AN2632
Application note

Communication between small page NAND and ST72651AR6 using I/O's

Introduction

This application note presents the practical example of communication between microcontroller and NAND Flash using general purpose input/output ports (GPIOs).

This document describes the hardware connections and software necessary to establish communication between the STMicroelectronics small page NAND Flash memory and input/output ports of the ST7 microcontroller. We have used the ST72651AR6 for the firmware description, but any ST7 MCU can be used as well.

The devices covered by this application note are:

- NAND128W3A
- NAND256W3A
- NAND512W3A
- NAND01GW3A

Contents

1 NAND Flash overview

The NAND Flash 528 Byte / 264 Word page is a family of non-volatile Flash memories that uses the Single Level Cell (SLC) NAND cell technology. It is referred to as the small page family. The devices range from 128 Mbits to 1Gbit and operate with either a 1.8 V or 3 V voltage supply. The size of a page is either 528 Bytes (512 + 16 spare) or 264 Words (256 + 8 spare) depending on whether the device has an x8 or x16 bus width. The address lines are multiplexed with the data input/output signals on a multiplexed x8 or x16 input/output bus.

1.1 Signal description

Table 1 describes all the small page NAND Flash signals. *Figure 1* shows the logical diagram of NAND Flash.

Table 3 describes the microcontroller signals used to connect to the small page NAND devices.

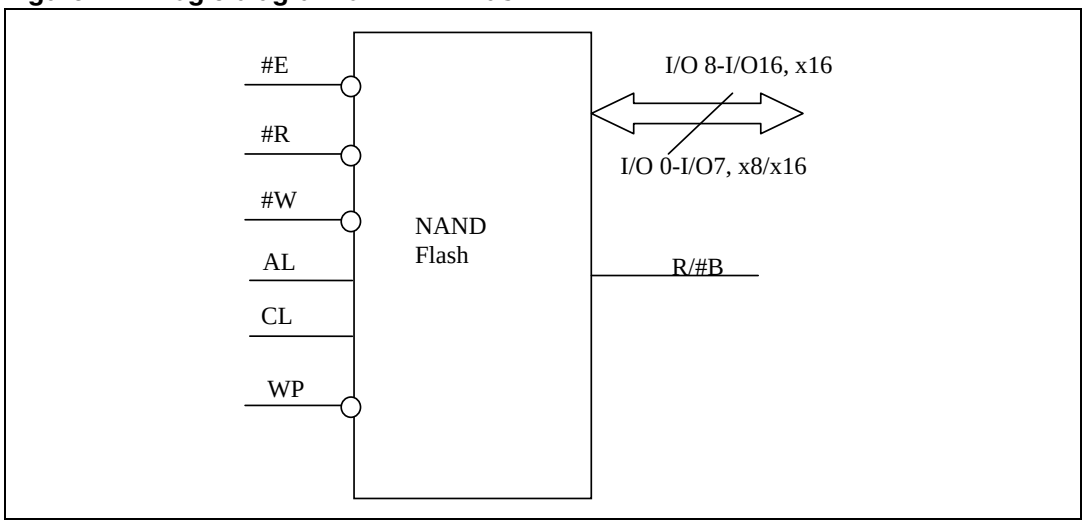
Table 1. Small page NAND Flash signal description

Signal	Signal name	Description
I/O8-15	Data input/outputs	Input/outputs 8 to 15 are only available in x16 devices. They are used to output the data during a read operation or input data during a write operation.
I/O0-7	Data input/outputs	Input/outputs 0 to 7 are used to input the selected address output the data during a read operation or input a command or data during a write operation. The inputs are latched on the rising edge of write enable. I/O0-I/O7 can be left floating when the device is deselected or the outputs are disabled.
AL	Address latch enable	The address latch enable activates the latching of the address inputs in the command Interface. When AL is high, the inputs are latched on the rising edge of write enable.
CL	Command latch enable	The command latch enable activates the latching of the command inputs in the Command Interface. When CL is high, the inputs are latched on the rising edge of write enable.
#E	Chip enable	The chip enable input activates the memory control logic, input buffers, decoders and sense amplifiers. When chip enable is low (VIL) the device is selected.
#R	Read enable	The Read Enable controls the sequential data output during Read operations. Data is valid after the falling edge of R. The falling edge of R also increments the internal column address counter by one.
R/#B	Ready/busy	The Ready/Busy output is an open-drain output that can be used to identify if the P/E/R controller is currently active. When ready/busy is low (VOL), a read, program or erase operation is in progress. When the operation completes, ready/busy goes high (VOH). The use of an open-drain output allows the ready/busy pins from several memories to be connected to a single pull-up resistor. A low then indicates that one, or more, of the memories is busy.
#W	Write enable	The write enable input controls writing to the command interface, input address and data latches. Both addresses and data are latched on the rising edge of write enable.
#WP	Write protect	The write protect pin is input that gives hardware protection against unwanted program or erase operations. When write protect is low (VIL) the device does not accept any program or erase operations.

Table 1. Small page NAND Flash signal description (continued)

Signal	Signal name	Description
Vcc	Supply voltage	VCC provides the power supply to the internal core of the memory device. It is the main power supply for all operations (read, program and erase). It is in the range of 1.65-1.95 V for 1.8 V devices and 2.7-3.6 V for 3 V devices.
Vss	Ground	Ground is the reference for the power supply. It must be connected to the system ground.

Figure 1. Logic diagram of NAND Flash



1.2 Memory array organization

The memory array is organized in blocks where each block contains 32 pages. The array is split into two areas, the main area and the spare area. The main area of the array is used to store data, whereas the spare area is typically used to store error correction codes, software flags or bad block identification.

In x8 devices the pages are split into a main area with two half pages of 256 Bytes each and a spare area of 16 Bytes. In the x16 devices the pages are split into a 256 Word main area and an 8 Word spare area.

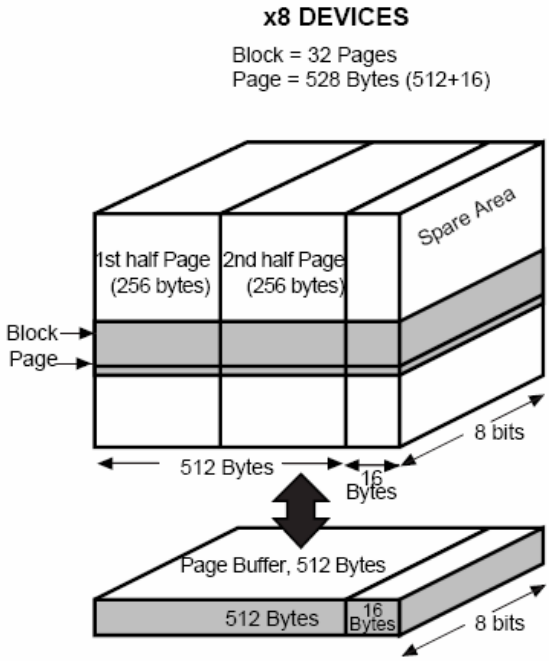
In this application note the communication with x8 devices is explained. Thus the memory map is divided as:

BLOCKS each having 32 pages and each page in turn having 528 Bytes. The number of BLOCKS is decided on the basis of the NAND Flash we are using. This application note explains the communication using NAND512W3A NAND Flash.

The memory organization for NAND512W3A is as follows:

4096 BLOCKS, 32 pages in each BLOCK, 528 Bytes in each page. *Figure 2* shows the memory arrangement for x8 devices.

Figure 2. Memory arrangement for x8 NAND Flash



Thus to access the specific location in Flash, the user must address the column number, page number and the block number. The addressing is done through the address insertion on the data line. The address cycles for x8 devices are shown in *Table 2*.

Table 2. Address insertion, x8 devices

Bus cycle	I/O7	I/O6	I/O5	I/O4	I/O3	I/O2	I/O1	I/O0
1st	A7	A6	A5	A4	A3	A2	A1	A0
2nd	A16	A15	A14	A13	A12	A11	A10	A9
3rd	A24	A23	A22	A21	A20	A19	A18	A17
4th	VIL	VIL	VIL	VIL	VIL	VIL	A26	A25

Note: A8 is don't care in x16 devices

any additional address input cycles are ignored

the 4th cycle is only required for 512 Mb and 1Gb devices

2 Hardware interface with microcontroller

NAND Flash is connected to the microcontroller through the I/O ports. All the signals of the NAND Flash are connected to the I/O ports of the microcontroller. In this application note we have used the ST72651AR6 microcontroller. The I/O pins and their connections with NAND are explained in *Table 3*.

Table 3. Port mapping of microcontroller and NAND

Pins (port of microcontroller)	NAND pin
PB0-PB7 (Port B)	I/O0-I/O7
PA0 (Port A - pin0)	CLE
PA1 (Port A - pin1)	#WE
PA2 (Port A – pin2)	ALE
PA3 (Port A – pin3)	#R
PA4 (Port A – pin4)	R/#B
PA7 (Port A – pin7)	#WP
PE0 (Port E – pin0)	#CE

3 Firmware

The example firmware for NAND communication through I/O's is organized in two files, nand_io.c and nand_io.h. In the header file, nand_io.h, there are function prototypes. In the file nand_io.c, all the functions to use the NAND Flash device are implemented. All the source files are in 'C' language and the application uses ST7 firmware library functions. The source files are only for guidance. STMicroelectronics shall not be liable for any direct, indirect or consequential damages with respect to any claim arising from use of this software.

3.1 NAND basic functions

There are 5 basic operations in NAND Flash which control its functionality:

1. Command Input: command (command type).
   Command input bus operations are used to give commands to the memory. Commands are accepted when chip enable is low, command latch enable is high, address latch enable is low and read enable is high. All bus write operations to the device are interpreted by the memory command interface. The commands are input on I/O0-I/O7 and are latched on the rising edge of write enable when the command latch enable signal is high. Device operations are selected by writing specific commands to the command register. The command registers are summarized in *Table 4*. *Figure 3* shows the signal status at different lines for command input operation.

Table 4. Command register

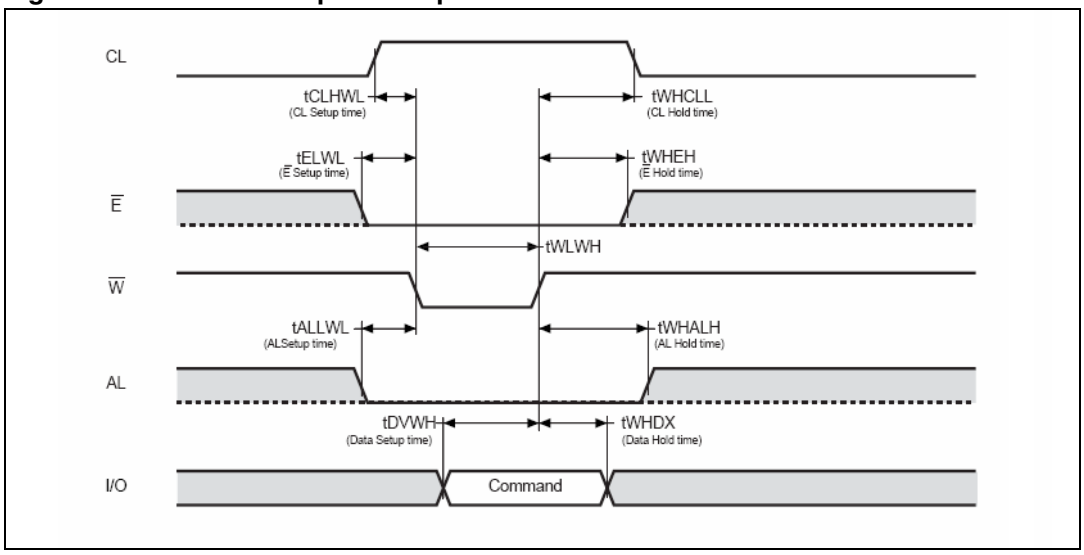
Command	Bus write operations(1)			Command accepted during busy
	1st cycle	2nd cycle	3rd cycle	
Read A	00h			
Read B	01h(2)			
Read C	50h			
Read electronic signature	90h			
Read status register	70h			Yes
Page program	80h	10h		
Copy back program	00h	8Ah	10h	
Block erase	60h	D0h		
Reset	FFh			Yes

1. The bus cycles are only shown for issuing the codes. The cycles required to input the addresses or input/output data are not shown.
2. Any undefined command sequence is ignored by the device.

Inside the header file, nand_io.h, the command types are defined as:

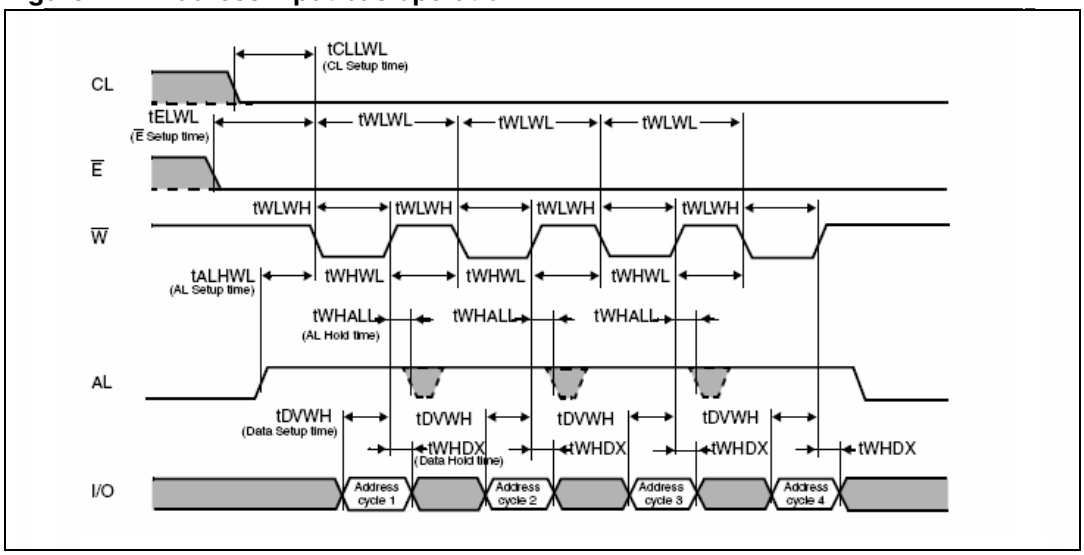
- #define nand_area_A; 0x00
- #define nand_area_B; 0x01
- #define nand_area_C; 0x50
- #define nand _readstatusreg; 0x70
- #define nand _pageprogram; 0x80
- #define nand _endpageprogram; 0x10
- #define nand _read_electsign; 0x90
- #define nand _blockerase; 0x60
- #define nand _confirmerase; 0xD0
- #define nand _reset; 0xFF

Figure 3. Command input bus operations



2. Address input cycle: address (column address, page address, block address).
Address Input bus operations are used to input the memory address. Three bus cycles are required to input the addresses for the 128 Mb and 256 Mb devices and four bus cycles are required to input the addresses for the 512 Mb and 1 GB devices (refer *Table 1*). Thus the user can access any specific location in the memory by giving its column location, page location and block number. There are 256 columns in both the first half page (area A) and the second half page (area B). The total number of pages is 32 in each block and the total number of blocks in NAND 512W3A is 4096. *Figure 4* shows the signal status during the address transmission.

Figure 4. Address input bus operation

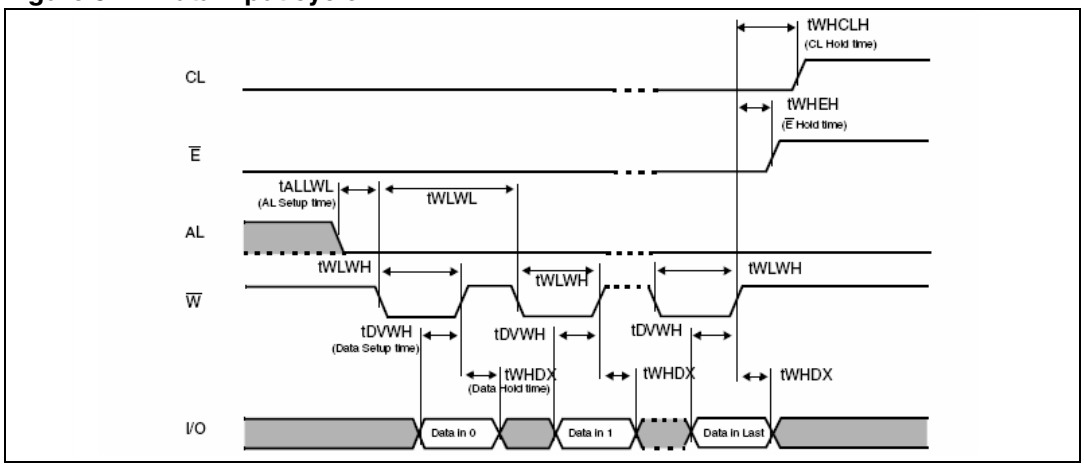


3. Data input cycle: nand_write (unsigned char *data, page number, column number, block number, number of bytes).
Data Input bus operations are used to input the data to be programmed. Data is accepted only when chip enable is low, address latch is low, command latch enable is low and read enable is high. The data is latched on the rising edge of the write enable signal. The data is input sequentially using the write enable signal. This function is

called nand writing. The user must pass the address of the variable which stores the data which is to be written in NAND. User can store the data bulk in an array in the main program and then finally pass the array pointer to this function for writing.

Note: NAND Flash supports the page write mode. Thus for the same page only three consecutive write cycles are allowed and after that the user has to erase the whole block before being able to write again. (Refer datasheet of NAND). Signal status at the time of sending the data is shown in Figure 5.

Figure 5. Data input cycle



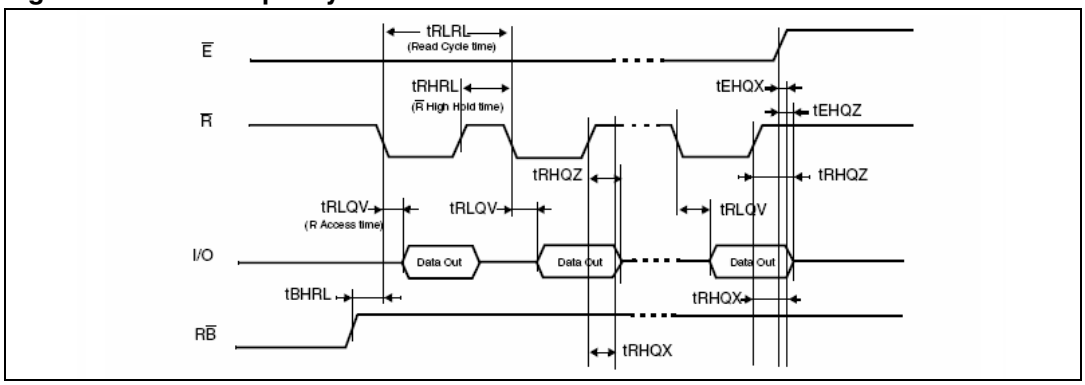
4. Data output cycle.
Data output bus operations are used to read the data in the memory array, the status register, electronic signature and serial number. Data is output when chip enable is low, write enable is high, address latch enable is low, and command latch enable is low. The data is output sequentially using the read enable signal.

   a) nand_read (bytes, page number, block number).
   This function is used to read bytes from the specified page of a specified block. The user must pass the number of bytes to be read (0-512 bytes), the page number from where to read and also the block number which is to be addressed. With the user specified information, this function reads the page data and stores it in an array (data_read[ ]). The user can read this array in his main program to get the data received from NAND.

   b) signature ().
   This function when called reads the electronic signature and serial number of the device. These data are stored in the variable named as manufacturer_code for electronic signature and device_code for serial number.

   c) statusreg_read().
   This function is used to read the status register of NAND and return the status. *Figure 6* shows the different signal status for data output cycle.

Figure 6. Data output cycle



5. Write protect: writeprotect_enable () and writeprotect_disable ().
   This cycle is used to protect the device from any write and erase operation. When the write protect signal is low, then the device is write protected. Calling the writeprotect_enable function protects the device from any access to the information stored in it, while writepotect_disable function makes the device accessible.

3.2 Remaining NAND functions

There are additional functions used in the NAND- I/O communication:

1. nand_io_init(): this function is used to initialize the I/O's to the default state for communication with NAND Flash.
2. readpulses(): this function is used to generate the pulsating read pulses in the case of nand read mode.
3. writepulses(): this function is used to generate write pulses in the case of writing to NAND Flash.
4. block_erase(block number): this function is used to erase the complete block. The user needs to enter the block number which is to be erased.
5. bad_block_decleration(block number): this function is used to declare the block as bad block. The user must pass the block number which is to be declared as bad block.

4 References

1. STNAND512W3A datasheet
2. ST7 software library user manual

5 Revision history

Table 5. Document revision history

Date	Revision	Changes
18-Oct-2007	1	Initial release