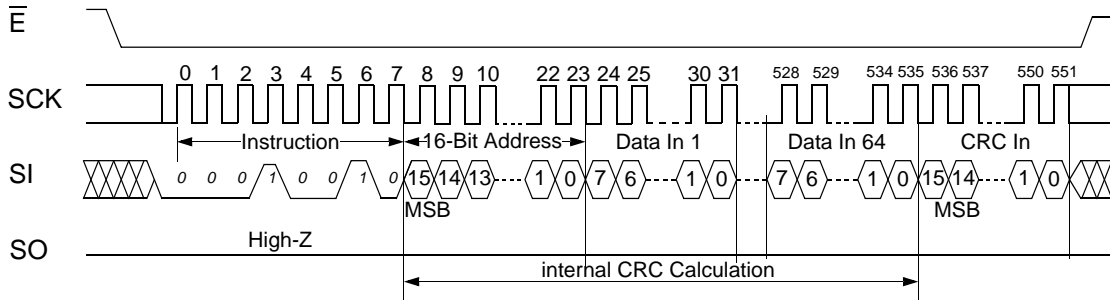
To enable the Secure WRITE operation a WREN has to occur first. Secure WRITE is a 64 Byte data WRITE to the memory array. A CRC is calculated, in parallel with the data transfer, from the 16 bit address and 64 Byte data. After last byte is written the 16 bit CRC has to be clocked in on SI. The actual 15 address bits will

be used for CRC calculation. The CRC16-CCITT polynomial used is  $x^{16}+x^{12}+x^5+1$ . Page roll over is defined for Secure WRITE. The initial value is 0xFFFF. The checksum must be transmitted with MSB first on SI. The Low-to-High transition of the  $\overline{E}$  pin must occur during the SCK low-time immediately after clocking in the

# ANV31A81A

CRC0 (LSB) bit, if not, the WRITE operation will not be executed. In addition the internally calculated CRC has to match the transmitted CRC. In this case data will be accepted. If the CRC's don't match data will be ignored, the existing memory data will stay and Status

Register bit 4 will be set to 1. With RDSR the success of Secure WRITE has to be checked after every Secure Write operation and bit 4 is reset to 0 at the begin of next Secure Write operation.



## STORE

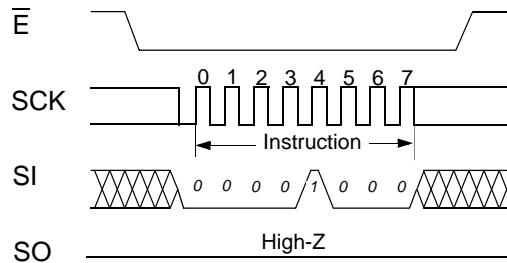
Data can be transferred from the SRAM to the non-volatile memory by a STORE command. During the STORE cycle, previous non-volatile data will be erased and then the new data stored into the non-volatile elements. Once a STORE cycle is initiated, further memory inputs and outputs are disabled until the cycle is completed. During a STORE operation, all commands will be ignored except the RDSR instruction.

The READY/BUSY status of the device can be determined by initiating a Read Status Register (RDSR) Instruction.

After the  $t_{STORE}$  cycle time has been fulfilled, the SRAM will again be activated for any READ or WRITE operations.

Instruction initiated STORE cycles are performed regardless of whether a WRITE operation has taken place after the last STORE or POWER-UP and is valid also for all non-volatile registers.

In case no command will be sent after STORE cycle completion a new STORE cycle will be initiated with the rising edge of  $\bar{E}$ .



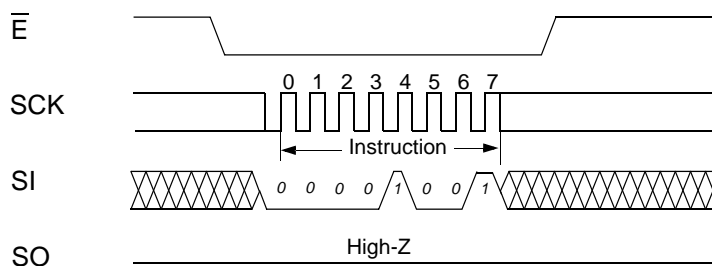
## RECALL

Data can be transferred from the non-volatile memory to the SRAM by a RECALL command.

During an Recall operation, all commands will be ignored except the RDSR instruction. In case no command will be sent after RECALL cycle completion a new RECALL cycle will be initiated with the rising edge of  $\bar{E}$ .

Internally, RECALL is a two-step procedure. First, the SRAM data is cleared, and second, the non-volatile information is transferred into the SRAM cells. After the  $t_{RECALL}$  cycle time, the SRAM will be activated for any operations. The RECALL operation in no way alters the data in the non-volatile storage elements.

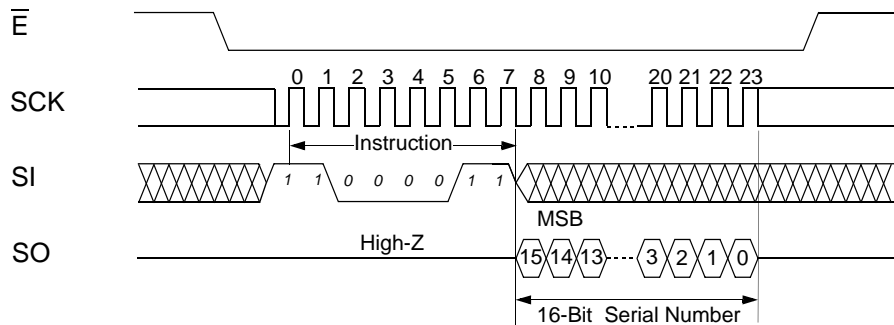
The READY/BUSY status of the device can be determined by initiating a Read Status Register (RDSR) Instruction.



**Read User Serial Number (RDSNR)**

Anv31A81 supports an additional non-volatile 16 bit register for a user-controlled serial number.

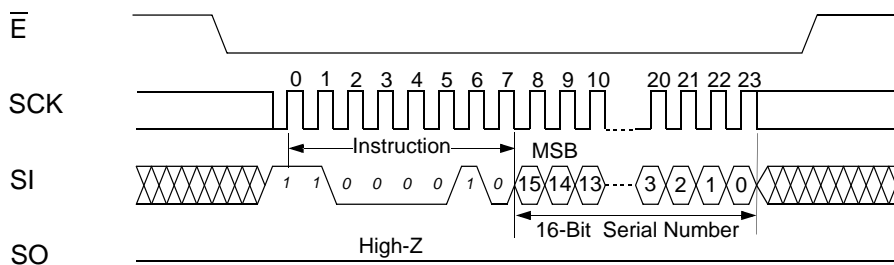
With RDSNR register content can be read out.



**Write User Serial Number (WRSNR)**

To enable the WRSNR operation a WREN has to occur first. With WRSNR a 16 bit user-controlled serial number can be written volatile to the register.

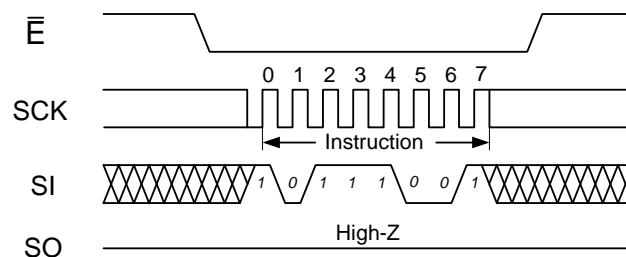
All 16 bit have to be clocked in otherwise the data will be ignored. With a STORE operation data in the register becomes non-volatile.



**Hibernate Mode:**

To enable the HIBERNATE mode the Hibernate command has to be transferred. After  $\bar{E}$  goes high the ANV31A81 will ignore any input signals until  $\bar{E}$  goes low again. During HIBERNATE mode the part will consume only the current  $I_{SBH}$ .

With the falling edge of  $\bar{E}$  an internal Power-up Recall cycle will be initiated and after this cycle is completed the device is ready for any operation.



## **ANV31A81A**

---

### **Power Up Recall:**

During power up or after any low-power condition ( $V_{CC} < V_{RESET}$ ), an internal RECALL request will be latched. When  $V_{CC}$  once again exceeds the sense voltage of  $V_{RESET}$ , a RECALL cycle will automatically be initiated and will take  $t_{RECALL}$  to complete.

During Power Up Recall operation, all commands will be ignored .

**ABSOLUTE MAXIMUM RATINGS<sup>a</sup>**

Voltage on Input Relative to Ground . . . . . -0.5V to 4.5V  
 Voltage on Input Relative to VSS . . . . . -0.6V to (VCC + 0.5V)  
 Temperature under Bias . . . . . -55°C to 125°C  
 Storage Temperature . . . . . -65°C to 150°C  
 Power Dissipation . . . . . 0.5W  
 DC Output Current (1 output at a time, 1s duration) . . . . . 15mA

a. Stresses greater than those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

**Operating Conditions**

| Symbol          | Parameter         | ANV31A81A |      | Unit |
|-----------------|-------------------|-----------|------|------|
|                 |                   | Min.      | Max. |      |
| V <sub>CC</sub> | Operating Voltage | 3.0       | 3.6  | V    |

**DC CHARACTERISTICS**

(V<sub>CC</sub> / V = 3.0 - 3.6)

| SYMBOL                        | PARAMETER                                    | COMMERCIAL            |                       | INDUSTRIAL            |                       | UNITS | NOTES  |
|-------------------------------|--|-----------------------|-----------------------|-----------------------|-----------------------|-------|--|
|                               |  | MIN                   | MAX                   | MIN                   | MAX                   |       |  |
| I <sub>CC1</sub> <sup>a</sup> | Average V <sub>CC</sub> Current at 66MHz     |                       | 5                     |                       | 5                     | mA    | Byte READ to Byte WRITE ratio 1:1<br>V <sub>IN</sub> ≤ 0.2V <sub>CC</sub> or ≥ 0.8V <sub>CC</sub>                  |
| I <sub>CC2</sub> <sup>b</sup> | Average V <sub>CC</sub> Current during STORE |                       | 2                     |                       | 2                     | mA    | All Inputs Don't Care, V <sub>CC</sub> = max   |
| I <sub>CC3</sub> <sup>a</sup> | Average V <sub>CC</sub> Current at 10MHz     |                       | 2                     |                       | 2                     | mA    | Byte READ to Byte WRITE ratio 1:1,<br>V <sub>IN</sub> ≤ 0.2V <sub>CC</sub> or ≥ 0.8V <sub>CC</sub>                 |
| I <sub>SB1</sub> <sup>c</sup> | Average V <sub>CC</sub> Current Standby      |                       | 400                   |                       | 400                   | µA    | $\bar{E} \geq V_{IH}$ , Cycling input levels   |
| I <sub>SB2</sub> <sup>c</sup> | V <sub>CC</sub> Standby Current              |                       | 300                   |                       | 300                   | µA    | $\bar{E} \geq (V_{CC} - 0.2V)$<br>All Others V <sub>IN</sub> ≤ 0.2V or ≥ (V <sub>CC</sub> - 0.2V)                  |
| I <sub>SBH</sub>              | Hibernate Standby Current                    |                       | 3                     |                       | 3                     | µA    |  |
| I <sub>ILK</sub>              | Input Leakage Current                        |                       | ±1                    |                       | ±1                    | µA    | V <sub>CC</sub> = max<br>V <sub>IN</sub> = V <sub>SS</sub> to V <sub>CC</sub>                                      |
| I <sub>OLK</sub>              | Off-State Output Leakage Current             |                       | ±1                    |                       | ±1                    | µA    | V <sub>CC</sub> = max<br>V <sub>IN</sub> = V <sub>SS</sub> to V <sub>CC</sub> , $\bar{E}$ or $\bar{G} \geq V_{IH}$ |
| V <sub>IH</sub>               | Input Logic "1" Voltage                      | 0.8V <sub>CC</sub>    | V <sub>CC</sub> + 0.5 | 0.8V <sub>CC</sub>    | V <sub>CC</sub> + 0.5 | V     | All Inputs   |
| V <sub>IL</sub>               | Input Logic "0" Voltage                      | V <sub>SS</sub> - 0.5 | 0.2V <sub>CC</sub>    | V <sub>SS</sub> - 0.5 | 0.2V <sub>CC</sub>    | V     | All Inputs   |
| V <sub>OH</sub>               | Output Logic "1" Voltage                     | V <sub>CC</sub> -0.5  |                       | V <sub>CC</sub> -0.5  |                       | V     | I <sub>OUT</sub> = -0.4 mA   |
| V <sub>OL</sub>               | Output Logic "0" Voltage                     |                       | 0.4                   |                       | 0.4                   | V     | I <sub>OUT</sub> = 2 mA  |
| T <sub>A</sub>                | Operating Temperature                        | 0                     | 70                    | -40                   | 85                    | °C    |  |
| NV <sub>C</sub>               | non-volatile STORE operations                | 100                   |                       | 100                   |                       | K     |  |
| DATA <sub>R</sub>             | Data Retention                               | 100                   |                       | 100                   |                       | Years | @55 °C   |

Note a: I<sub>CC1</sub> and I<sub>CC3</sub> are dependent on output loading and cycle rate. The specified values are obtained with outputs unloaded.

Note b: I<sub>CC2</sub> is the average current required for the duration of the respective STORE cycles (t<sub>STORE</sub>).

Note c:  $\bar{E} \geq V_{IH}$  will not produce standby current levels until any non-volatile cycle in progress has timed out.

| Switching Characteristics    | Symbol     |              | Min. | Max. | Unit |
|------------------------------|------------|--------------|------|------|------|
|                              | Alt.       | IEC          |      |      |      |
| SCK Clock Frequency          | $f_{SCK}$  | $f_C$        | 0    | 66   | MHz  |
| Chip Enable Setup Time       | $t_{CSS}$  | $t_{su(E)}$  | 6    |      | ns   |
| /E High Time                 | $t_{CS}$   | $t_2$        | 7    |      | ns   |
| /E Hold Time                 | $t_{CSH}$  | $t_{h(E)}$   | 6    |      | ns   |
| Clock Setup time             | $t_{SKSH}$ | $t_{su(C)}$  | 3    |      | ns   |
| Clock High Time <sup>d</sup> | $t_{CLH}$  | $t_5$        | 7    |      | ns   |
| Clock Low Time <sup>d</sup>  | $t_{CLL}$  | $t_6$        | 7    |      | ns   |
| Clock Rise Time              | $t_{RC}$   | $t_7$        |      | 100  | ns   |
| Clock Fall Time              | $t_{FC}$   | $t_8$        |      | 100  | ns   |
| Input Rise Time              | $t_{RD}$   |              |      | 100  | ns   |
| Input Fall Time              | $t_{FD}$   |              |      | 100  | ns   |
| Data Setup Time              | $t_{DSU}$  | $t_{su(D)}$  | 4    |      | ns   |
| Data Hold Time               | $t_{DH}$   | $t_{h(D)}$   | 4    |      | ns   |
| /HOLD Hold Time              | $t_{HH}$   | $t_{h(H)}$   | 5    |      | ns   |
| /HOLD Setup Time             | $t_{HSU}$  | $t_{su(H)}$  | 0    |      | ns   |
| Output Disable Time          | $t_{DIS}$  | $t_{dis(E)}$ |      | 20   | ns   |
| Clock Low to Output Valid    | $t_V$      | $t_{en(C)}$  |      | 10   | ns   |
| Output Hold Time             | $t_{HO}$   | $t_{h(D)}$   | 0    |      | ns   |
| /HOLD High to Output Low-Z   | $t_{LZ}$   | $t_{en(H)}$  |      | 10   | ns   |
| /HOLD Low to Output High-Z   | $t_{HZ}$   | $t_{dis(H)}$ |      | 10   | ns   |

<sup>e</sup>  $t_{CH} + t_{CL} \geq 1 / f_{SCK}$

## AC TEST CONDITIONS

|  |              |
|--|--------------|
| Input Pulse Levels                       | 0V to 3V     |
| Input Rise and Fall Times                | $\leq 5$ ns  |
| Input and Output Timing Reference Levels | 1.5V         |
| Output Load                              | See Figure 1 |

## CAPACITANCE<sup>d</sup> ( $T_A = 25^\circ\text{C}$ , $f = 1.0\text{MHz}$ )

| SYMBOL    | PARAMETER          | MAX | UNITS | CONDITIONS             |
|-----------|--------------------|-----|-------|------------------------|
| $C_{IN}$  | Input Capacitance  | 5   | pF    | $\Delta V = 0$ to 3.3V |
| $C_{OUT}$ | Output Capacitance | 7   | pF    | $\Delta V = 0$ to 3.3V |

Note d: These parameters are guaranteed but not tested.



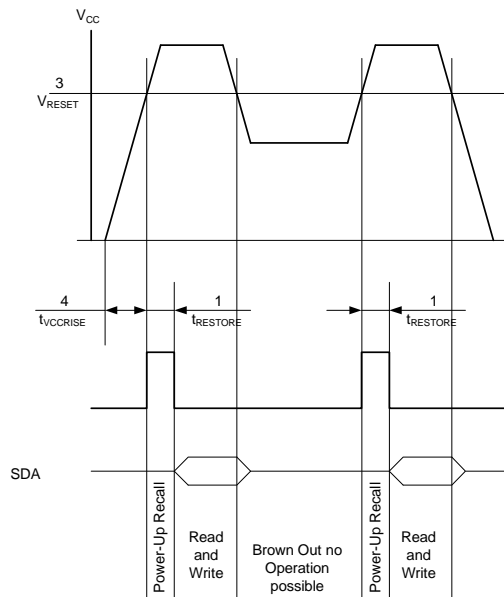
Figure 1. AC Output Loading

**STORE / POWER-UP RECALL**

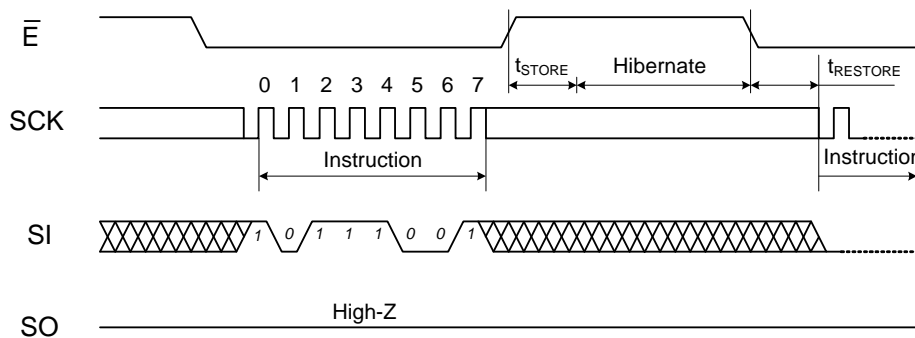
| NO. | SYMBOLS        |           | PARAMETER  | ANV31A81A |      | UNITS   | NOTES |
|-----|----------------|-----------|--|-----------|------|---------|-------|
|     | Standard       | Alternate |  | MIN       | MAX  |         |       |
| 1   | $t_{RESTORE}$  |           | Power-up <i>RECALL</i> Duration                      |           | 200  | $\mu$ s | e     |
| 2   | $t_{STORE}$    |           | <i>STORE</i> Cycle Duration                          |           | 8    | ms      |       |
| 3   | $V_{RESET}$    |           | Low Voltage Trigger Level                            | 2.3       | 2.65 | V       |       |
| 4   | $t_{VCCRRISE}$ |           | $V_{CC}$ rise time                                   | 100       |      | $\mu$ s |       |
| 5   | $t_{RECALL}$   |           | <i>RECALL</i> Duration (normal operating conditions) |           | 50   | $\mu$ s | f     |

Note e:  $t_{RESTORE}$  starts from the time  $V_{CC}$  rises above  $V_{SWITCH}$ .  
 Note f:  $V_{CC} > V_{CCmin}$

**POWER-UP RECAL**

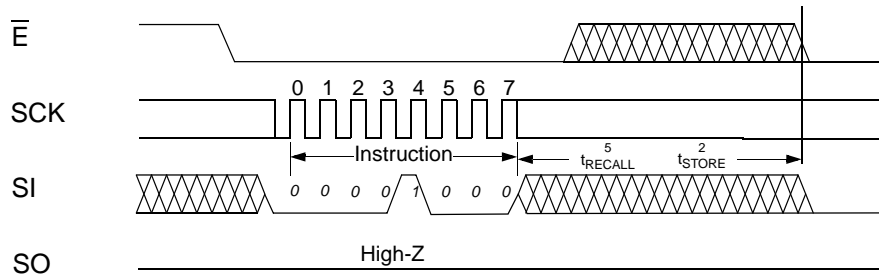


**Hibernate**

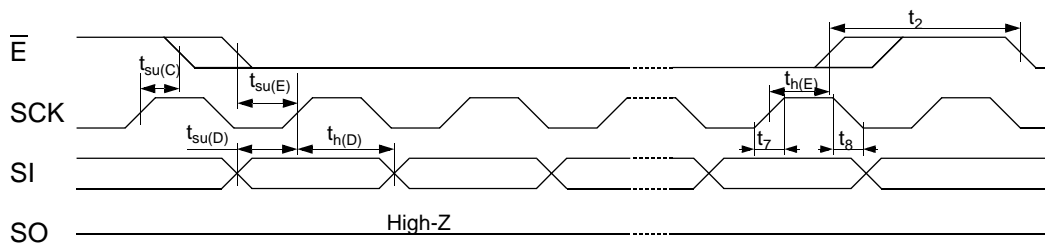


# ANV31A81A

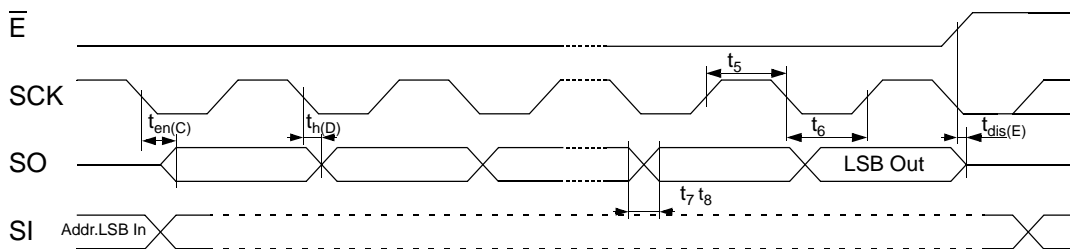
## STORE/RECALL CYCLE ( $V_{CC} > V_{CCmin}$ )



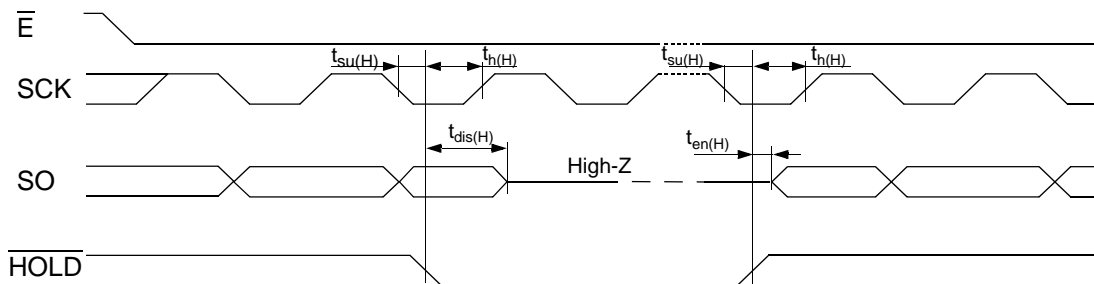
### Serial Input Timing



### Serial Output Timing



### Hold Timing





**Product Versions**

The ANV31A81A will be available with the feature sets:

- Supply voltage range 3.0 to 3.6V
- Hardware WRITE protection with  $\overline{WP}$  pin.

**Initial Delivery State**

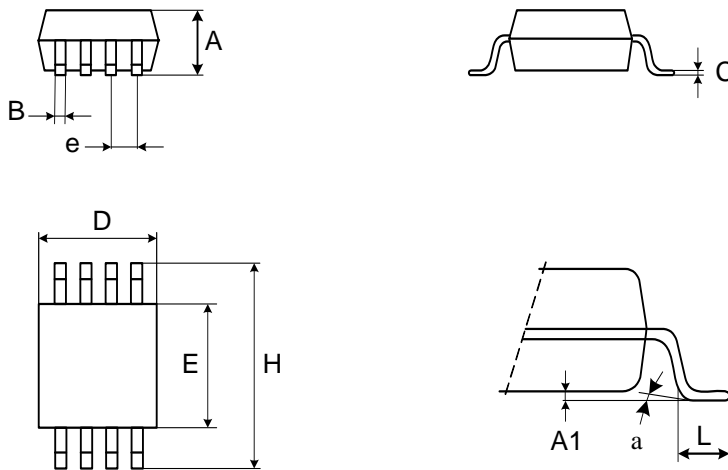
The device is delivered with Status Register 00x00xx, non-volatile memory array „0“.

**NOISE CONSIDERATIONS**

The ANV31A81A is a high-speed memory and so must have a high-frequency bypass capacitor of approximately 0.1µF connected between V<sub>CC</sub> and V<sub>SS</sub>, using leads and traces that are as short as possible. As with all high-speed CMOS ICs, normal careful routing of power, ground and signals will help prevent noise problems.

**Package**

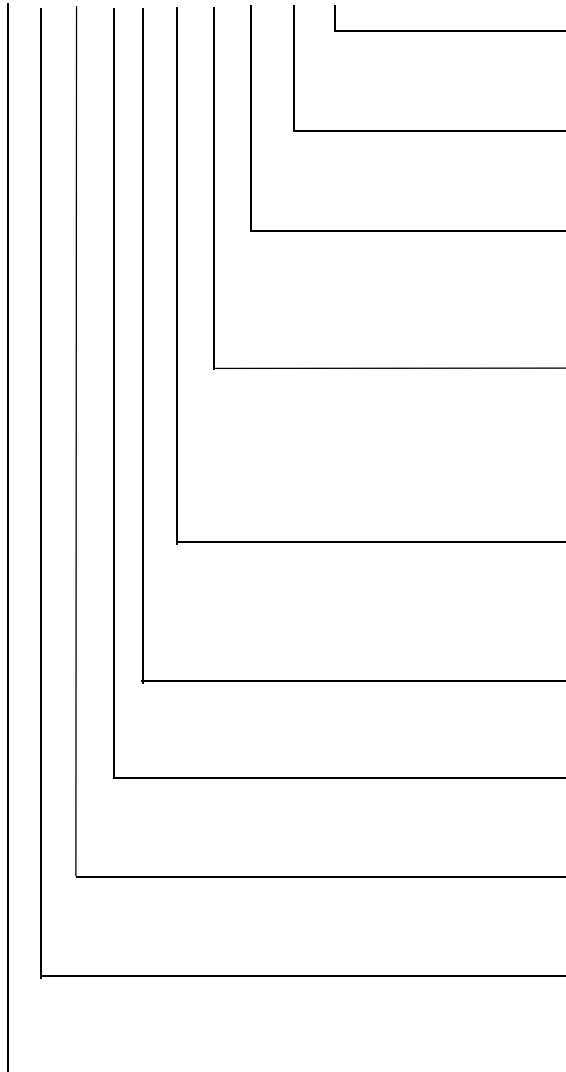
**8-pin 150mil SOIC**



| Symbol | mm    |      |      | inches |       |       |
|--------|-------|------|------|--------|-------|-------|
|        | typ.. | min. | max  | typ.   | min.  | max.  |
| A      |       | 1.35 | 1.75 |        | 0.053 | 0.069 |
| A1     |       | 0.1  | 0.25 |        | 0.004 | 0.010 |
| B      |       | 0.33 | 0.51 |        | 0.013 | 0.020 |
| C      |       | 0.19 | 0.25 |        | 0.007 | 0.010 |
| D      |       | 4.80 | 5.00 |        | 0.189 | 0.197 |
| E      |       | 3.80 | 4.00 |        | 0.150 | 0.157 |
| e      | 1.27  |      |      | 0.050  |       |       |
| H      |       | 5.80 | 6.20 |        | 0.228 | 0.244 |
| L      |       | 0.40 | 0.90 |        | 0.016 | 0.035 |
| α      |       | 0°   | 8°   |        | 0°    | 8°    |

## Ordering Information

ANV 3 1 A 8 1 A S C 66 \_



### Lead Finish

Blank = 100% Sn (Matte Tin)

### Clock Rate

66 = 66 MHz

### Temperature Range

C = Commercial (0 to 70°C)

K = Industrial (-40 to 85°C)

### Package

S = Plastic 8-pin 150 mil SOIC

### Power Supply

A = 3.0V ... 3.6V

### Interface

1 = SPI

### Density

8 = 256Kb

### Version

A = actual

### Store Type

1 = Instruction

### Memory Type

3= serial nvSRAM

**Document Revision History**

| Revision | Date           | Summary         |
|----------|----------------|-----------------|
| 1.0      | September 2018 | initial version |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |
|          |                |                 |

Anvo-Systems Dresden ANV31A81A Datasheet, September 2018  
Copyright 2018, Anvo-Systems Dresden GmbH. All rights reserved.

Компания «Life Electronics» занимается поставками электронных компонентов импортного и отечественного производства от производителей и со складов крупных дистрибьюторов Европы, Америки и Азии.

С конца 2013 года компания активно расширяет линейку поставок компонентов по направлению коаксиальный кабель, кварцевые генераторы и конденсаторы (керамические, пленочные, электролитические), за счёт заключения дистрибьюторских договоров

Мы предлагаем:

- Конкурентоспособные цены и скидки постоянным клиентам.
- Специальные условия для постоянных клиентов.
- Подбор аналогов.
- Поставку компонентов в любых объемах, удовлетворяющих вашим потребностям.
- Приемлемые сроки поставки, возможна ускоренная поставка.
- Доставку товара в любую точку России и стран СНГ.
- Комплексную поставку.
- Работу по проектам и поставку образцов.
- Формирование склада под заказчика.
- Сертификаты соответствия на поставляемую продукцию (по желанию клиента).
- Тестирование поставляемой продукции.
- Поставку компонентов, требующих военную и космическую приемку.
- Входной контроль качества.
- Наличие сертификата ISO.

В составе нашей компании организован Конструкторский отдел, призванный помогать разработчикам, и инженерам.

Конструкторский отдел помогает осуществить:

- Регистрацию проекта у производителя компонентов.
- Техническую поддержку проекта.
- Защиту от снятия компонента с производства.
- Оценку стоимости проекта по компонентам.
- Изготовление тестовой платы монтаж и пусконаладочные работы.



Тел: +7 (812) 336 43 04 (многоканальный)

Email: [org@lifeelectronics.ru](mailto:org@lifeelectronics.ru)