NATIONAL INSTITUTE FOR NUCLEAR AND HIGH ENERGY PHYSICS

December 2002
ET 22920.X
revision may 2003

Notes on the lock and fuse bits of the Atmega128

When the lock and fuse bits are programmed incorrectly, the device may lock-up permanently. This note tries to clarify the meaning of those bits and how to use the in circuit programmers correctly.

H.L. Groenstege
J.J. Kuijt

mailto: h.groenstege@nikhef.nl

http://www.nikhef.nl/pub/departments/et/misc/atmega128/fuse_lock.pdf

NIKHEF, DEPARTMENT OF ELECTRONIC TECHNOLOGY
P.O.box 41882, NL 1009 DB AMSTERDAM

Contents

Revision may 2003

Additional information on page 7 for the Pony programmer settings.
Link added to new version of the ATMega128 plug on board.

Lock and fuse bits

Great care must be taken about the *lock* and *fuse bits* when programming a device. Once some wrong pattern has been programmed, the device is no longer accessible (permanently).

When using BasCom to program the device, the fuse bits are read automatically and one can change those needed. Then the pattern is written back after the write button is hit.

The Pony-programmer however must be instructed to read the bits from the device manually. Then apply your changes and write back the pattern. Here things easily go wrong. For commonly used settings see page 7.

In both cases it is possible to lock up the device permanently. In some cases applying an external clock to XTAL1 pin enables access again.

A copy of the Atmega documentation can be found at:
http://www.nikhef.nl/pub/departments/et/misc/atmega128/atmega128_dat.pdf

Settings

The values in ***bold italic*** are the default (factory) settings. The table refers to the names used in the Atmel documentation (as used by the Pony-programmer) and, in **bold**, the BasCom convention. The latter refers to the position of the bits in the byte programmed. A marked box (✓) in the Pony-programmers' *configurations and security bits* section means a '0' (programmed).

Boot Lock bits

Boot Lock bits		
BLB		**access control**
12	11	
5	**4**	
0	0	SPM is not allowed to write in the boot-loader section and LPM executing from the application section is not allowed to read from the boot-loader section. If interrupt vectors are placed in the application section, interrupts are disabled while executing from the boot-loader section.
0	1	LPM executing from the application section is not allowed to read from the boot-loader section. If interrupt vectors are placed in the application section, interrupts are disabled while executing from the boot-loader section.
1	0	SPM is not allowed to write to the boot-loader section.
1	***1***	No restrictions for SPM or LPM accessing the boot-loader section
Atmega128 documentation page 271		

Boot Lock bits		
BLB		**access control**
02	01	
3	**2**	
0	0	SPM is not allowed to write in the application section, LPM executing from the Boot Loader section is not allowed to read from the Application section. If interrupt vectors are placed in the Boot Loader section, interrupts are disabled while executing from the Application section.
0	1	LPM executing from the boot-loader section is not allowed to read from the application section. If interrupt vectors are placed in the boot-loader section, interrupts are disabled while executing from the application section.
1	0	SPM is not allowed to write to the application section
1	***1***	No restrictions for SPM or LPM accessing the application section
Atmega128 documentation page 272		

Lock bits		
LB		**Access control**
02	01	
1	**0**	
0	0	Further programming and verification of the Flash and EEPROM is disabled in Parallel and SPI/JTAG Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode.
1	0	Further programming of the Flash and EEPROM is disabled in Parallel and SPI/JTAG Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode.
1	***1***	No memory lock features enabled.

Fuse bits low

Fuse bits low		
SUT		**Start-up time**
1	0	
9	**8**	
1	***0***	See Atmega128 documentation page 36, 39

Fuse bits low			
BODlevel	**7**	0	
		1	Brown-out detector level: Vcc = 2.7 V
BODen	**6**	0	
		1	Disable brown-out detector

Fuse bits low				
CKSEL				**Device clocking options**
3	2	1	0	
D	**C**	**B**	**A**	
0	0	0	0	External clock
0	***0***	***0***	***1***	1 MHz internal RC oscillator
0	1	0	0	
0	1	0	1	External RC oscillator

1	0	0	0	
1	0	0	1	External low frequency crystal
1	0	1	0	External crystal or ceramic resonator (see oscillator operating modes)

1	1	1	1	
For details of the clock settings (and CKOPT) see Atmega128 documentation page 34				

Fuse bits low				
CKSEL				**Oscillator operating modes** Frequency in MHz and load capacitor values
3	2	1	0	
D	**C**	**B**	**A**	
1	0	1	0	1.0 … max. frequency, 12 … 22 pF.
1	1	0	0	
1	1	1	0	
1	0	1	1	0.4 … 0.9 MHz, no capacitors (resonators?)
1	1	0	1	0.9 … 3.0 MHz, 12 … 22 pF
1	1	1	1	3.0 … 8.0 (?) MHz, 12 … 22 pF
CKSEL0 is used in the *start up time* setting see Atmega128 documentation page 36				

Fuse bits High

Fuse bits High			
OCDEN	**E**	0	Enables system clocks in sleep mode (increased power consumption)
		1	OCD disabled
JTAGEN	**F**	***0***	JTAG enabled
		1	JTAG disabled
SPIEN	**G**	***0***	Enable serial downloading (!)
		1	Disable serial downloading (!)
CKOPT	**H**	0	Full swing oscillator signal for noise immunity
		1	Oscillator options (depends on Clock source, CKSEL bits) *
EESAVE	**I**	0	EEPROM preserved when erasing chip
		1	EEPROM erased when erasing chip
BOOTRST	**M**	0	
		1	Reset vector: $0000 ??
*: For details of the clock settings see Atmega128 documentation page 34			

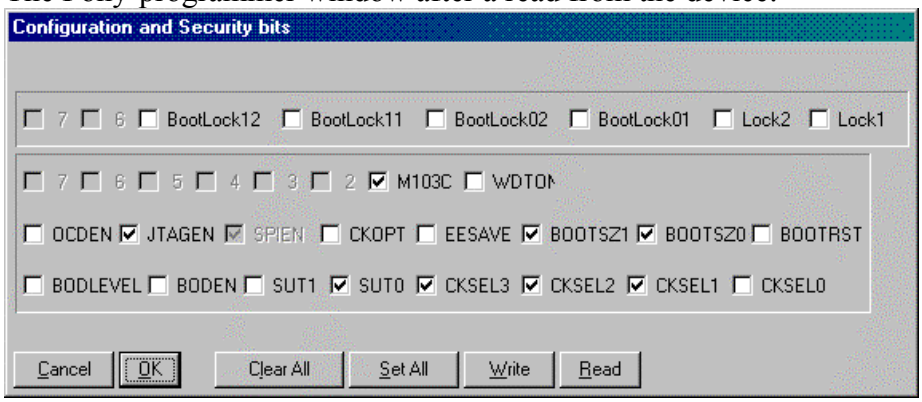
Fuse bits High		
BOOTSZ1	BOOTSZ0	**Select bootsize**
K	**L**	
0	***0***	For details see Atmega128 documentation page 280

Fuse bits Extended

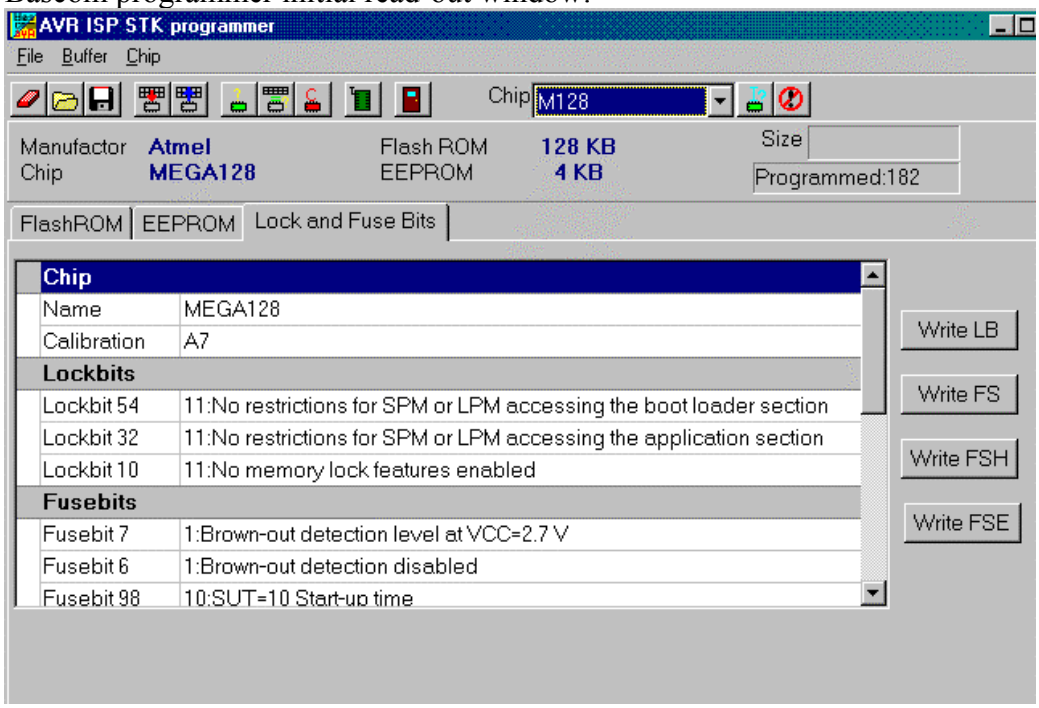
Fuse bits Extended				
M103C	**P**	***0***	Atmega103 compatibility mode	For details see Atmega128 documentation page 4
		1	Atmega128 mode	
WDTON	**Q**	0	Watchdog timer on	For details see Atmega128 documentation page 51
		1	Watchdog timer not programmed	

Default (factory) settings

The Pony-programmer window after a read from the device:



Bascom programmer initial read-out window:



The other bits (*Fusebits* should be *fusebits low*):

Fusebits	
Fusebit 7	1:Brown-out detection level at VCC=2.7 V
Fusebit 6	1:Brown-out detection disabled
Fusebit 98	10:SUT=10 Start-up time
Fusebit DCBA	0001:CKSEL=0001 Internal RC Ocsillator 1 MHz
Fusebits High	
Fusebit E	1:Disable OCD
Fusebit F	0:Enable JTAG
Fusebit G	0:Enable serial downloading
Fusebit H	1:osc 1
Fusebit I	1:EEPROM memory is erased when erasing chip
Fusebit KL	00:Bootsize 4096 words at $F000
Fusebit M	1:Reset vector is $0000
Fusebits Extended	
Fusebit P	0:ATMEGA103 compatibility mode set
Fusebit Q	1:Watchdog timer not programmed

Settings for the ATMega128 board

The ATMega128 prototype board is described over here:
http://www.nikhef.nl/pub/departments/et/misc/atmega128/atmega128_wrd.pdf
The production version board is described over here:
http://www.nikhef.nl/pub/departments/et/misc/atmega128/atmega128v2.pdf

For normal use of the plug-on board, the factory settings for most of the bits are fine. A few bits need to be changed:

- The clock selection (default 0001) should be external crystal: CKSEL = 111X.
- JTAG should be disabled to free the port pins for general use.
- The M103 compatibility mode should be disabled to make full use of the M128 capabilities.

For the Bascom programmer this is easy to change/read. This, because it shows the current setting; bit-pattern and meaning.

Keep in mind that for the Pony-programmer you explicitly have to *read* the pattern first (hit the button). Then it indicates programmed bits (set to zero) with a marked box (✓).

So, it should look like this before you hit *write*.