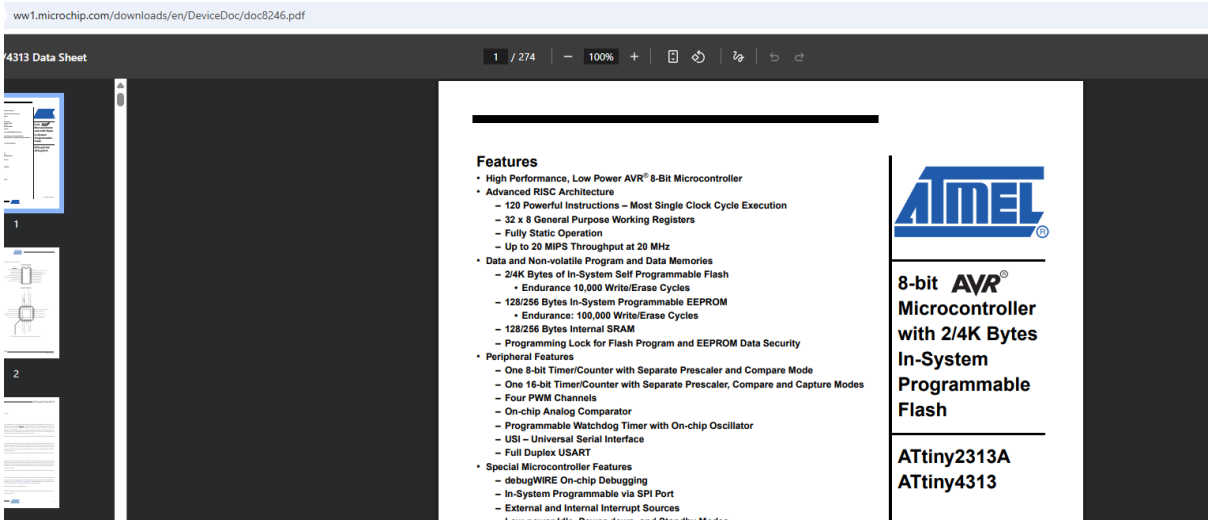
# AVR Description File Setup Guide

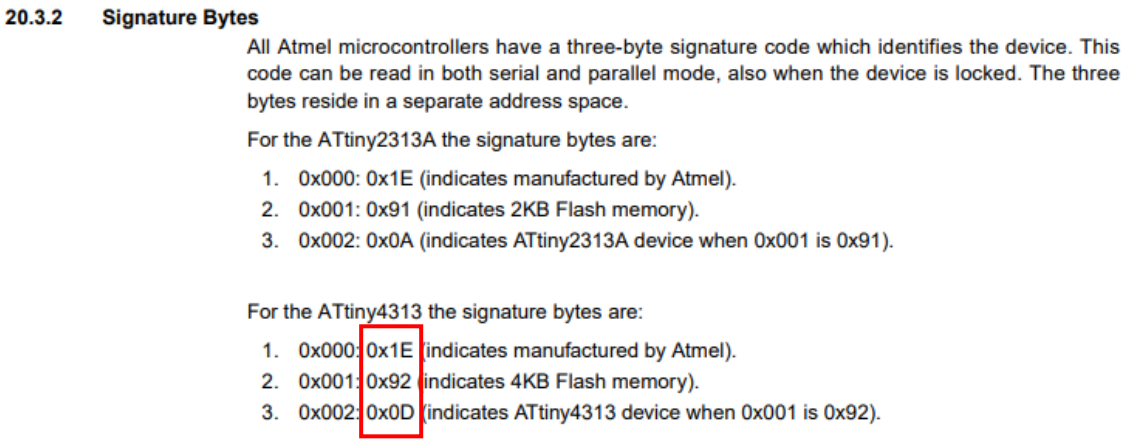
Follow these steps if you are using an AVR that is not supported on the Github repository.

For the purposes of this demonstration. We will be looking at the ATtiny4313 and we will be using the built in Microsoft notepad text editor to make our file.

**1) Acquire the complete datasheet of your selected AVR. You may see a summary version. Do not use that.**



**2) If on windows press F3 and pull up the search function and type “signature bytes”. You should scroll down until you see the information for your selected chip. It may be in a table or in a numbered bullet pointed format. You are looking for three hex values as can be seen in the highlighted square below. Your values will be different. But the order in which they are presented should be the same.**



**20.3.2 Signature Bytes**

All Atmel microcontrollers have a three-byte signature code which identifies the device. This code can be read in both serial and parallel mode, also when the device is locked. The three bytes reside in a separate address space.

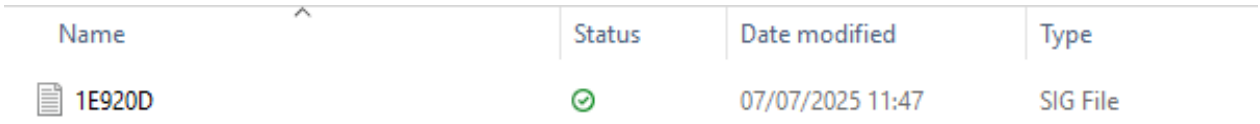
For the ATtiny2313A the signature bytes are:

1. 0x000: 0x1E (indicates manufactured by Atmel).
2. 0x001: 0x91 (indicates 2KB Flash memory).
3. 0x002: 0x0A (indicates ATtiny2313A device when 0x001 is 0x91).

For the ATtiny4313 the signature bytes are:

1. 0x000: 0x1E (indicates manufactured by Atmel).
2. 0x001: 0x92 (indicates 4KB Flash memory).
3. 0x002: 0x0D (indicates ATtiny4313 device when 0x001 is 0x92).

**3) Create a new text document and name it in the same order as I have for this example. In this case the order in which I have named my file follows the numbering scheme in which Atmel have listed their signature bytes. Following their numbering scheme 1,2,3. I have named it 1E920D. Omitting the 0x term for each value and giving the file a “. SIG” extension. This is to differentiate it from the other file types that the device uses.**



| Name | Status | Date modified | Type |
|---|---|---|---|
| 1E920D | ✓ | 07/07/2025 11:47 | SIG File |

**4) Open the file you have just created and have your datasheet side by side. You should be able to locate a table like what I have below. So you can reference the right information.**

### 21.1 Memory Parametrics

Flash memory parametrics are summarised in Table 21-1, below.

**Table 21-1.** No. of Words in a Page and No. of Pages in the Flash

| Device | Flash Size | Page Size | PCWORD(1) | Pages | PCPAGE | PCMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 1K word (2K bytes) | 16 words | PC[3:0] | 64 | PC[9:4] | 9 |
| ATtiny4313 | 2K words (4K bytes) | 32 words | PC[4:0] | 64 | PC[10:5] | 10 |

Note: 1. See Table 19-1 on page 174.

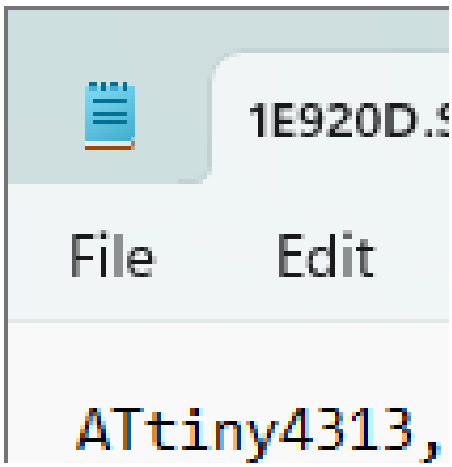
EEPROM parametrics are summarised in Table 21-2, below.

**Table 21-2.** No. of Words in a Page and No. of Pages in the EEPROM

| Device | EEPROM Size | Page Size | PCWORD(1) | Pages | PCPAGE(1) | EEAMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 128 bytes | 4 bytes | EEA[1:0] | 32 | EEA[6:2] | 6 |
| ATtiny4313 | 256 bytes | 4 bytes | EEA[1:0] | 64 | EEA[7:2] | 7 |

Note: 1. See Table 19-1 on page 174.

**5)The first text you need to enter is the chips name followed by a comma. This name minus the comma itself is what will be seen on the screen when backing up or flashing the AVR and what the device will save the file as on the micro-SD card. The max number of characters that can be seen on your screen is 11. So have that in mind. In this case I have type "ATtiny4313,".**



**6)The second piece of information we need to enter is the flash memory size in kb. In our case I can see this device has a 4kb flash size. Therefore, I will add "4," to our current line.**

### 21.1 Memory Parametrics

Flash memory parametrics are summarised in Table 21-1, below.

**Table 21-1.** No. of Words in a Page and No. of Pages in the Flash

| Device | Flash Size | Page Size | PCWORD(1) | Pages | PCPAGE | PCMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 1K word (2K bytes) | 16 words | PC[3:0] | 64 | PC[9:4] | 9 |
| ATtiny4313 | 2K words (4K bytes) | 32 words | PC[4:0] | 64 | PC[10:5] | 10 |

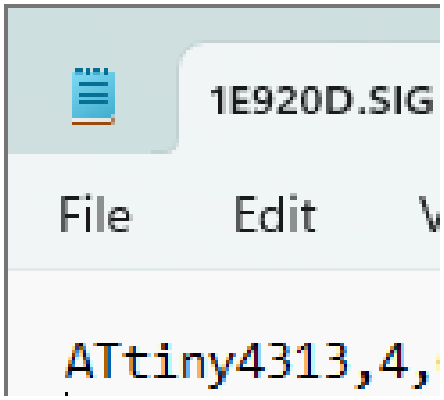
Note: 1. See Table 19-1 on page 174.

EEPROM parametrics are summarised in Table 21-2, below.

**Table 21-2.** No. of Words in a Page and No. of Pages in the EEPROM

| Device | EEPROM Size | Page Size | PCWORD(1) | Pages | PCPAGE(1) | EEAMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 128 bytes | 4 bytes | EEA[1:0] | 32 | EEA[6:2] | 6 |
| ATtiny4313 | 256 bytes | 4 bytes | EEA[1:0] | 64 | EEA[7:2] | 7 |

Note: 1. See Table 19-1 on page 174.



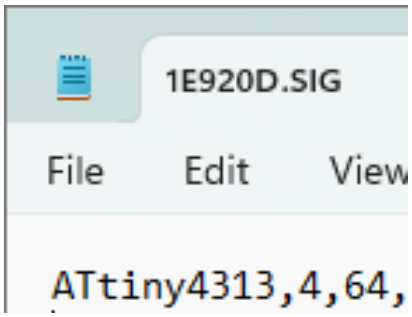
**7) We then require the flash page size in bytes. You can see below that the size is in words not bytes. All you need to do is double the value as 1 word equals two bytes. I have added "64," to my file.**

### 21.1 Memory Parametrics

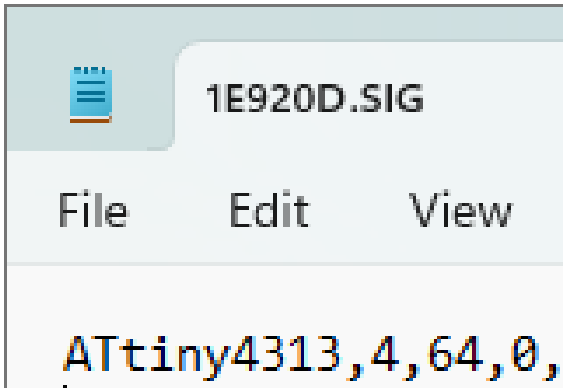
Flash memory parametrics are summarised in Table 21-1, below.

**Table 21-1.** No. of Words in a Page and No. of Pages in the Flash

| Device | Flash Size | Page Size | PCWORD(1) | Pages | PCPAGE | PCMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 1K word (2K bytes) | 16 words | PC[3:0] | 64 | PC[9:4] | 9 |
| ATtiny4313 | 2K words (4K bytes) | 32 words | PC[4:0] | 64 | PC[10:5] | 10 |




**8)Next, we need to tell the programmer if we want the writes cycles to be timed. Setting the value to 1 if we want to monitor the writing sequence flag or 0 if we use a delay instead. In this case we have opted for the delay to be used. Hence, I have added "0,".**



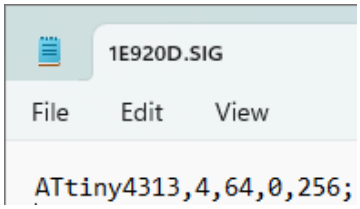
**9) Finally, we need to include the EEPROM size in bytes. Looking at the same table as before we can see the EEPROM for this AVR is 256 bytes in size. Therefore, I have added "256;". This time adding a semicolon instead of a comma to indicate the end of this description file.**

EEPROM parametrics are summarised in Table 21-2, below.

**Table 21-2.** No. of Words in a Page and No. of Pages in the EEPROM

| Device | EEPROM Size | Page Size | PCWORD(1) | Pages | PCPAGE(1) | EEAMSB |
|---|---|---|---|---|---|---|
| ATtiny2313A | 128 bytes | 4 bytes | EEA[1:0] | 32 | EEA[6:2] | 6 |
| ATtiny4313 | 256 bytes | 4 bytes | EEA[1:0] | 64 | EEA[7:2] | 7 |

Note: 1. See Table 19-1 on page 174.




**10) At the end of this guide you should have a .SIG file with the signature bytes as its name. Your file should contain the Name of the AVR, flash size in kb, flash page size in bytes, timed write enabled or disabled 1 or 0 respectively and finally the EEPROM size in bytes. Each separated by a comma with the last entry having a semicolon instead to complete the file.**

**11) Save the file in the root of the micro SD card.**

## !!! Warning !!!

The formatting of the text i.e. the commas and semicolon and the order in which the information has been shown in this guide must be followed. Otherwise, you will not be able to use this device.

As always, I strongly recommend a backup of all files should be made and put somewhere in your local storage. In the case of SD card corruption or accidental deletion.