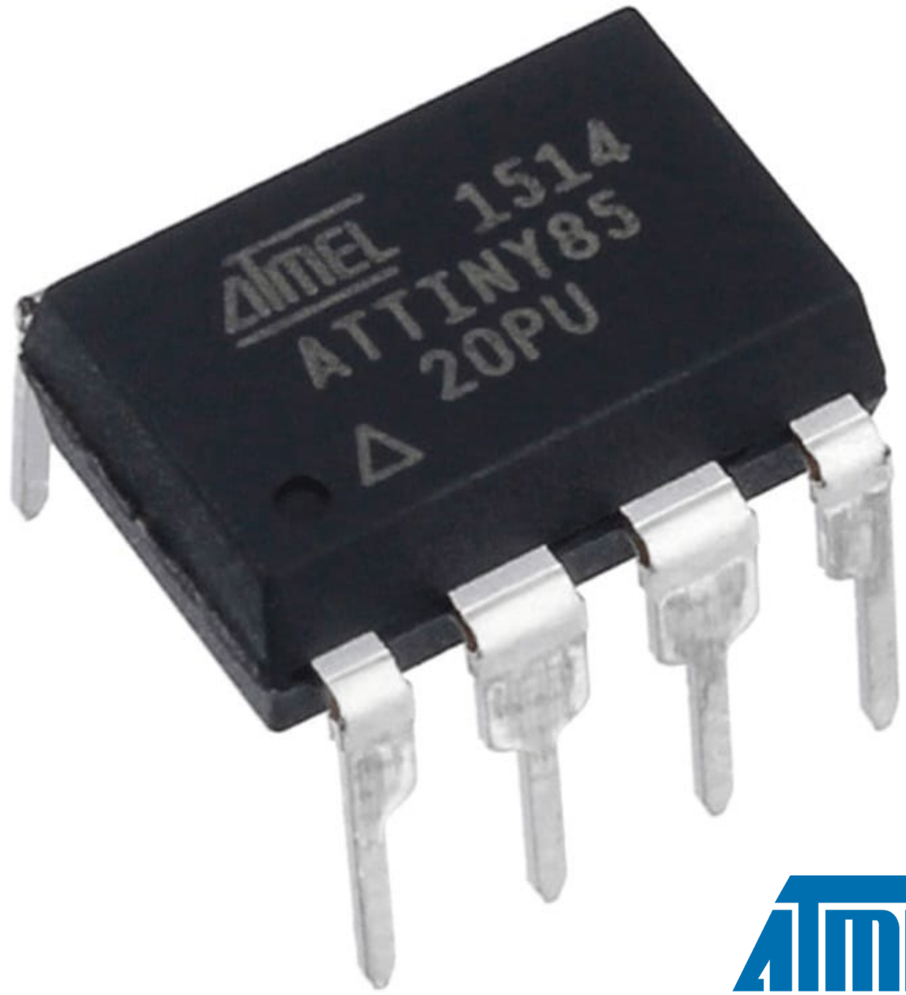


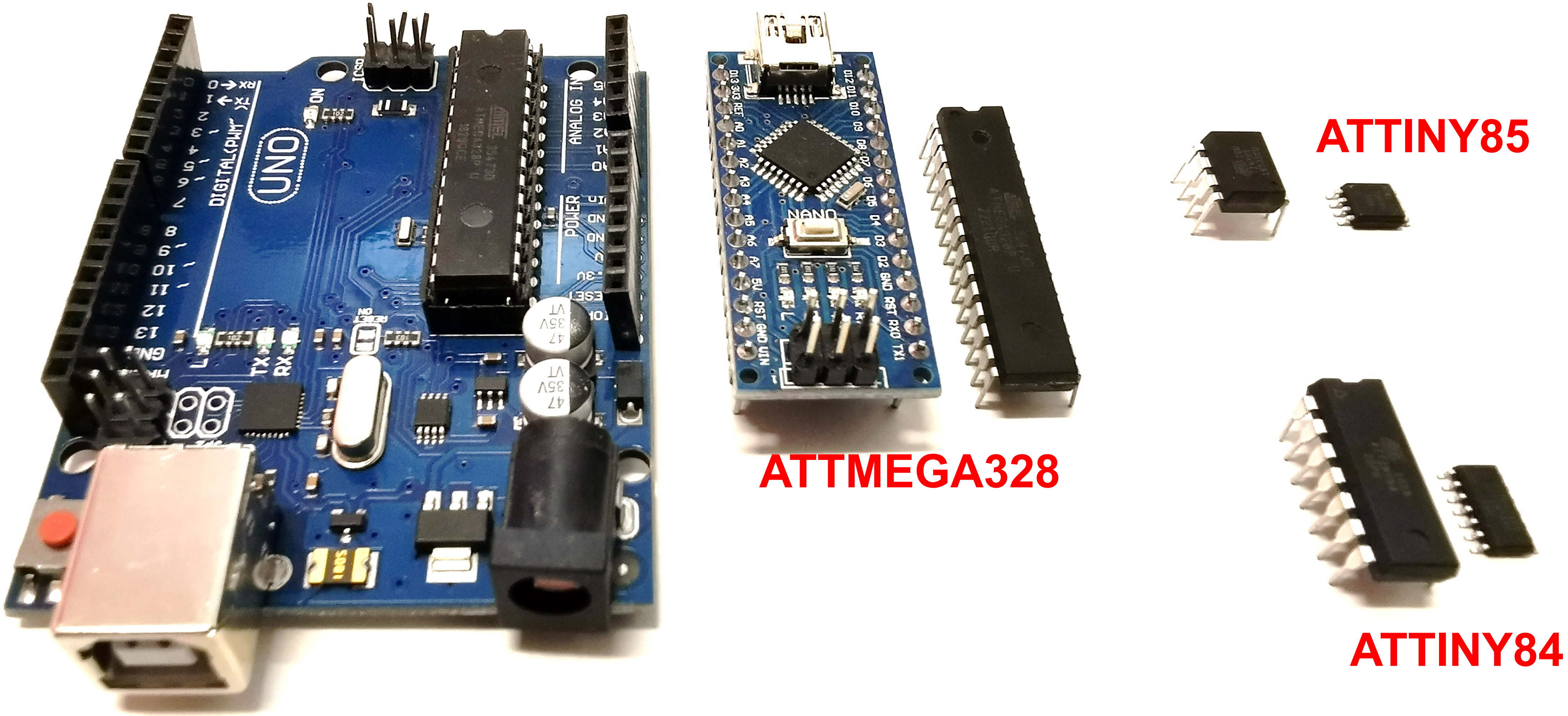
ATTINY85

Huber Girón Nieto

IBERO PUEBLA ®

Departamento de Ciencias e Ingenierías /

Instituto de Diseño e Innovación Tecnológica IDIT /


ATTINY85
ATTMEGA328
ATTINY84

Atmel®

Atmel 8-bit AVR Microcontroller with 2/4/8K Bytes In-System Programmable Flash

ATtiny25/V / ATtiny45/V / ATtiny85/V Summary

Features

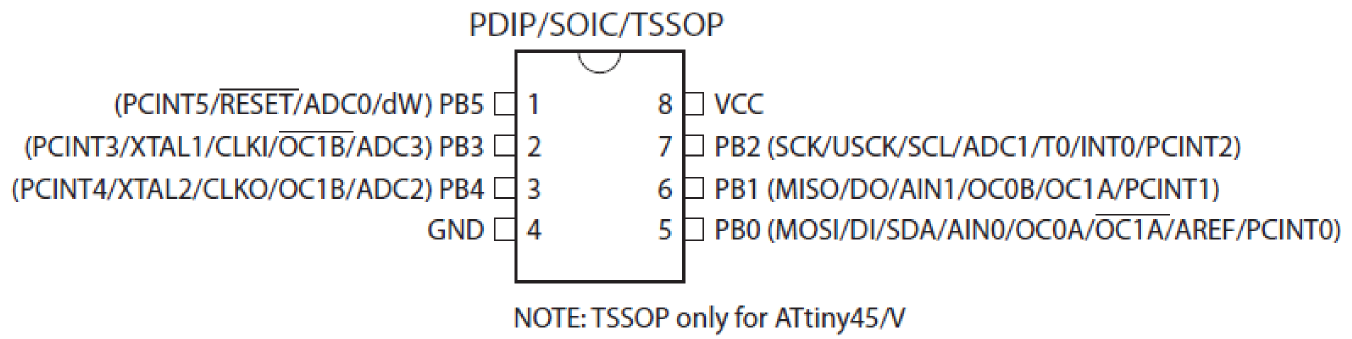
- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  – 120 Powerful Instructions – Most Single Clock Cycle Execution
  – 32 x 8 General Purpose Working Registers
  – Fully Static Operation
- Non-volatile Program and Data Memories
  – $\boxed{\text{2/4/8K Bytes of In-System Programmable Program Memory Flash}}$
    - Endurance: 10,000 Write/Erase Cycles
  – $\boxed{\text{128/256/512 Bytes In-System Programmable EEPROM}}$
    - Endurance: 100,000 Write/Erase Cycles
  – $\boxed{\text{128/256/512 Bytes Internal SRAM}}$
  – Programming Lock for Self-Programming Flash Program and EEPROM Data Security

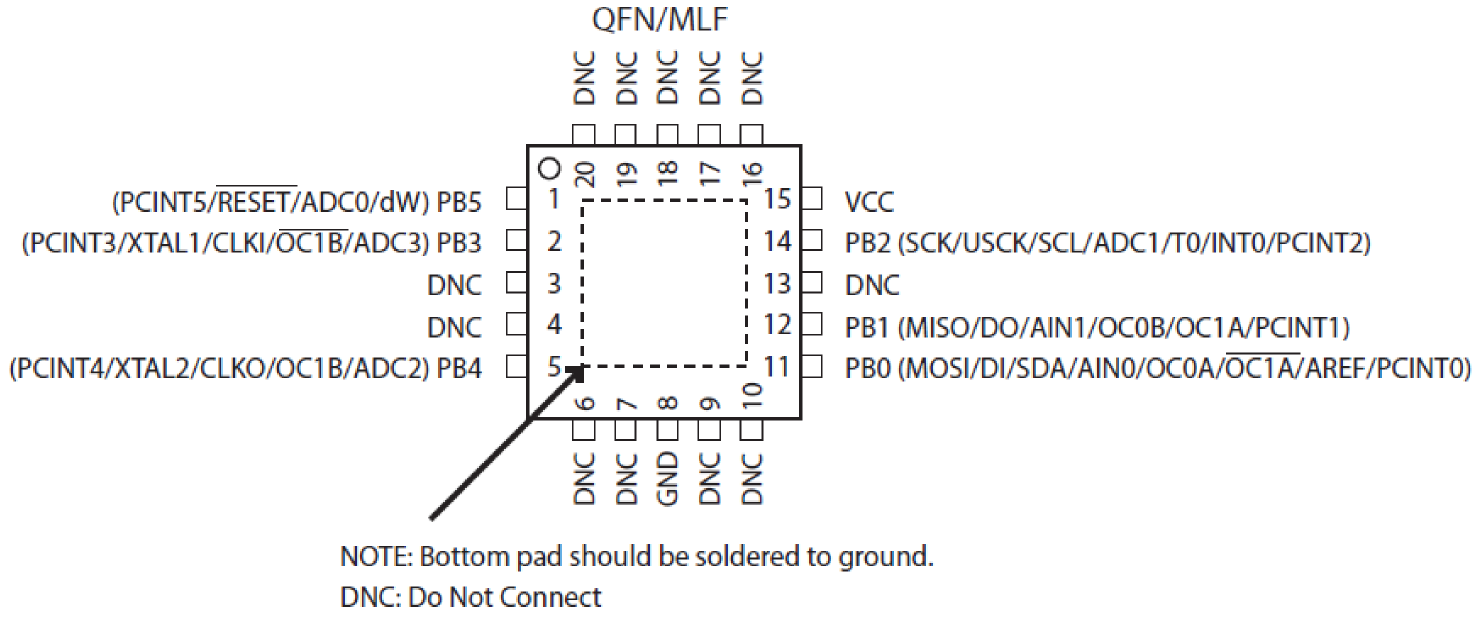
- Peripheral Features
  - 8-bit Timer/Counter with Prescaler and Two PWM Channels
  - 8-bit High Speed Timer/Counter with Separate Prescaler
    - 2 High Frequency PWM Outputs with Separate Output Compare Registers
    - Programmable Dead Time Generator
  - USI – Universal Serial Interface with Start Condition Detector
  - 10-bit ADC
    - 4 Single Ended Channels
    - 2 Differential ADC Channel Pairs with Programmable Gain (1x, 20x)
    - Temperature Measurement
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - debugWIRE On-chip Debug System
  - In-System Programmable via SPI Port
  - External and Internal Interrupt Sources
  - Low Power Idle, ADC Noise Reduction, and Power-down Modes
  - Enhanced Power-on Reset Circuit
  - Programmable Brown-out Detection Circuit
  - Internal Calibrated Oscillator

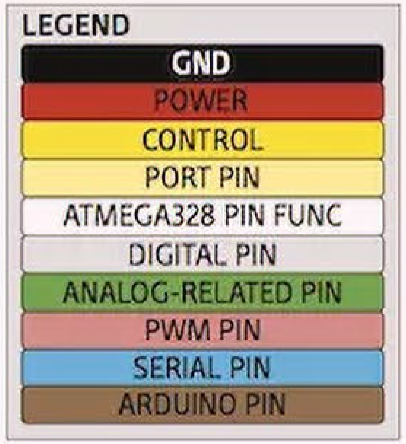
- I/O and Packages
  - Six Programmable I/O Lines
  - 8-pin PDIP, 8-pin SOIC, 20-pad QFN/MLF, and 8-pin TSSOP (only ATtiny45/V)
- Operating Voltage
  - 1.8 - 5.5V for ATtiny25V/45V/85V
  - 2.7 - 5.5V for ATtiny25/45/85
- Speed Grade
  - ATtiny25V/45V/85V: 0 – 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V
  - ATtiny25/45/85: 0 – 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- Industrial Temperature Range
- Low Power Consumption
  - Active Mode:
    - 1 MHz, 1.8V: 300 µA
  - Power-down Mode:
    - 0.1 µA at 1.8V

1. Pin Configurations

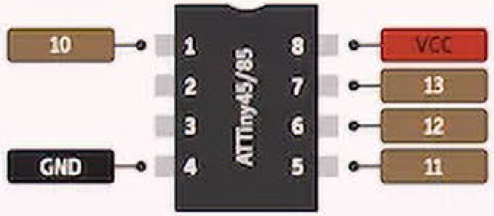
Figure 1-1. Pinout ATtiny25/45/85

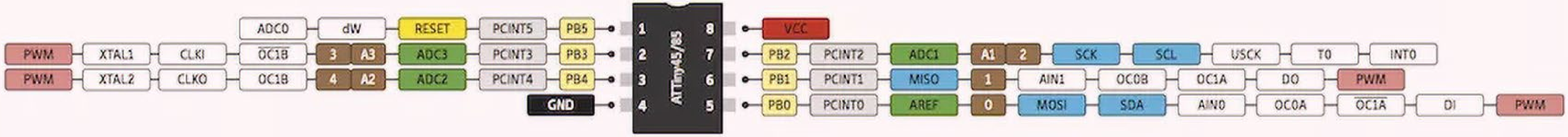

PDIP/SOIC/TSSOP
(PCINT5/$\overline{\text{RESET}}$/ADC0/dW) PB5 1
(PCINT3/XTAL1/CLKI/$\overline{\text{OC1B}}$/ADC3) PB3 2
(PCINT4/XTAL2/CLKO/OC1B/ADC2) PB4 3
GND 4
8 VCC
7 PB2 (SCK/USCK/SCL/ADC1/T0/INT0/PCINT2)
6 PB1 (MISO/DO/AIN1/OC0B/OC1A/PCINT1)
5 PB0 (MOSI/DI/SDA/AIN0/OC0A/$\overline{\text{OC1A}}$/AREF/PCINT0)
NOTE: TSSOP only for ATtiny45/V


QFN/MLF
DNC DNC DNC DNC DNC
20 19 18 17 16
(PCINT5/$\overline{\text{RESET}}$/ADC0/dW) PB5 1
(PCINT3/XTAL1/CLKI/$\overline{\text{OC1B}}$/ADC3) PB3 2
DNC 3
DNC 4
(PCINT4/XTAL2/CLKO/OC1B/ADC2) PB4 5
15 VCC
14 PB2 (SCK/USCK/SCL/ADC1/T0/INT0/PCINT2)
13 DNC
12 PB1 (MISO/DO/AIN1/OC0B/OC1A/PCINT1)
11 PB0 (MOSI/DI/SDA/AIN0/OC0A/$\overline{\text{OC1A}}$/AREF/PCINT0)
6 7 8 9 10
DNC DNC GND DNC DNC
NOTE: Bottom pad should be soldered to ground.
DNC: Do Not Connect



Using Arduino as ICSP Programmer for ATTiny45/85



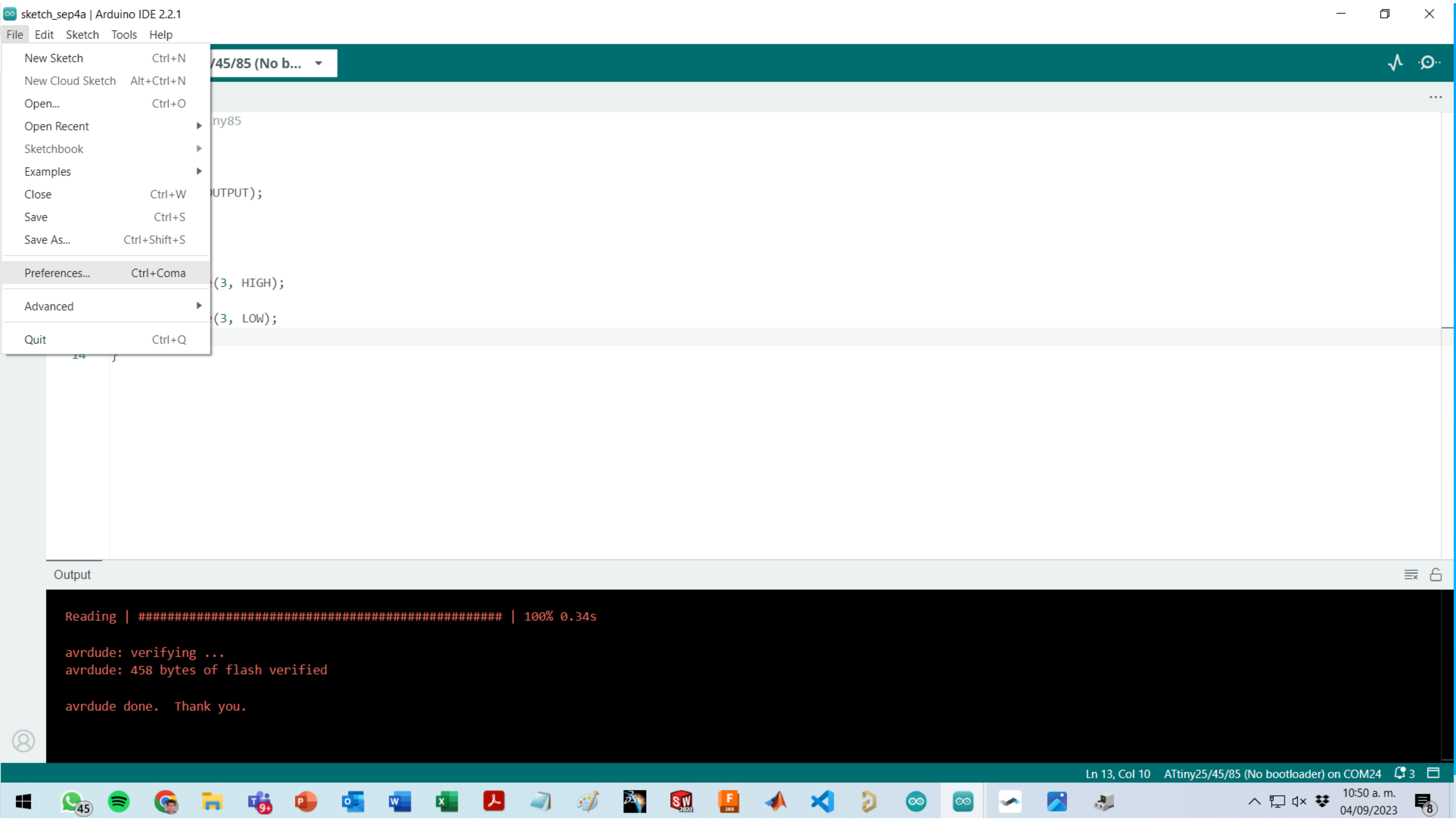

ATTiny45/85
ADC0 — dW — RESET — PCINT5 — PB5 — 1
PWM — XTAL1 — CLKI — $\overline{\text{OC1B}}$ — 3 — A3 — ADC3 — PCINT3 — PB3 — 2
PWM — XTAL2 — CLKO — OC1B — 4 — A2 — ADC2 — PCINT4 — PB4 — 3
GND — 4
8 — VCC
7 — PB2 — PCINT2 — ADC1 — A1 — 2 — SCK — SCL — USCK — T0 — INT0
6 — PB1 — PCINT1 — MISO — 1 — AIN1 — OC0B — OC1A — DO — PWM
5 — PB0 — PCINT0 — AREF — 0 — MOSI — SDA — AIN0 — OC0A — $\overline{\text{OC1A}}$ — DI — PWM

1. Instalar librería de Attiny

Opción 1 http://drazzy.com/package_drazzy.com_index.json

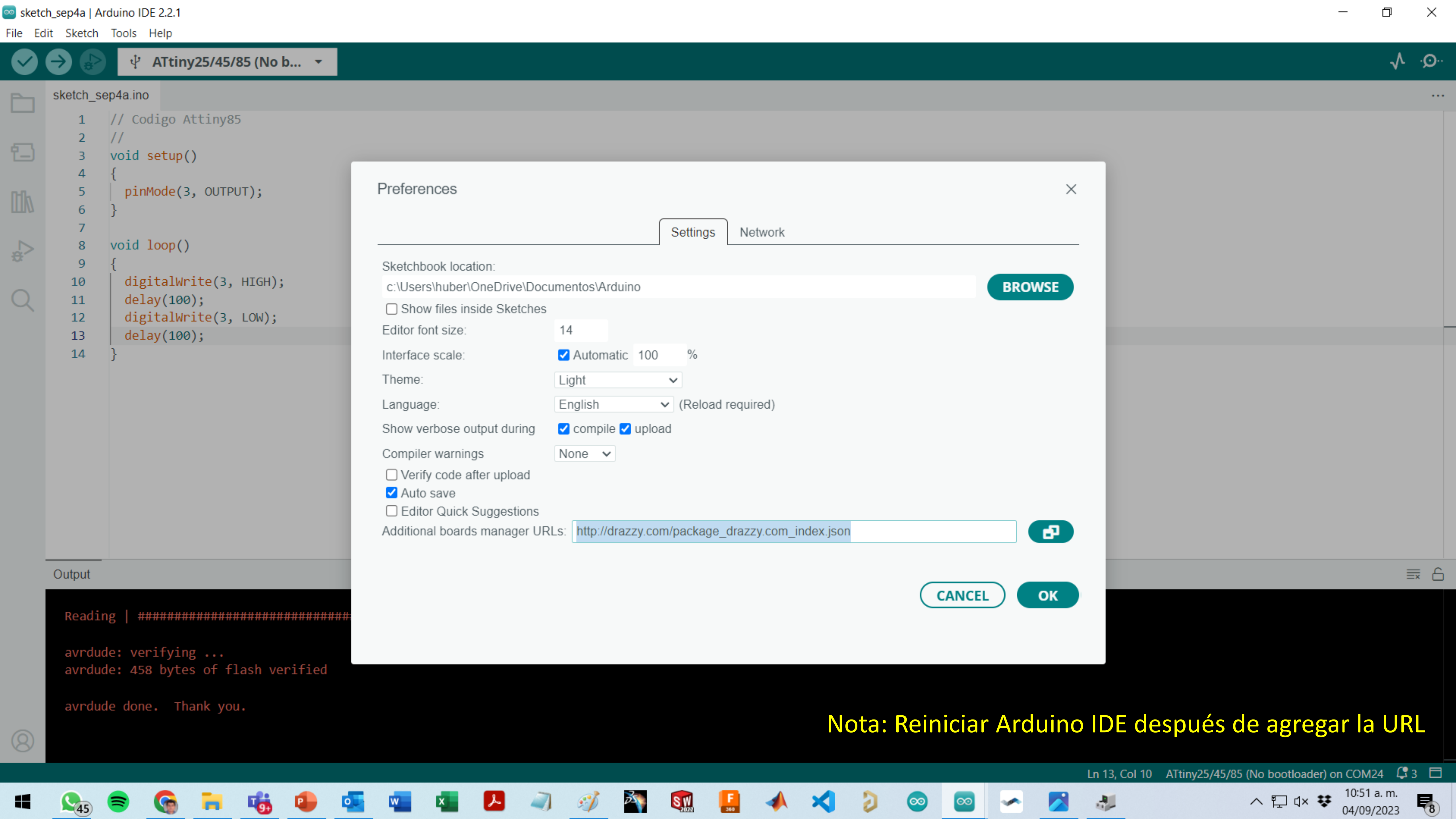
o

Opción 2 https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json



```
ny85
UTPUT);
(3, HIGH);
(3, LOW);
}
```

```
Reading | ################################################## | 100% 0.34s

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```



sketch_sep4a | Arduino IDE 2.2.1

File Edit Sketch Tools Help

ATtiny25/45/85 (No b...

sketch_sep4a.ino

```
// Codigo Attiny85
//
void setup()
{
  pinMode(3, OUTPUT);
}

void loop()
{
  digitalWrite(3, HIGH);
  delay(100);
  digitalWrite(3, LOW);
  delay(100);
}
```

Preferences

Settings | Network

Sketchbook location:
c:\Users\huber\OneDrive\Documentos\Arduino BROWSE

☐ Show files inside Sketches

Editor font size: 14

Interface scale: ☑ Automatic 100 %

Theme: Light

Language: English (Reload required)

Show verbose output during ☑ compile ☑ upload

Compiler warnings: None

☐ Verify code after upload
☑ Auto save
☐ Editor Quick Suggestions

Additional boards manager URLs: http://drazzy.com/package_drazzy.com_index.json

CANCEL OK

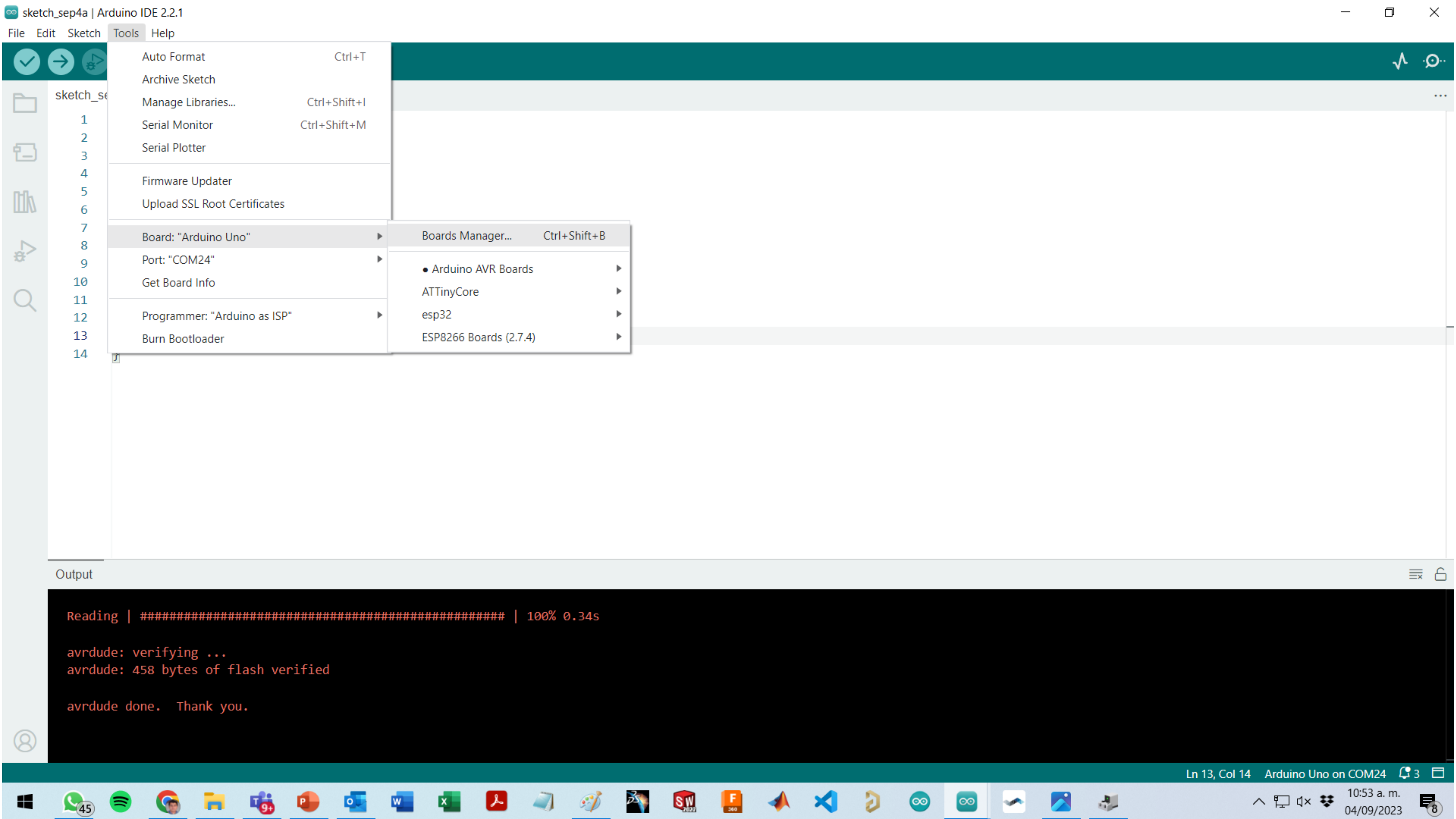
Output

```
Reading | ################################

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

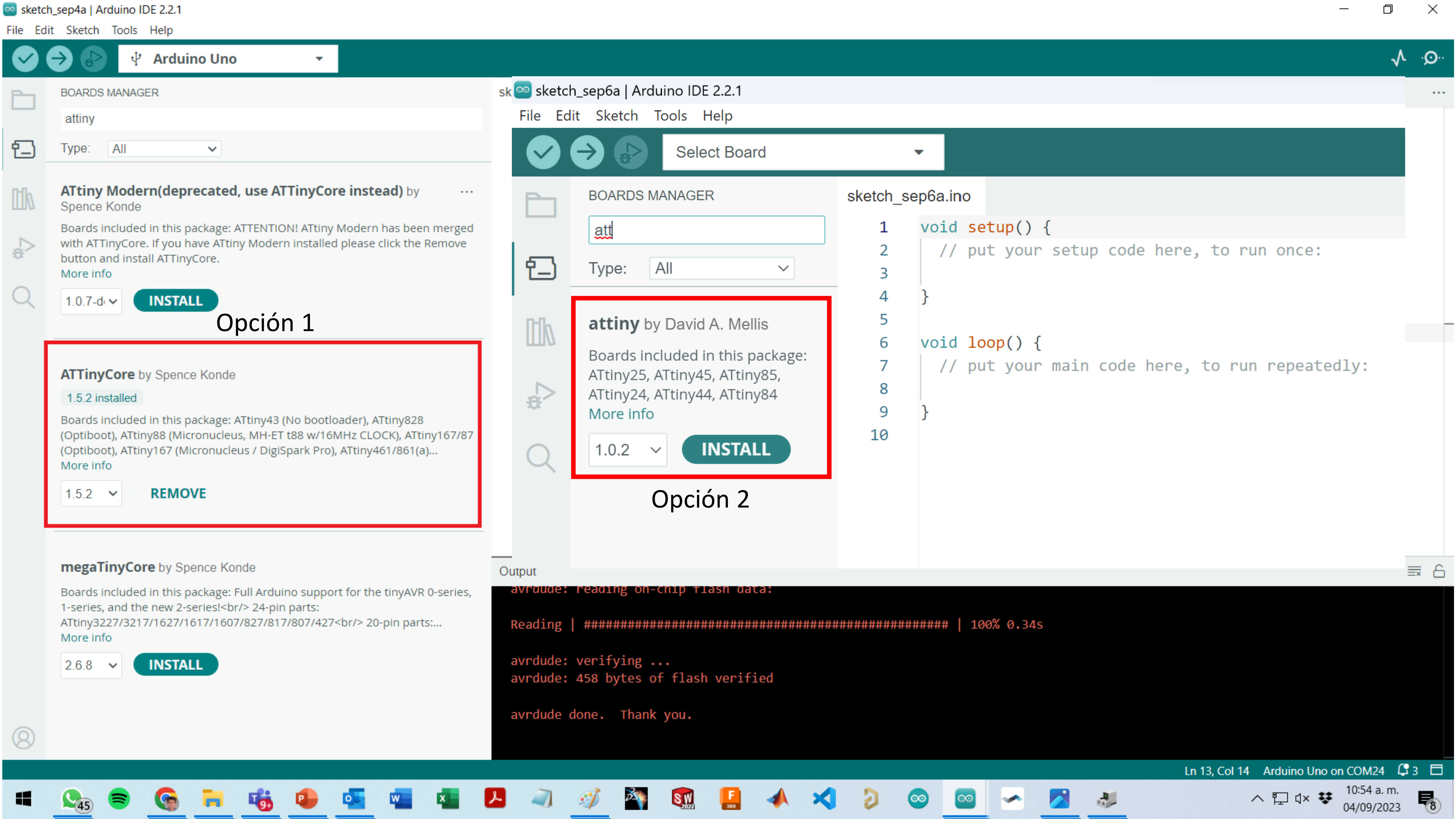
Nota: Reiniciar Arduino IDE después de agregar la URL

Ln 13, Col 10 ATtiny25/45/85 (No bootloader) on COM24

10:51 a. m.
04/09/2023



```
Reading | ################################################## | 100% 0.34s

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```


sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
BOARDS MANAGER
attiny
Type: All
sketch_sep6a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Select Board
BOARDS MANAGER
att
Type: All
sketch_sep6a.ino
Output
Ln 13, Col 14 Arduino Uno on COM24
10:54 a. m.
04/09/2023

ATtiny Modern(deprecated, use ATTinyCore instead) by Spence Konde

Boards included in this package: ATTENTION! ATtiny Modern has been merged with ATTinyCore. If you have ATtiny Modern installed please click the Remove button and install ATTinyCore.
More info

1.0.7-d INSTALL

Opción 1

ATTinyCore by Spence Konde

1.5.2 installed

Boards included in this package: ATtiny43 (No bootloader), ATtiny828 (Optiboot), ATtiny88 (Micronucleus, MH-ET t88 w/16MHz CLOCK), ATtiny167/87 (Optiboot), ATtiny167 (Micronucleus / DigiSpark Pro), ATtiny461/861(a)...
More info

1.5.2 REMOVE

megaTinyCore by Spence Konde

Boards included in this package: Full Arduino support for the tinyAVR 0-series, 1-series, and the new 2-series!<br/> 24-pin parts: ATtiny3227/3217/1627/1617/1607/827/817/807/427<br/> 20-pin parts:...
More info

2.6.8 INSTALL

attiny by David A. Mellis

Boards included in this package: ATtiny25, ATtiny45, ATtiny85, ATtiny24, ATtiny44, ATtiny84
More info

1.0.2 INSTALL

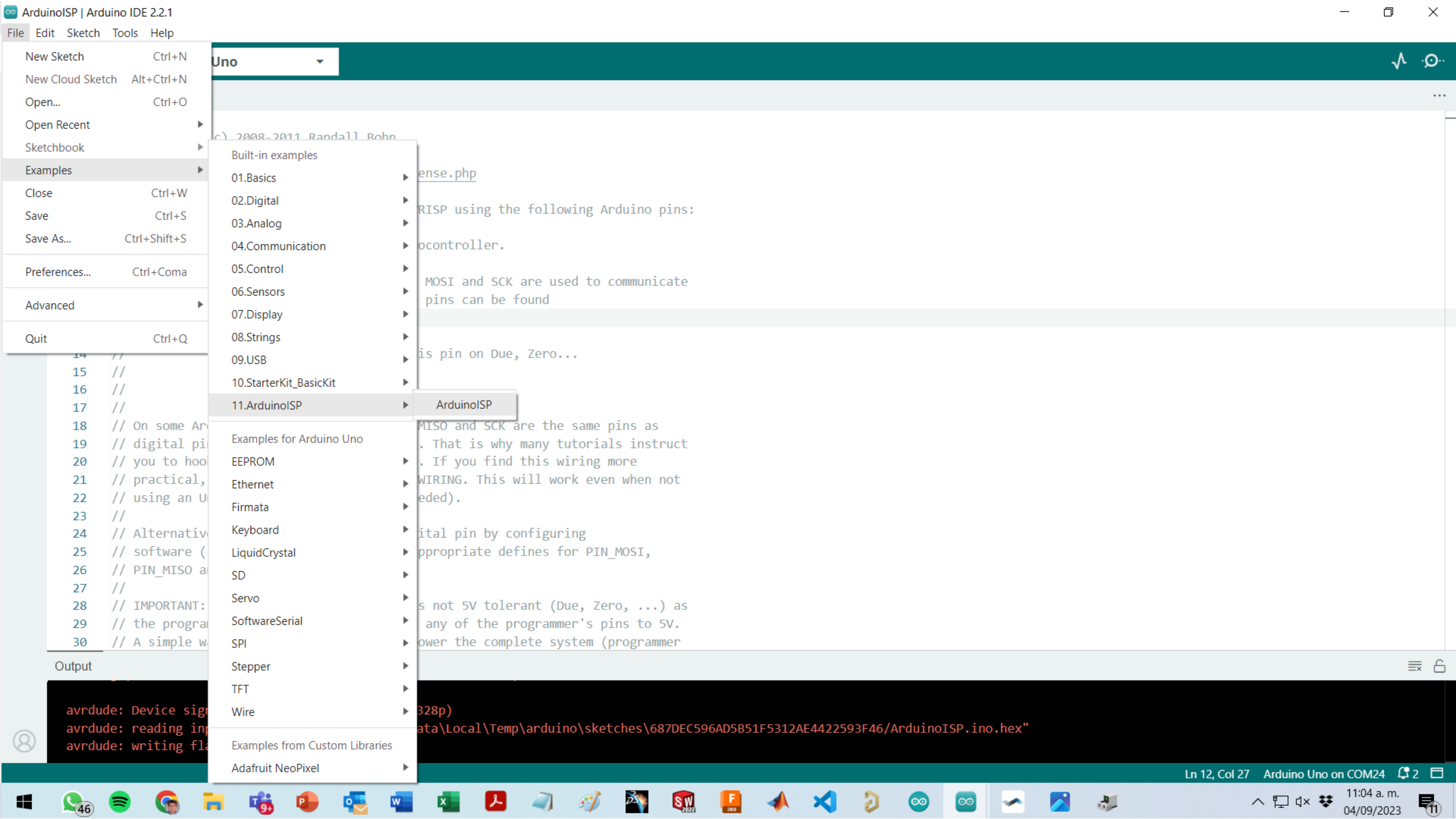
Opción 2

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.34s

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

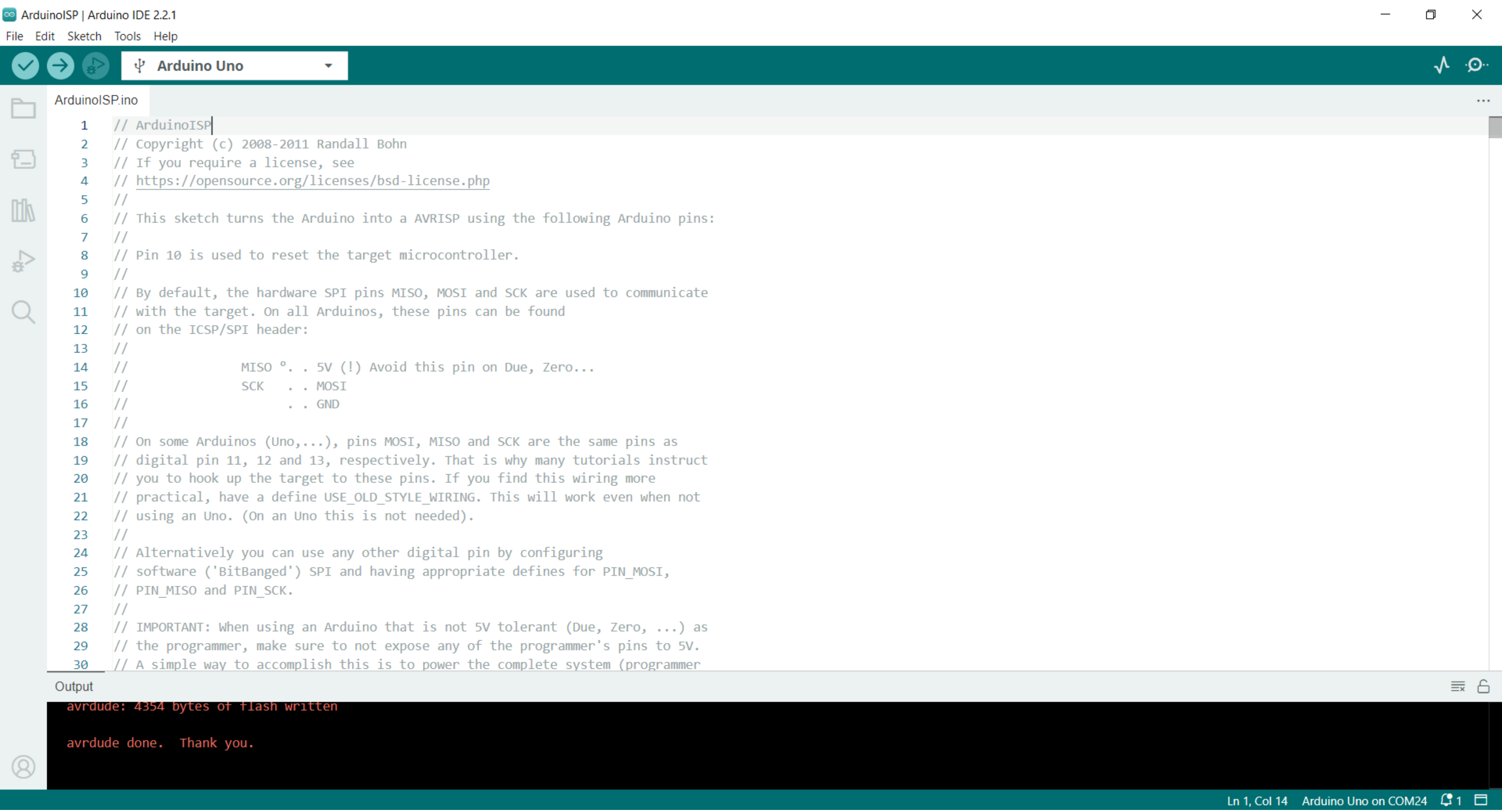
2. Arduino as ISP

UNO
DIGITAL PWM(~)
TX→1
RX←0
POWER
ANALOG IN
RX TX 5V GND
SCL SDA 5V GND
3.3V 3.3V GND GND
12.000
16.000
47 25V RVT
ON
RX
TX
L



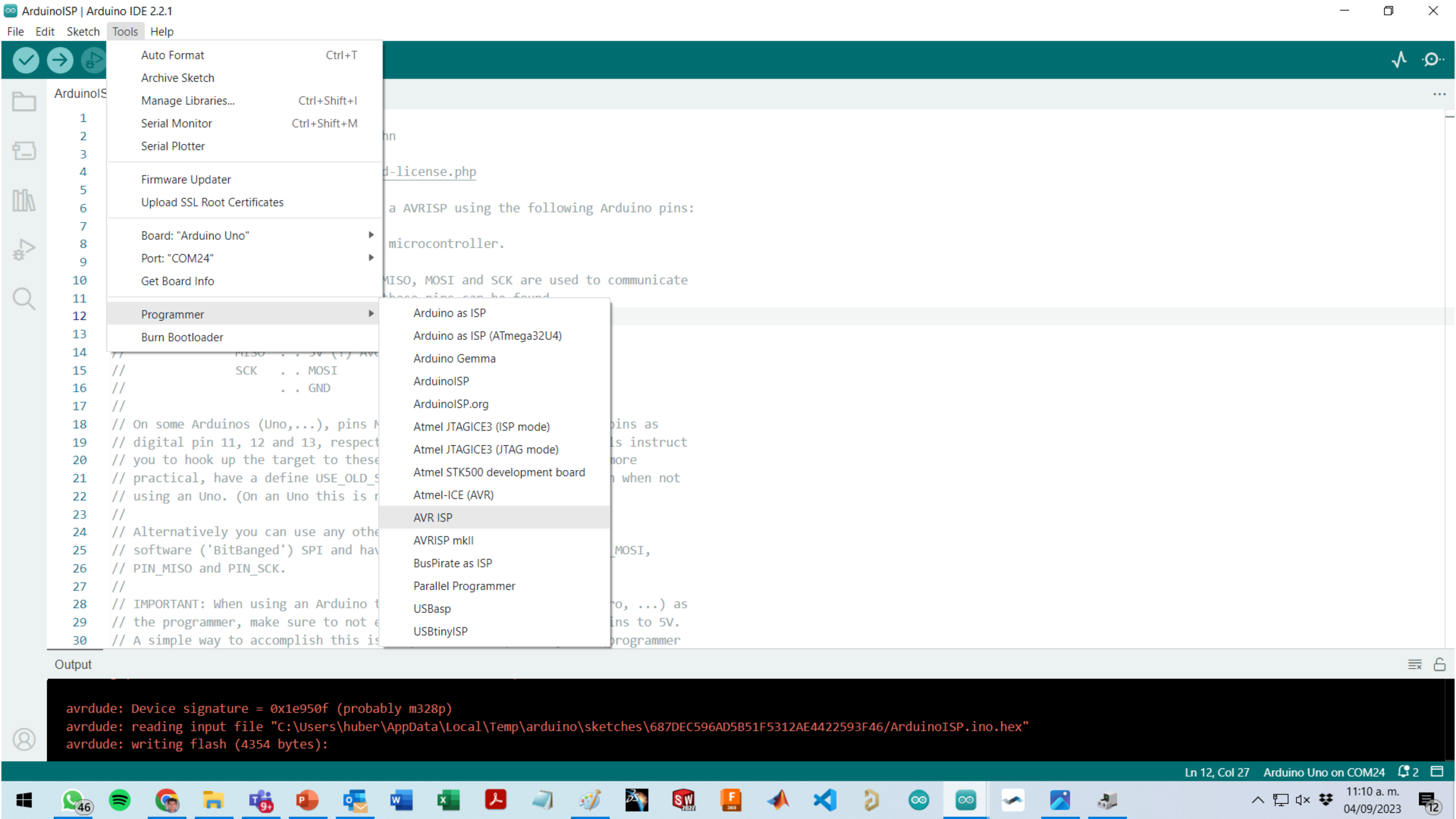
```
c) 2008-2011 Randall Bohn
ense.php
RISP using the following Arduino pins:
ocontroller.
 MOSI and SCK are used to communicate
 pins can be found
is pin on Due, Zero...
//
//
//
// On some Ar              MISO and SCK are the same pins as
// digital pi              . That is why many tutorials instruct
// you to hoo              . If you find this wiring more
// practical,              WIRING. This will work even when not
// using an U              eded).
//
// Alternativ              ital pin by configuring
// software (              ppropriate defines for PIN_MOSI,
// PIN_MISO a
//
// IMPORTANT:              s not 5V tolerant (Due, Zero, ...) as
// the progra               any of the programmer's pins to 5V.
// A simple w              ower the complete system (programmer
```

```
avrdude: Device sig                328p)
avrdude: reading in                ata\Local\Temp\arduino\sketches\687DEC596AD5B51F5312AE4422593F46/ArduinoISP.ino.hex"
avrdude: writing fl
```


ArduinoISP | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
ArduinoISP.ino
Output
Ln 1, Col 14 Arduino Uno on COM24

```
// ArduinoISP
// Copyright (c) 2008-2011 Randall Bohn
// If you require a license, see
// https://opensource.org/licenses/bsd-license.php
//
// This sketch turns the Arduino into a AVRISP using the following Arduino pins:
//
// Pin 10 is used to reset the target microcontroller.
//
// By default, the hardware SPI pins MISO, MOSI and SCK are used to communicate
// with the target. On all Arduinos, these pins can be found
// on the ICSP/SPI header:
//
//               MISO °. . 5V (!) Avoid this pin on Due, Zero...
//               SCK    . . MOSI
//                      . . GND
//
// On some Arduinos (Uno,...), pins MOSI, MISO and SCK are the same pins as
// digital pin 11, 12 and 13, respectively. That is why many tutorials instruct
// you to hook up the target to these pins. If you find this wiring more
// practical, have a define USE_OLD_STYLE_WIRING. This will work even when not
// using an Uno. (On an Uno this is not needed).
//
// Alternatively you can use any other digital pin by configuring
// software ('BitBanged') SPI and having appropriate defines for PIN_MOSI,
// PIN_MISO and PIN_SCK.
//
// IMPORTANT: When using an Arduino that is not 5V tolerant (Due, Zero, ...) as
// the programmer, make sure to not expose any of the programmer's pins to 5V.
// A simple way to accomplish this is to power the complete system (programmer
```

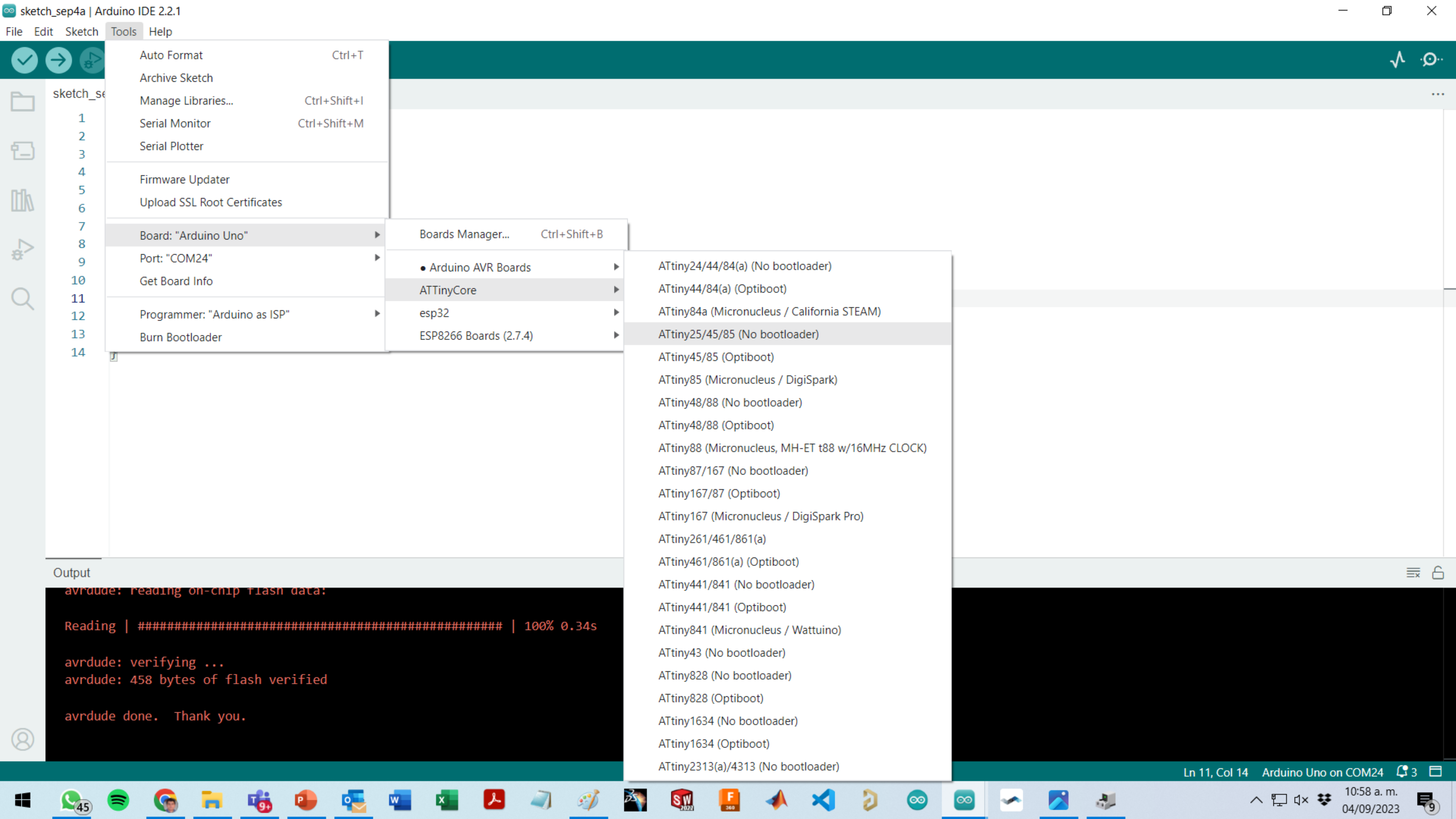
```
avrdude: 4354 bytes of flash written

avrdude done.  Thank you.
```



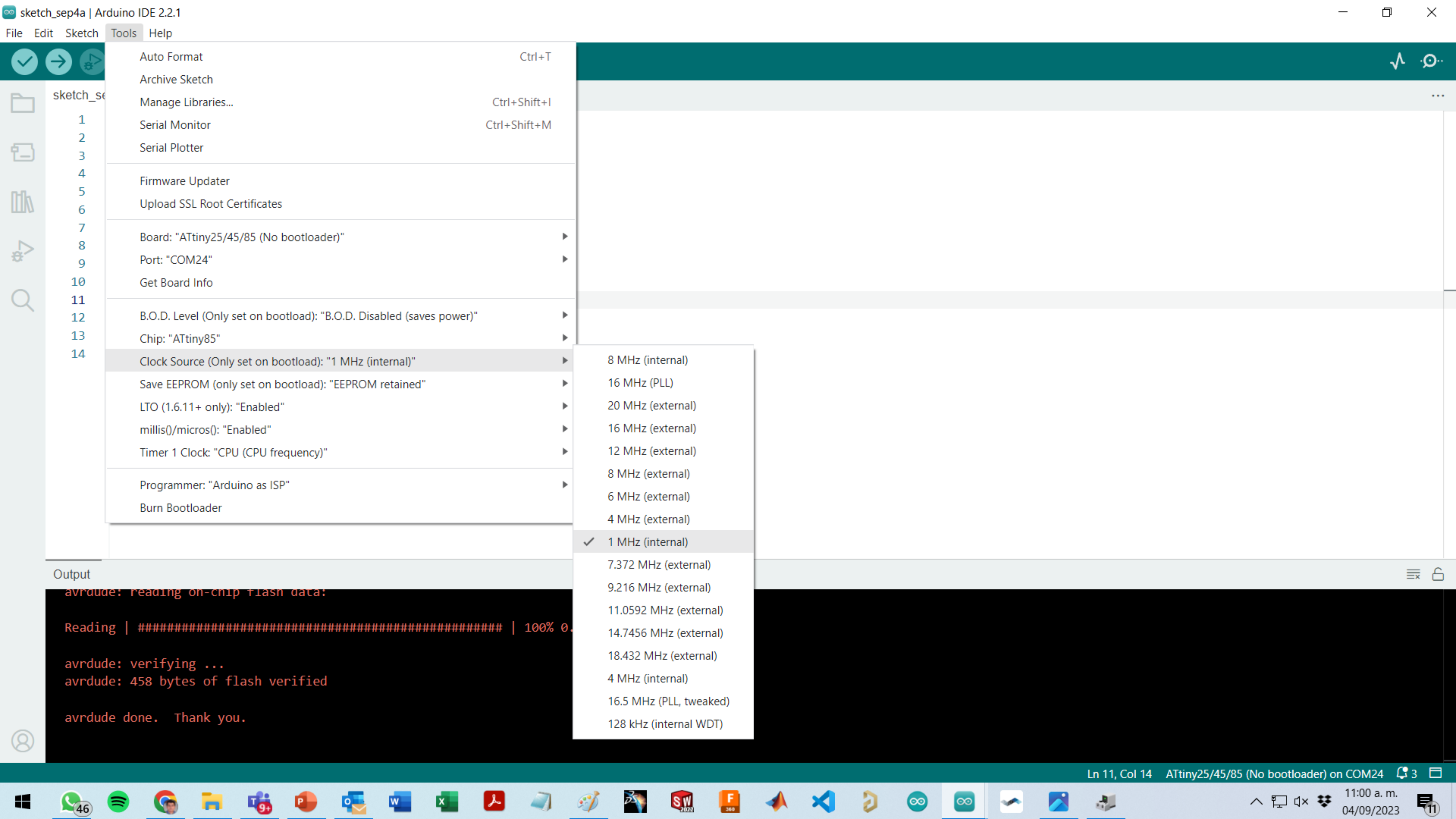
```
hn
d-license.php
 a AVRISP using the following Arduino pins:
 microcontroller.
MISO, MOSI and SCK are used to communicate
//              SCK   . . MOSI
//                    . . GND
//
// On some Arduinos (Uno,...), pins M                    pins as
// digital pin 11, 12 and 13, respect                    ls instruct
// you to hook up the target to these                    more
// practical, have a define USE_OLD_S                    n when not
// using an Uno. (On an Uno this is r
//
// Alternatively you can use any othe
// software ('BitBanged') SPI and hav                    _MOSI,
// PIN_MISO and PIN_SCK.
//
// IMPORTANT: When using an Arduino t                    ro, ...) as
// the programmer, make sure to not e                    ins to 5V.
// A simple way to accomplish this is                    programmer
```

```
avrdude: Device signature = 0x1e950f (probably m328p)
avrdude: reading input file "C:\Users\huber\AppData\Local\Temp\arduino\sketches\687DEC596AD5B51F5312AE4422593F46/ArduinoISP.ino.hex"
avrdude: writing flash (4354 bytes):
```

3. Configurar Tarjeta Attiny


sketch_sep4a | Arduino IDE 2.2.1
File
Edit
Sketch
Tools
Help
Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter
Firmware Updater
Upload SSL Root Certificates
Board: "Arduino Uno"
Port: "COM24"
Get Board Info
Programmer: "Arduino as ISP"
Burn Bootloader
Boards Manager... Ctrl+Shift+B
● Arduino AVR Boards
ATTinyCore
esp32
ESP8266 Boards (2.7.4)
ATtiny24/44/84(a) (No bootloader)
ATtiny44/84(a) (Optiboot)
ATtiny84a (Micronucleus / California STEAM)
ATtiny25/45/85 (No bootloader)
ATtiny45/85 (Optiboot)
ATtiny85 (Micronucleus / DigiSpark)
ATtiny48/88 (No bootloader)
ATtiny48/88 (Optiboot)
ATtiny88 (Micronucleus, MH-ET t88 w/16MHz CLOCK)
ATtiny87/167 (No bootloader)
ATtiny167/87 (Optiboot)
ATtiny167 (Micronucleus / DigiSpark Pro)
ATtiny261/461/861(a)
ATtiny461/861(a) (Optiboot)
ATtiny441/841 (No bootloader)
ATtiny441/841 (Optiboot)
ATtiny841 (Micronucleus / Wattuino)
ATtiny43 (No bootloader)
ATtiny828 (No bootloader)
ATtiny828 (Optiboot)
ATtiny1634 (No bootloader)
ATtiny1634 (Optiboot)
ATtiny2313(a)/4313 (No bootloader)
Output
Ln 11, Col 14
Arduino Uno on COM24
10:58 a. m.
04/09/2023

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.34s

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```



Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter

Firmware Updater
Upload SSL Root Certificates

Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"
Get Board Info

B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"

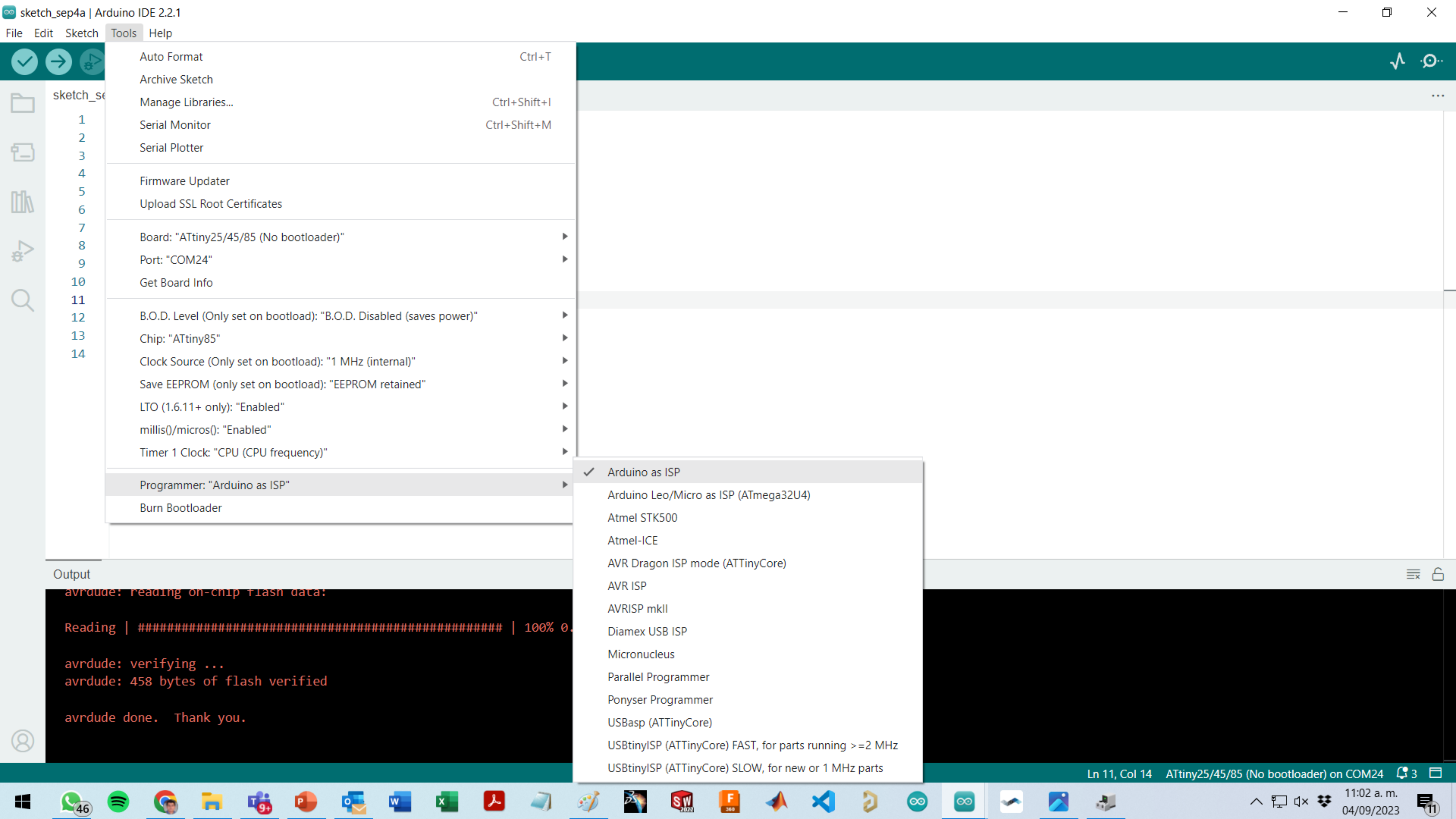
Programmer: "Arduino as ISP"
Burn Bootloader

- 8 MHz (internal)
- 16 MHz (PLL)
- 20 MHz (external)
- 16 MHz (external)
- 12 MHz (external)
- 8 MHz (external)
- 6 MHz (external)
- 4 MHz (external)
- ✓ 1 MHz (internal)
- 7.372 MHz (external)
- 9.216 MHz (external)
- 11.0592 MHz (external)
- 14.7456 MHz (external)
- 18.432 MHz (external)
- 4 MHz (internal)
- 16.5 MHz (PLL, tweaked)
- 128 kHz (internal WDT)

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

Ln 11, Col 14   ATtiny25/45/85 (No bootloader) on COM24   3

11:00 a. m.
04/09/2023



Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"

B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"

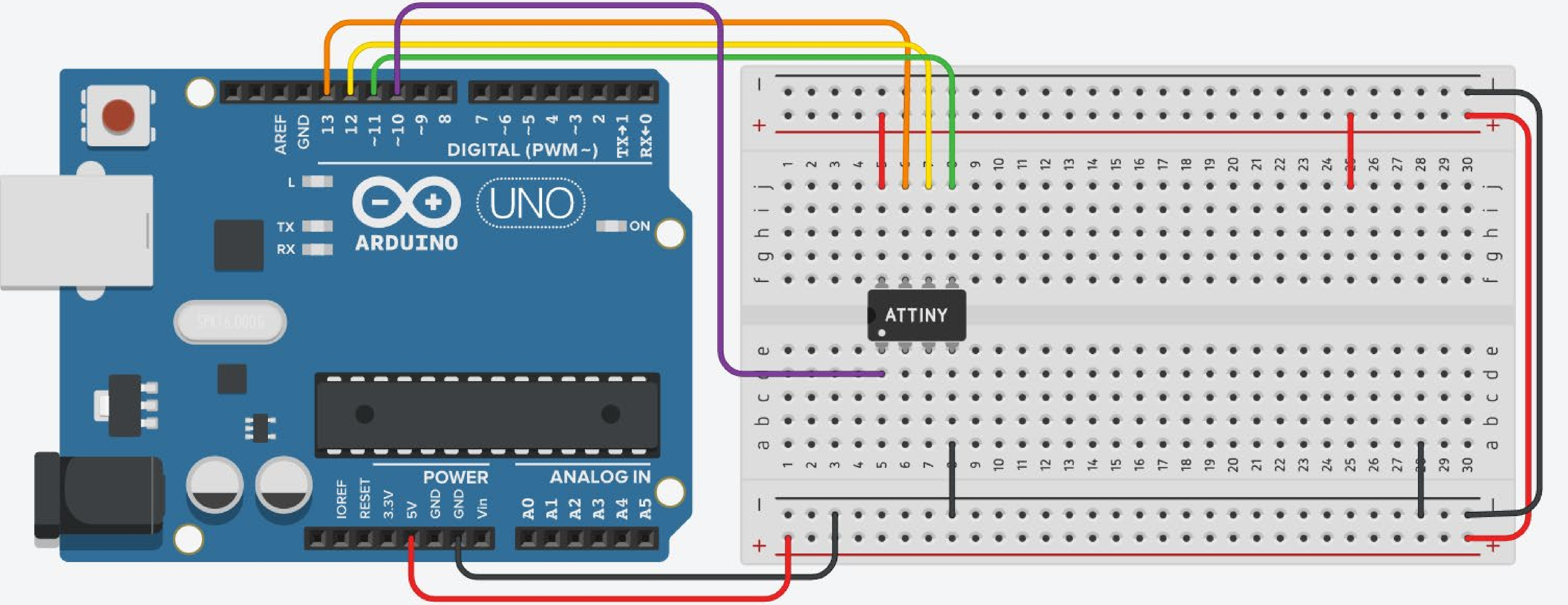
Programmer: "Arduino as ISP"

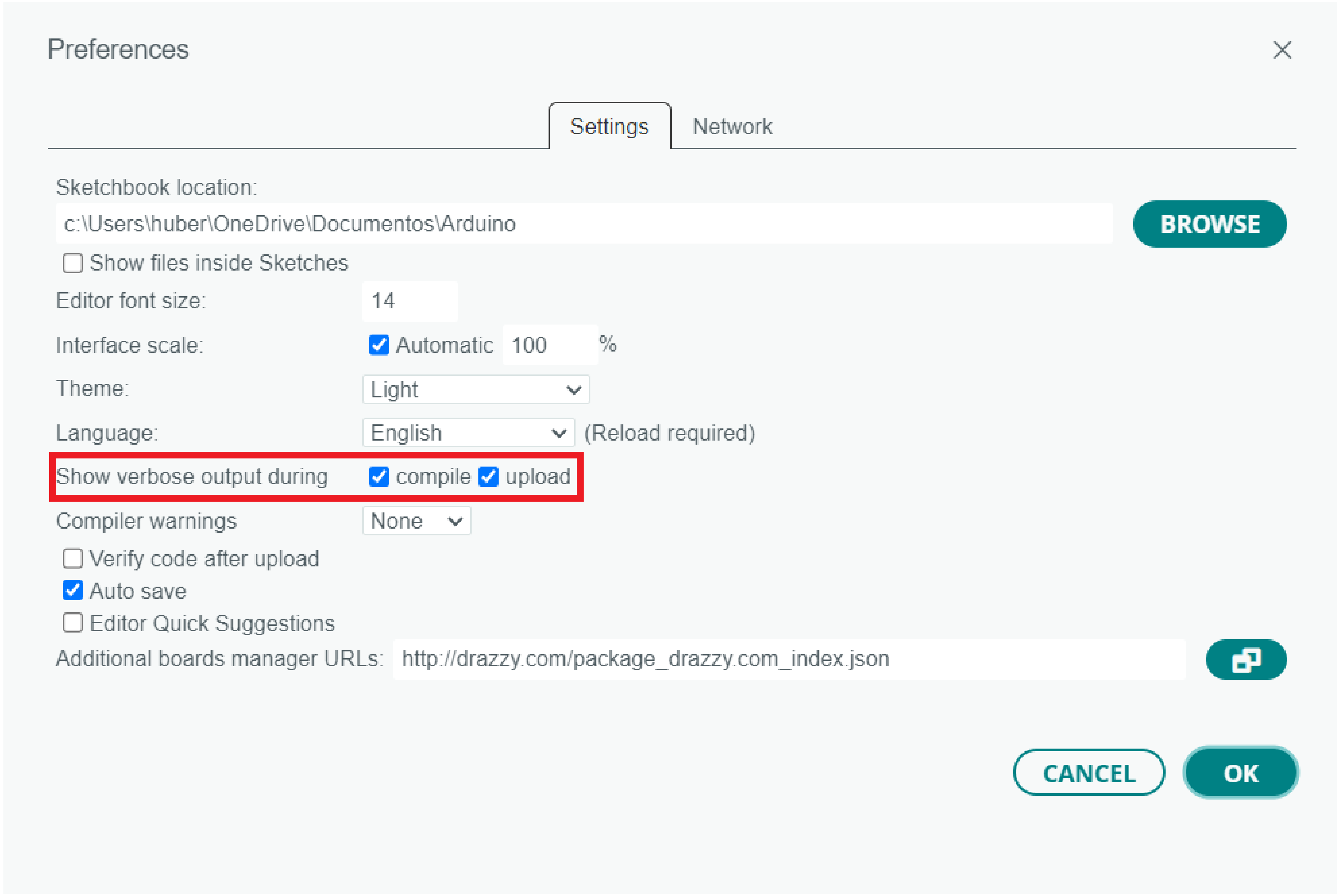
- ✓ Arduino as ISP
- Arduino Leo/Micro as ISP (ATmega32U4)
- Atmel STK500
- Atmel-ICE
- AVR Dragon ISP mode (ATTinyCore)
- AVR ISP
- AVRISP mkII
- Diamex USB ISP
- Micronucleus
- Parallel Programmer
- Ponyser Programmer
- USBasp (ATTinyCore)
- USBtinyISP (ATTinyCore) FAST, for parts running >=2 MHz
- USBtinyISP (ATTinyCore) SLOW, for new or 1 MHz parts

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

Ln 11, Col 14   ATtiny25/45/85 (No bootloader) on COM24

4. Quemar Bootloader

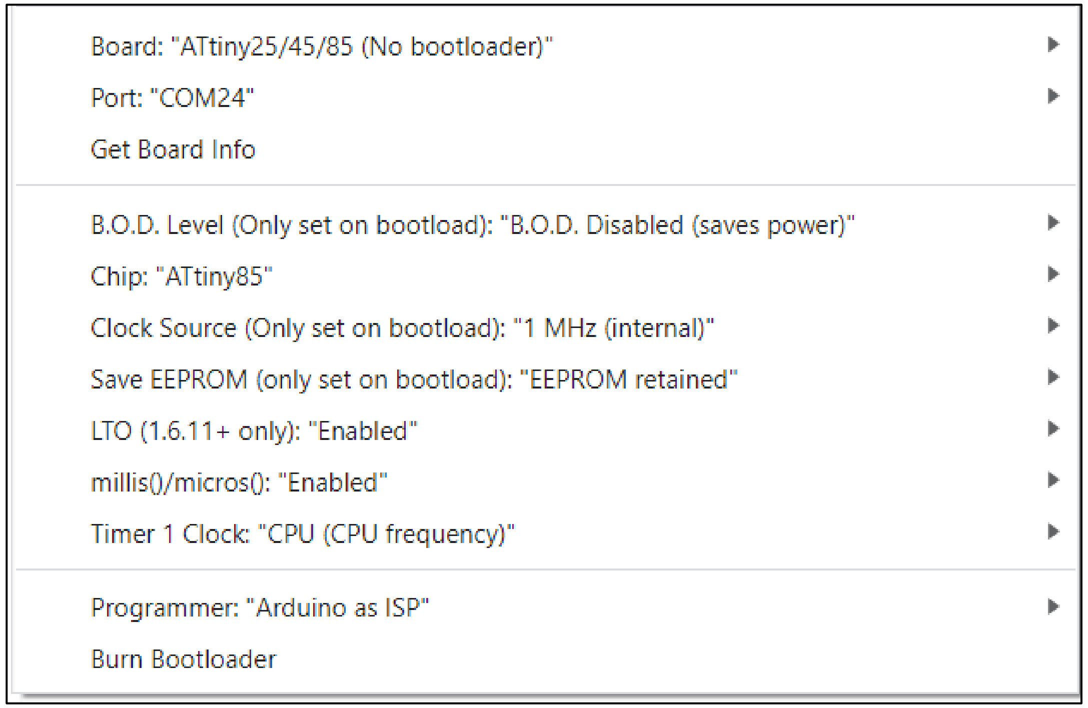

AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
ATTINY



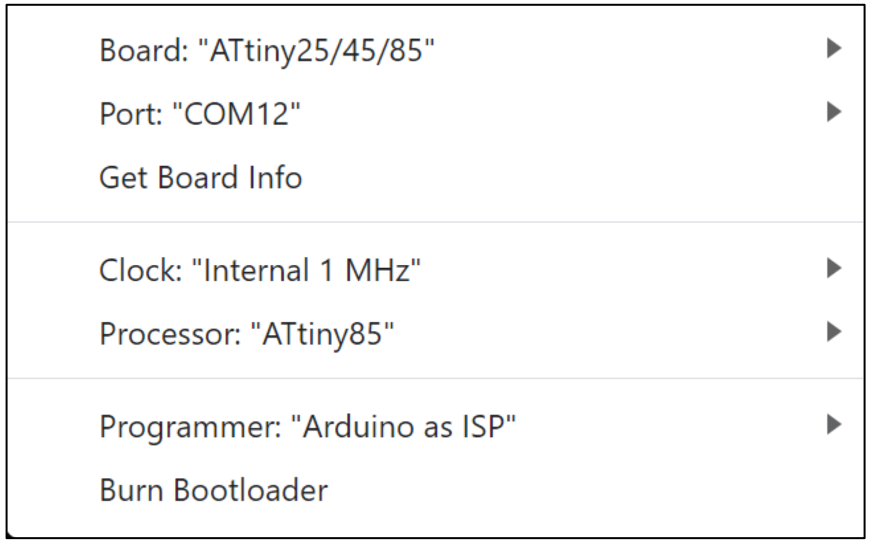
Sketchbook location:
c:\Users\huber\OneDrive\Documentos\Arduino

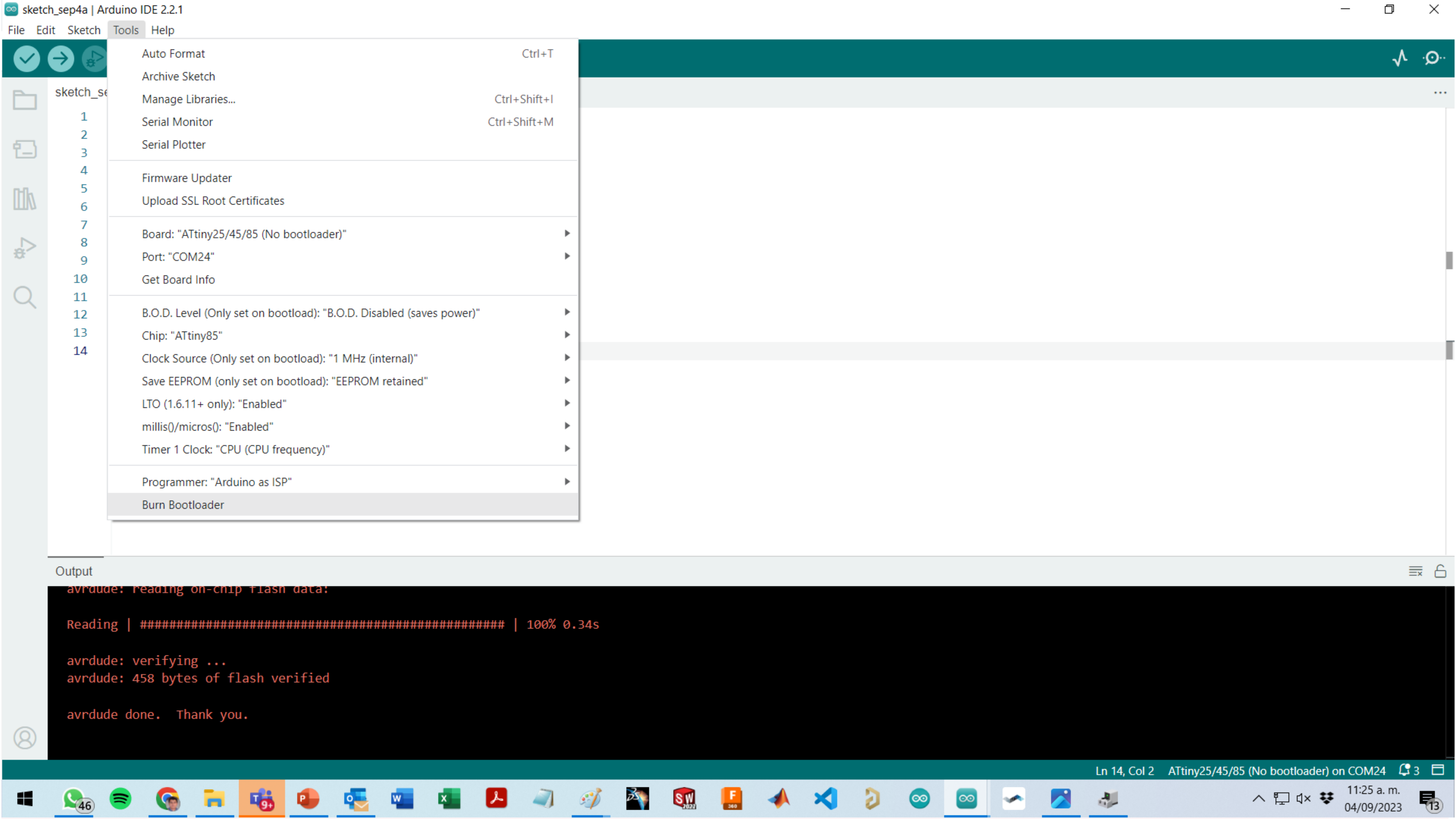
☐ Show files inside Sketches
Editor font size: 14
Interface scale: ☑ Automatic 100 %
Theme: Light
Language: English (Reload required)
Show verbose output during ☑ compile ☑ upload
Compiler warnings None
☐ Verify code after upload
☑ Auto save
☐ Editor Quick Suggestions
Additional boards manager URLs: http://drazzy.com/package_drazzy.com_index.json

Opción 1


Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"
Get Board Info
B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"
Programmer: "Arduino as ISP"
Burn Bootloader

Opción 2


Board: "ATtiny25/45/85"
Port: "COM12"
Get Board Info
Clock: "Internal 1 MHz"
Processor: "ATtiny85"
Programmer: "Arduino as ISP"
Burn Bootloader



Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter

Firmware Updater
Upload SSL Root Certificates

Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"
Get Board Info

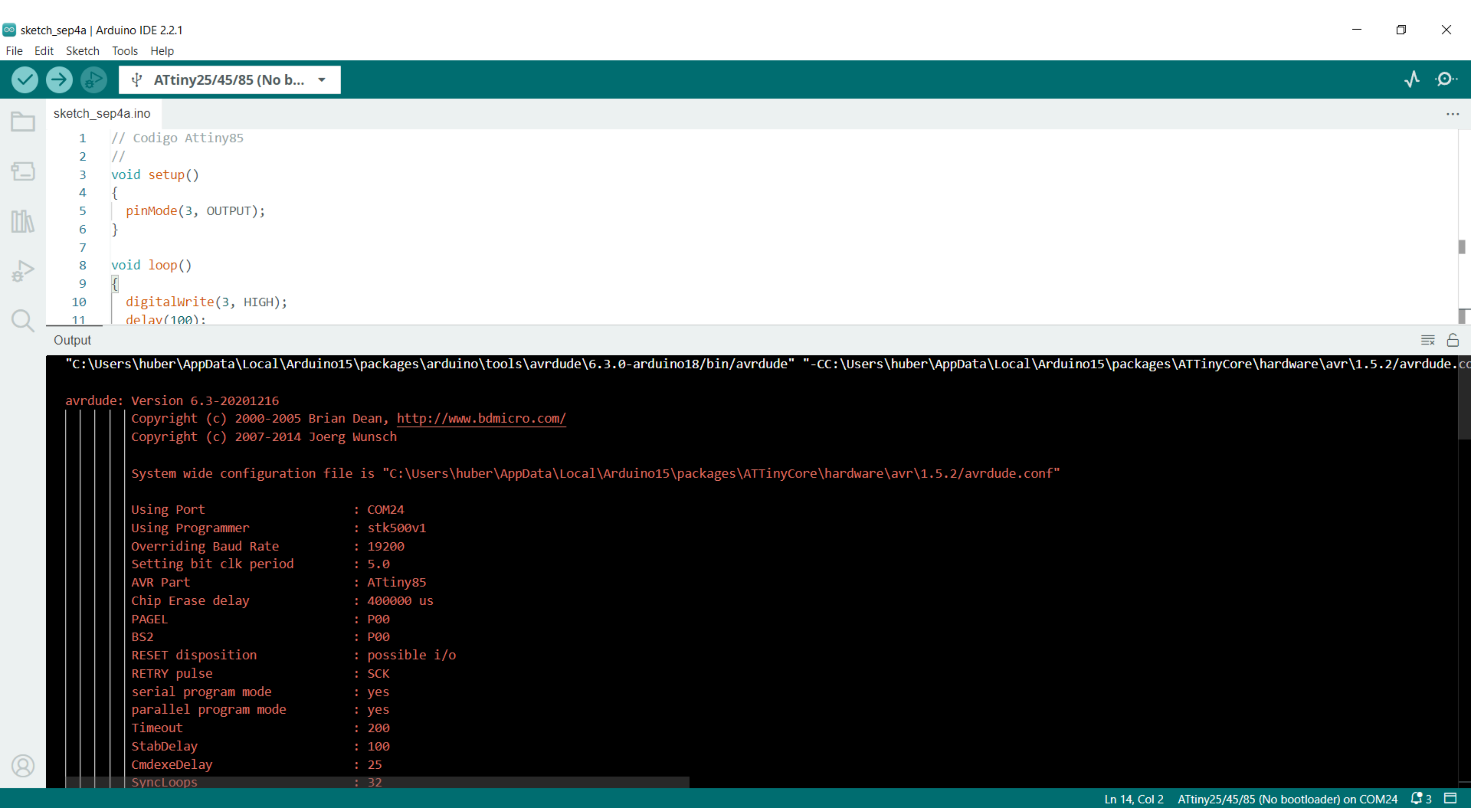
B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"

Programmer: "Arduino as ISP"
Burn Bootloader

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.34s

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

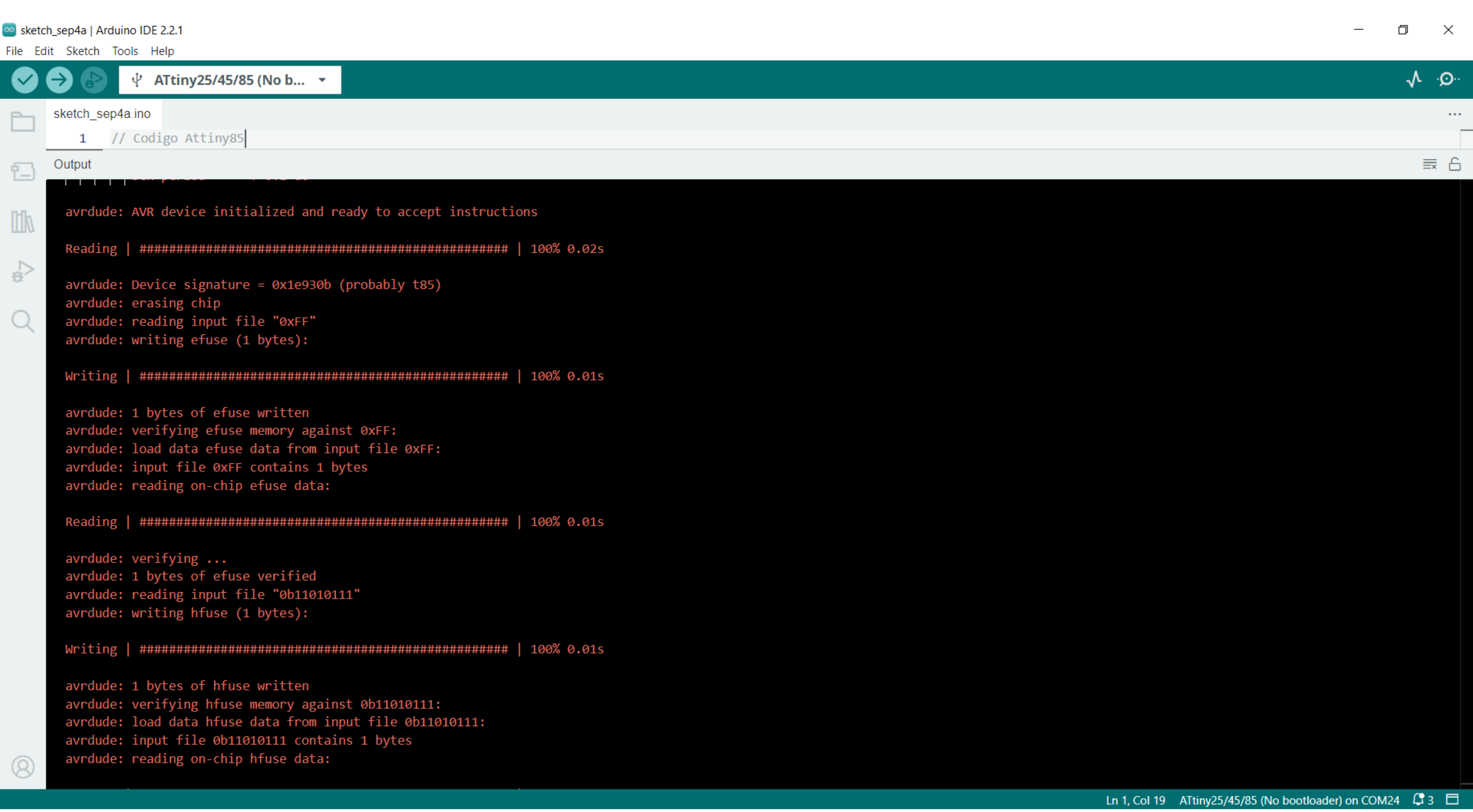
Ln 14, Col 2   ATtiny25/45/85 (No bootloader) on COM24

11:25 a. m.
04/09/2023


sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
ATtiny25/45/85 (No b...
sketch_sep4a.ino
Output
Ln 14, Col 2 ATtiny25/45/85 (No bootloader) on COM24 3

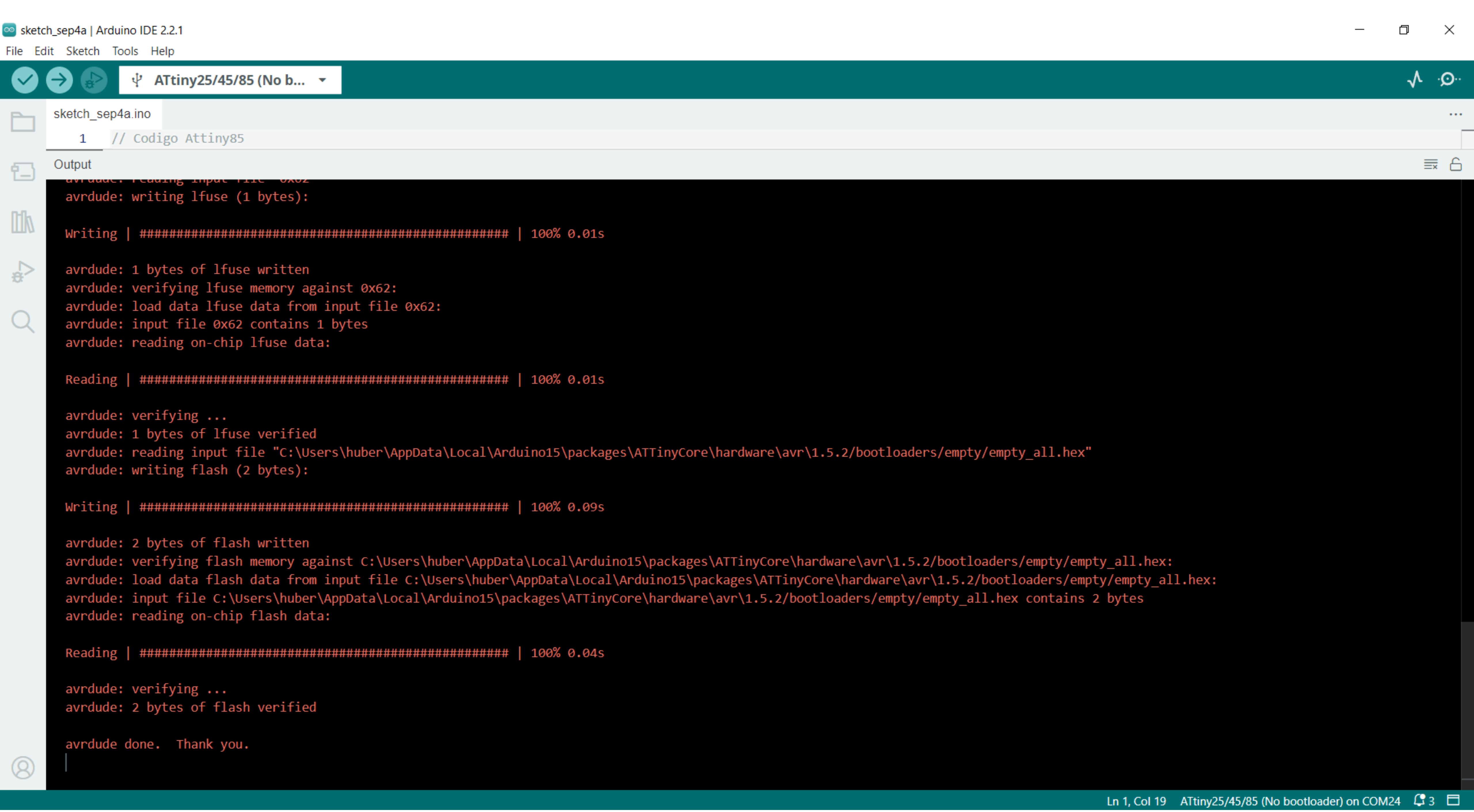
```
// Codigo Attiny85
//
void setup()
{
  pinMode(3, OUTPUT);
}

void loop()
{
  digitalWrite(3, HIGH);
  delay(100);
```

```
"C:\Users\huber\AppData\Local\Arduino15\packages\arduino\tools\avrdude\6.3.0-arduino18/bin/avrdude" "-CC:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/avrdude.c

avrdude: Version 6.3-20201216
         Copyright (c) 2000-2005 Brian Dean, http://www.bdmicro.com/
         Copyright (c) 2007-2014 Joerg Wunsch

         System wide configuration file is "C:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/avrdude.conf"

         Using Port                    : COM24
         Using Programmer              : stk500v1
         Overriding Baud Rate          : 19200
         Setting bit clk period        : 5.0
         AVR Part                      : ATtiny85
         Chip Erase delay              : 400000 us
         PAGEL                         : P00
         BS2                           : P00
         RESET disposition             : possible i/o
         RETRY pulse                   : SCK
         serial program mode           : yes
         parallel program mode         : yes
         Timeout                       : 200
         StabDelay                     : 100
         CmdexeDelay                   : 25
         SyncLoops                     : 32
```


sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
ATtiny25/45/85 (No b...
sketch_sep4a.ino
Output
Ln 1, Col 19 ATtiny25/45/85 (No bootloader) on COM24 3

```
// Codigo Attiny85
```

```
avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.02s

avrdude: Device signature = 0x1e930b (probably t85)
avrdude: erasing chip
avrdude: reading input file "0xFF"
avrdude: writing efuse (1 bytes):

Writing | ################################################## | 100% 0.01s

avrdude: 1 bytes of efuse written
avrdude: verifying efuse memory against 0xFF:
avrdude: load data efuse data from input file 0xFF:
avrdude: input file 0xFF contains 1 bytes
avrdude: reading on-chip efuse data:

Reading | ################################################## | 100% 0.01s

avrdude: verifying ...
avrdude: 1 bytes of efuse verified
avrdude: reading input file "0b11010111"
avrdude: writing hfuse (1 bytes):

Writing | ################################################## | 100% 0.01s

avrdude: 1 bytes of hfuse written
avrdude: verifying hfuse memory against 0b11010111:
avrdude: load data hfuse data from input file 0b11010111:
avrdude: input file 0b11010111 contains 1 bytes
avrdude: reading on-chip hfuse data:
```

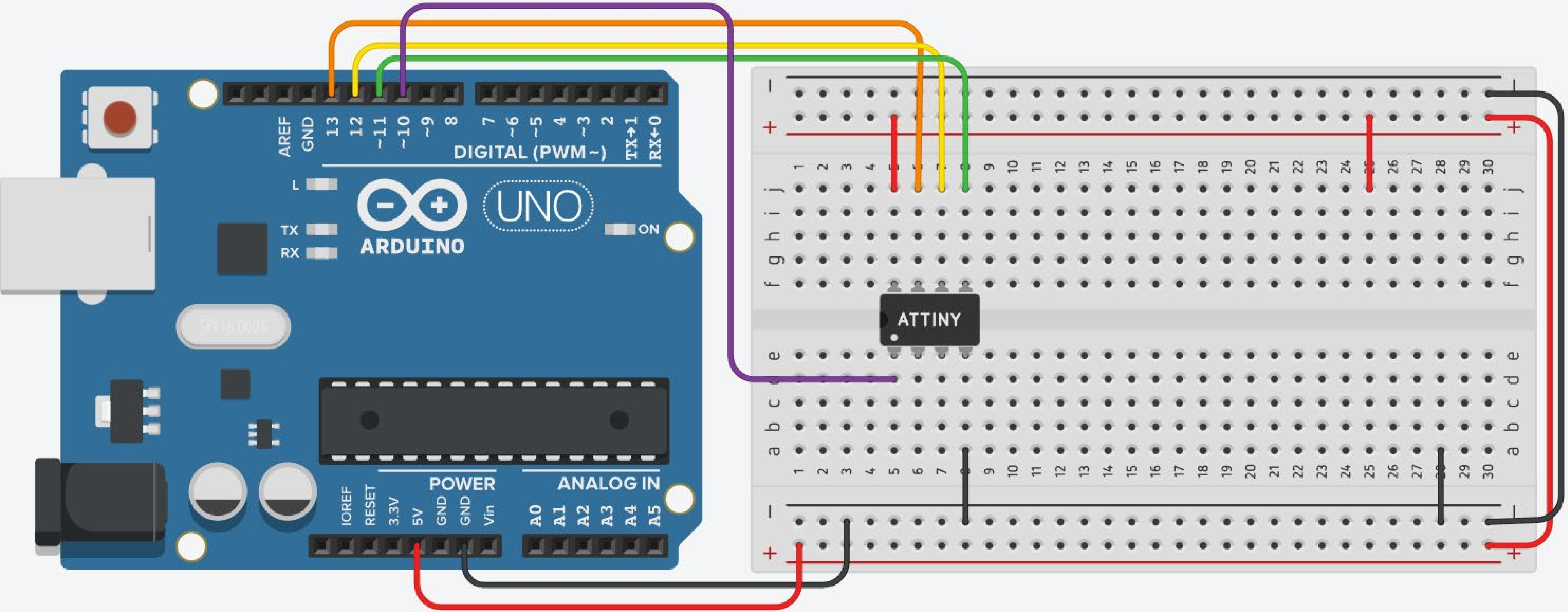

sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
ATtiny25/45/85 (No b...
sketch_sep4a.ino
Output
Ln 1, Col 19 ATtiny25/45/85 (No bootloader) on COM24 3

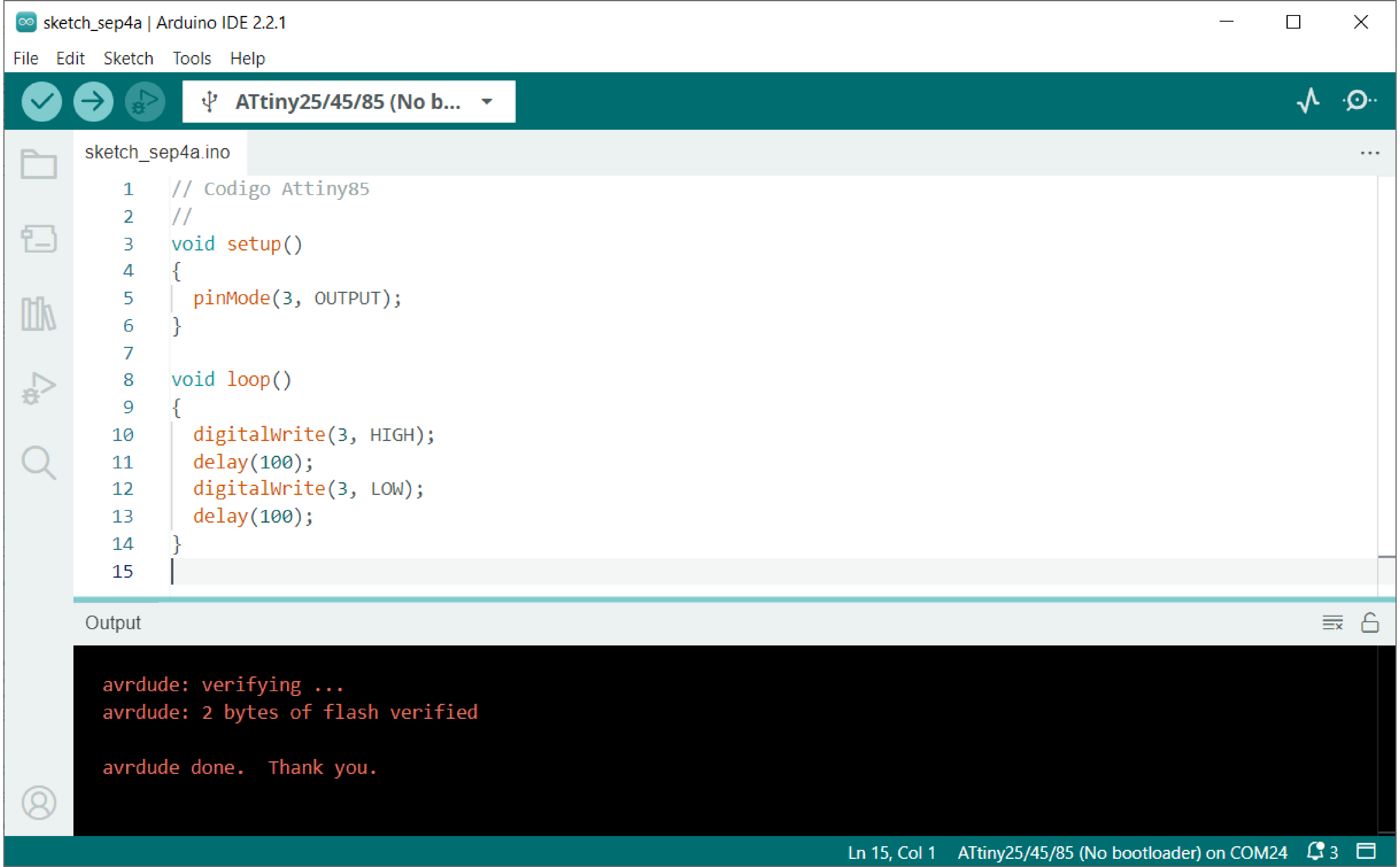
```
// Codigo Attiny85
```

```
avrdude: writing lfuse (1 bytes):

Writing | ################################################## | 100% 0.01s

avrdude: 1 bytes of lfuse written
avrdude: verifying lfuse memory against 0x62:
avrdude: load data lfuse data from input file 0x62:
avrdude: input file 0x62 contains 1 bytes
avrdude: reading on-chip lfuse data:

Reading | ################################################## | 100% 0.01s

avrdude: verifying ...
avrdude: 1 bytes of lfuse verified
avrdude: reading input file "C:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/bootloaders/empty/empty_all.hex"
avrdude: writing flash (2 bytes):

Writing | ################################################## | 100% 0.09s

avrdude: 2 bytes of flash written
avrdude: verifying flash memory against C:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/bootloaders/empty/empty_all.hex:
avrdude: load data flash data from input file C:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/bootloaders/empty/empty_all.hex:
avrdude: input file C:\Users\huber\AppData\Local\Arduino15\packages\ATTinyCore\hardware\avr\1.5.2/bootloaders/empty/empty_all.hex contains 2 bytes
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.04s

avrdude: verifying ...
avrdude: 2 bytes of flash verified

avrdude done.  Thank you.
```

5. Probar Código de Ejemplo

Blink

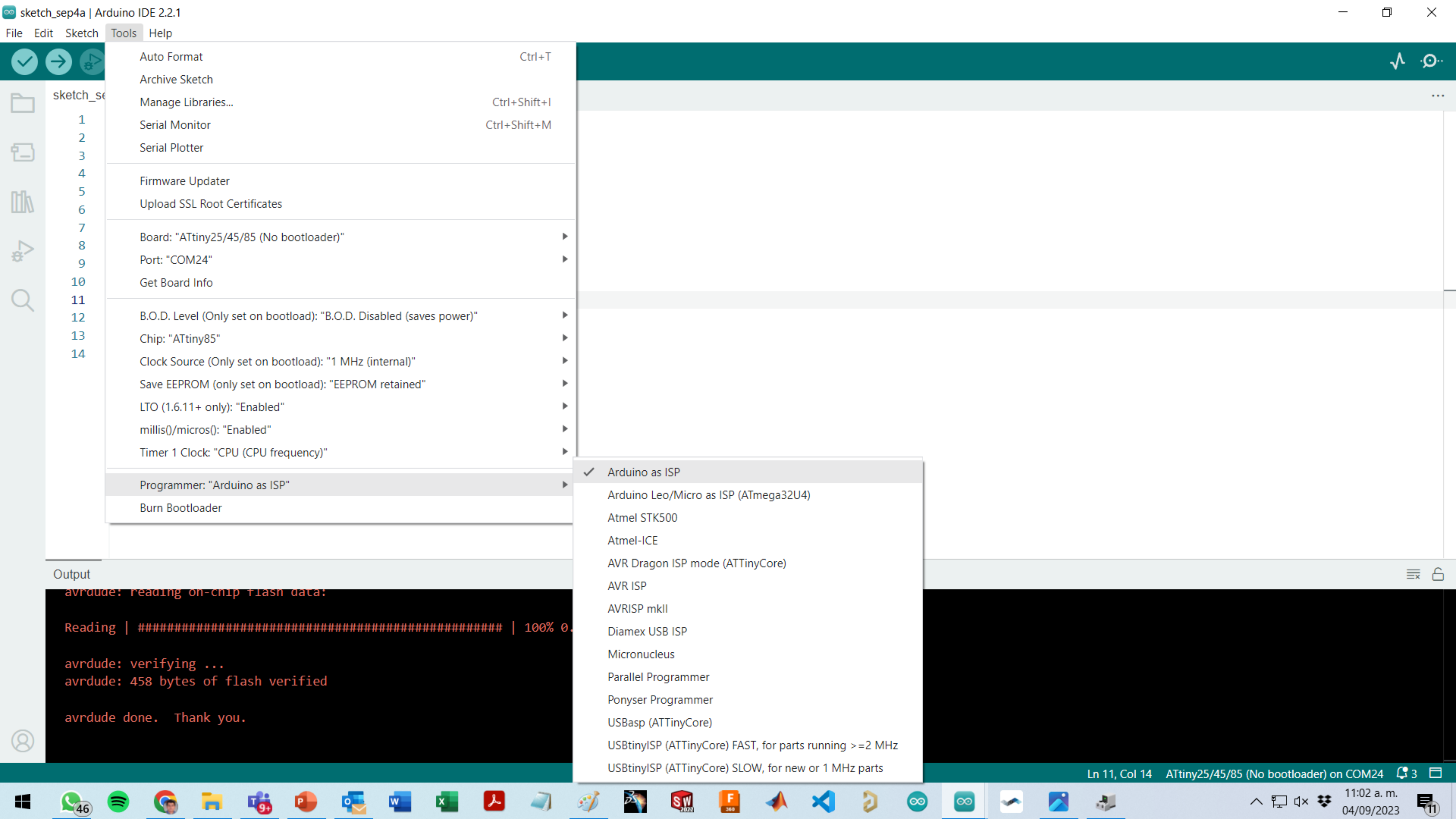

AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5
ATTINY



```
// Codigo Attiny85
//
void setup()
{
  pinMode(3, OUTPUT);
}

void loop()
{
  digitalWrite(3, HIGH);
  delay(100);
  digitalWrite(3, LOW);
  delay(100);
}

```

```
avrdude: verifying ...
avrdude: 2 bytes of flash verified

avrdude done.  Thank you.
```

Ln 15, Col 1   ATtiny25/45/85 (No bootloader) on COM24   3



sketch_sep4a | Arduino IDE 2.2.1

File Edit Sketch Tools Help

```
Auto Format                                                          Ctrl+T
Archive Sketch
Manage Libraries...                                                  Ctrl+Shift+I
Serial Monitor                                                       Ctrl+Shift+M
Serial Plotter

Firmware Updater
Upload SSL Root Certificates

Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"
Get Board Info

B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"

Programmer: "Arduino as ISP"
Burn Bootloader
```

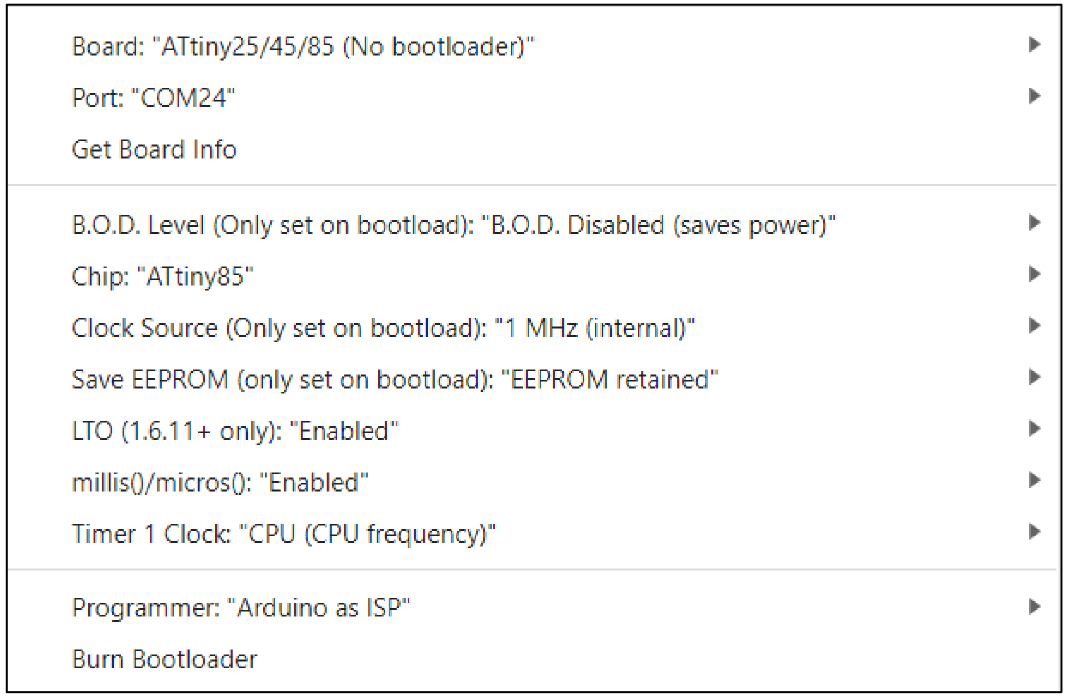
```
✓ Arduino as ISP
Arduino Leo/Micro as ISP (ATmega32U4)
Atmel STK500
Atmel-ICE
AVR Dragon ISP mode (ATTinyCore)
AVR ISP
AVRISP mkII
Diamex USB ISP
Micronucleus
Parallel Programmer
Ponyser Programmer
USBasp (ATTinyCore)
USBtinyISP (ATTinyCore) FAST, for parts running >=2 MHz
USBtinyISP (ATTinyCore) SLOW, for new or 1 MHz parts
```

Output

```
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.

avrdude: verifying ...
avrdude: 458 bytes of flash verified

avrdude done.  Thank you.
```

Ln 11, Col 14   ATtiny25/45/85 (No bootloader) on COM24

Opción 1

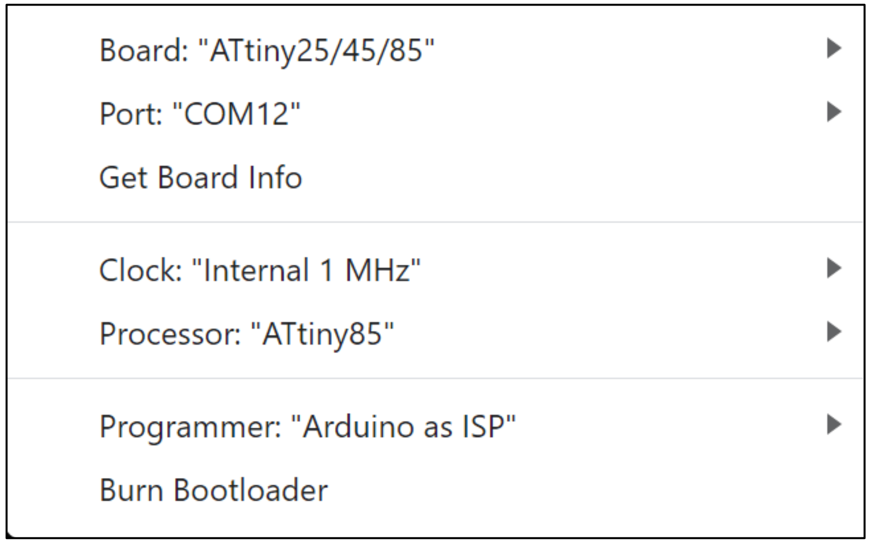


Board: "ATtiny25/45/85 (No bootloader)"
Port: "COM24"
Get Board Info

B.O.D. Level (Only set on bootload): "B.O.D. Disabled (saves power)"
Chip: "ATtiny85"
Clock Source (Only set on bootload): "1 MHz (internal)"
Save EEPROM (only set on bootload): "EEPROM retained"
LTO (1.6.11+ only): "Enabled"
millis()/micros(): "Enabled"
Timer 1 Clock: "CPU (CPU frequency)"

Programmer: "Arduino as ISP"
Burn Bootloader

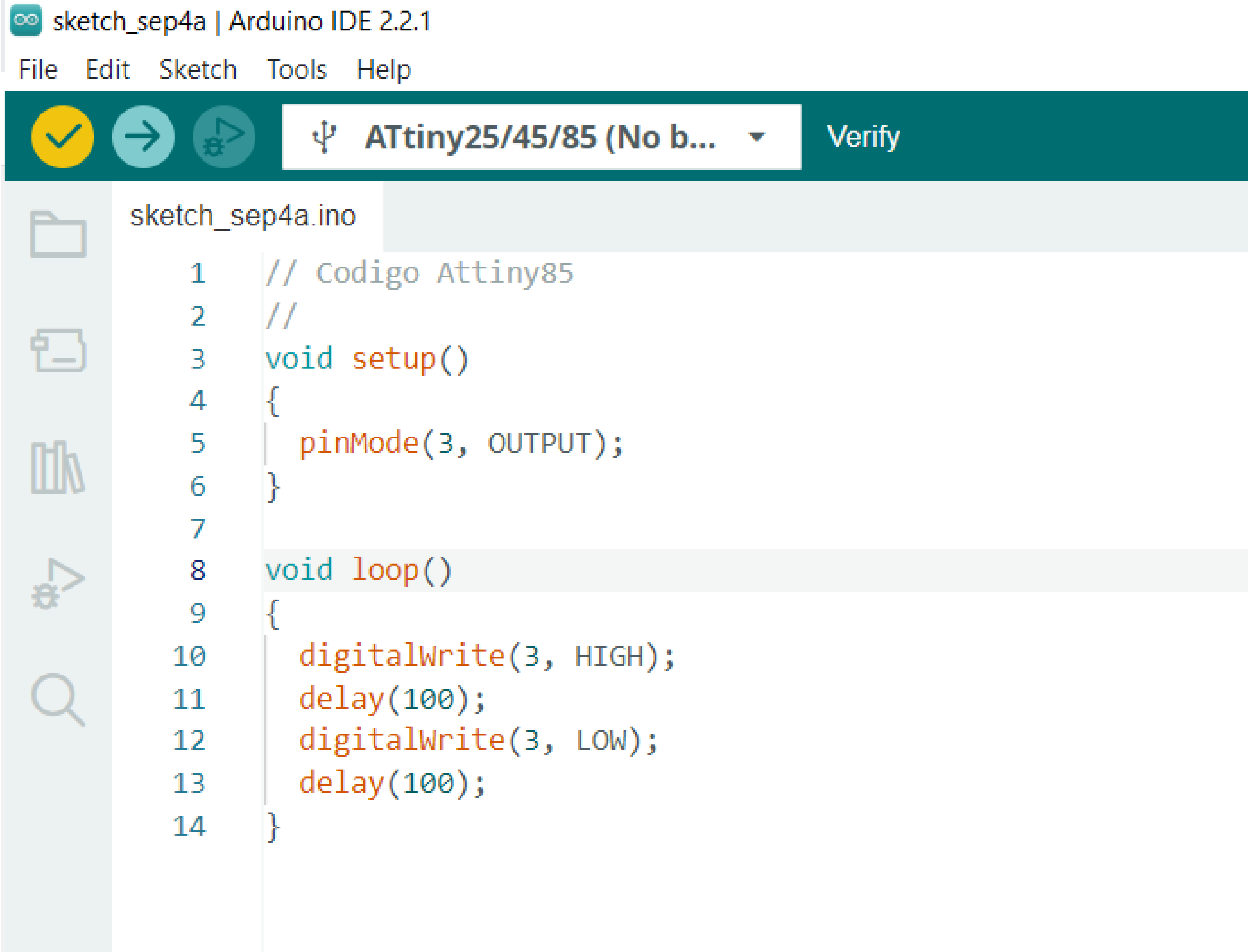
Opción 2



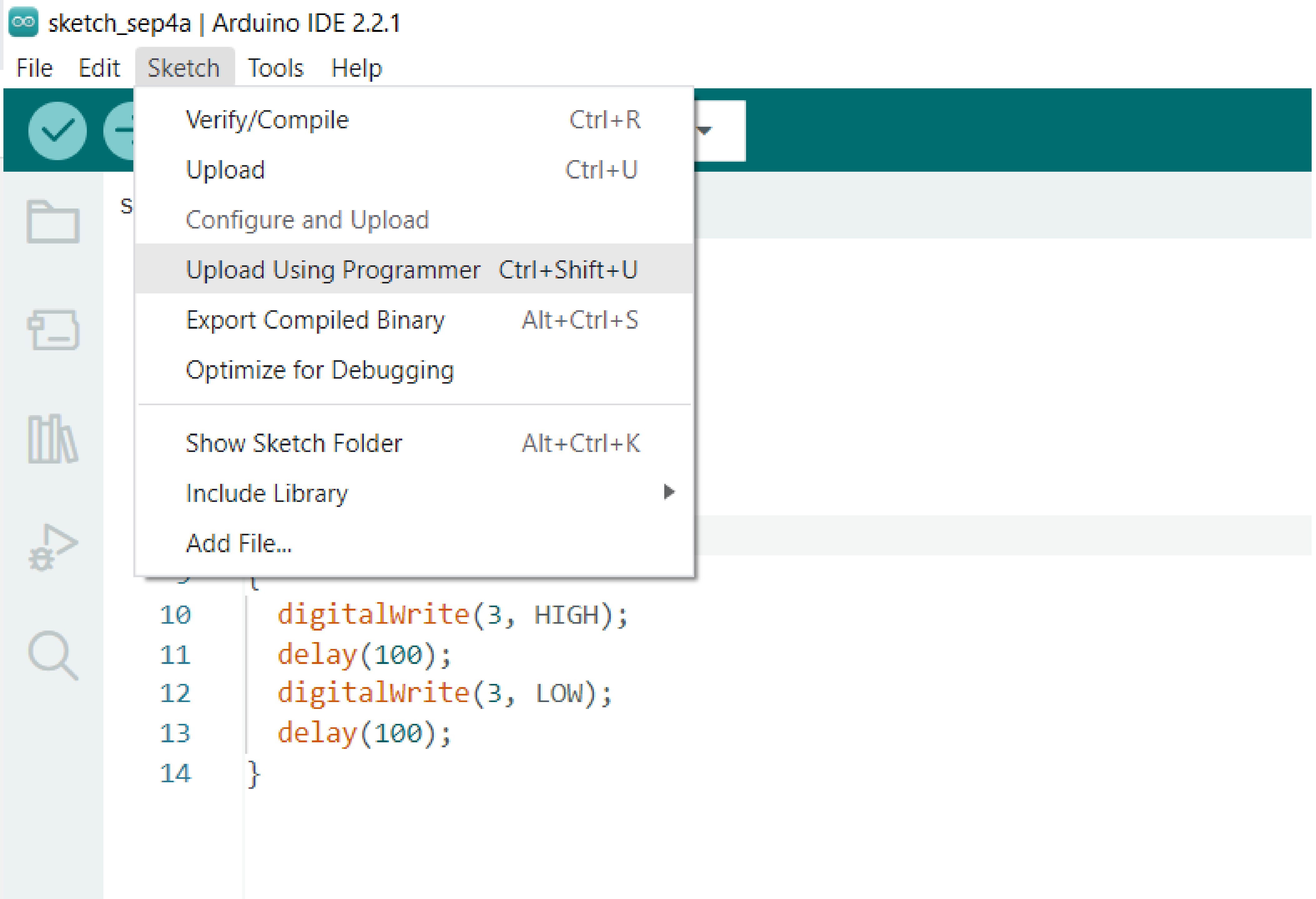
Board: "ATtiny25/45/85"
Port: "COM12"
Get Board Info

Clock: "Internal 1 MHz"
Processor: "ATtiny85"

Programmer: "Arduino as ISP"
Burn Bootloader

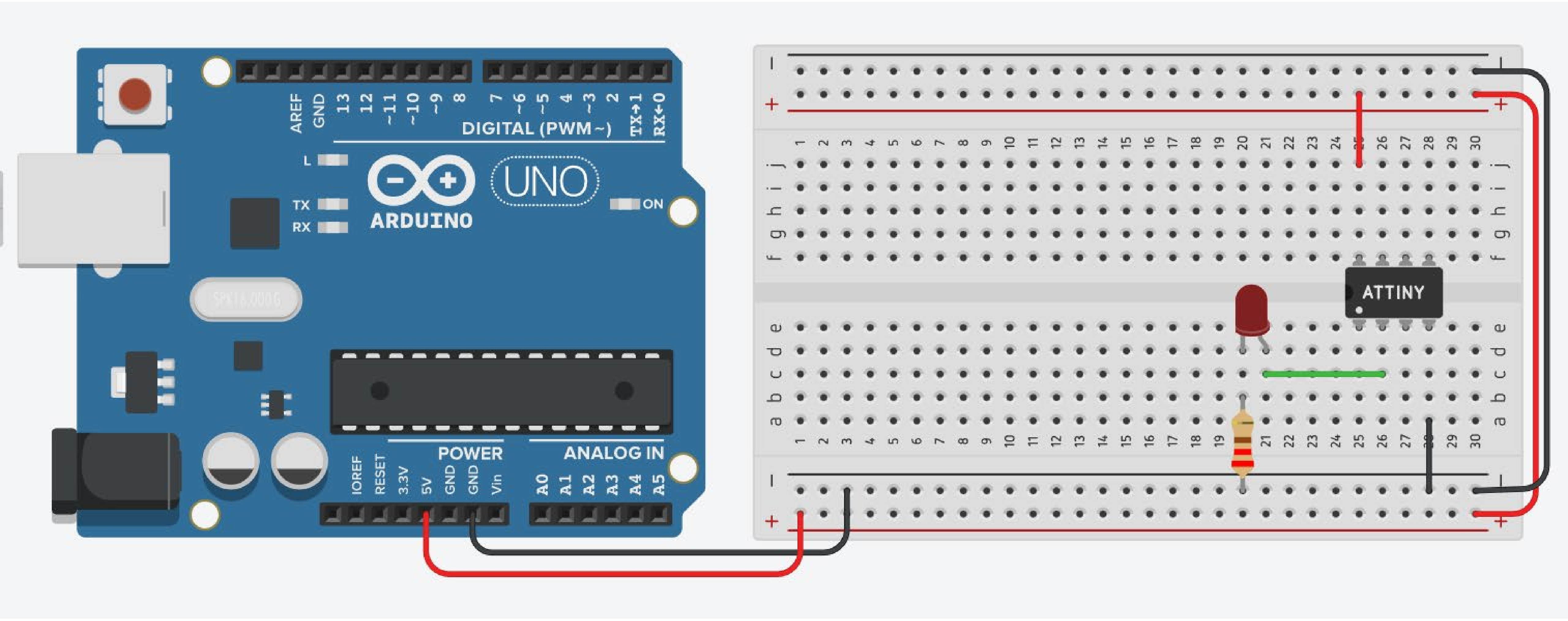


```
// Codigo Attiny85
//
void setup()
{
  pinMode(3, OUTPUT);
}

void loop()
{
  digitalWrite(3, HIGH);
  delay(100);
  digitalWrite(3, LOW);
  delay(100);
}
```



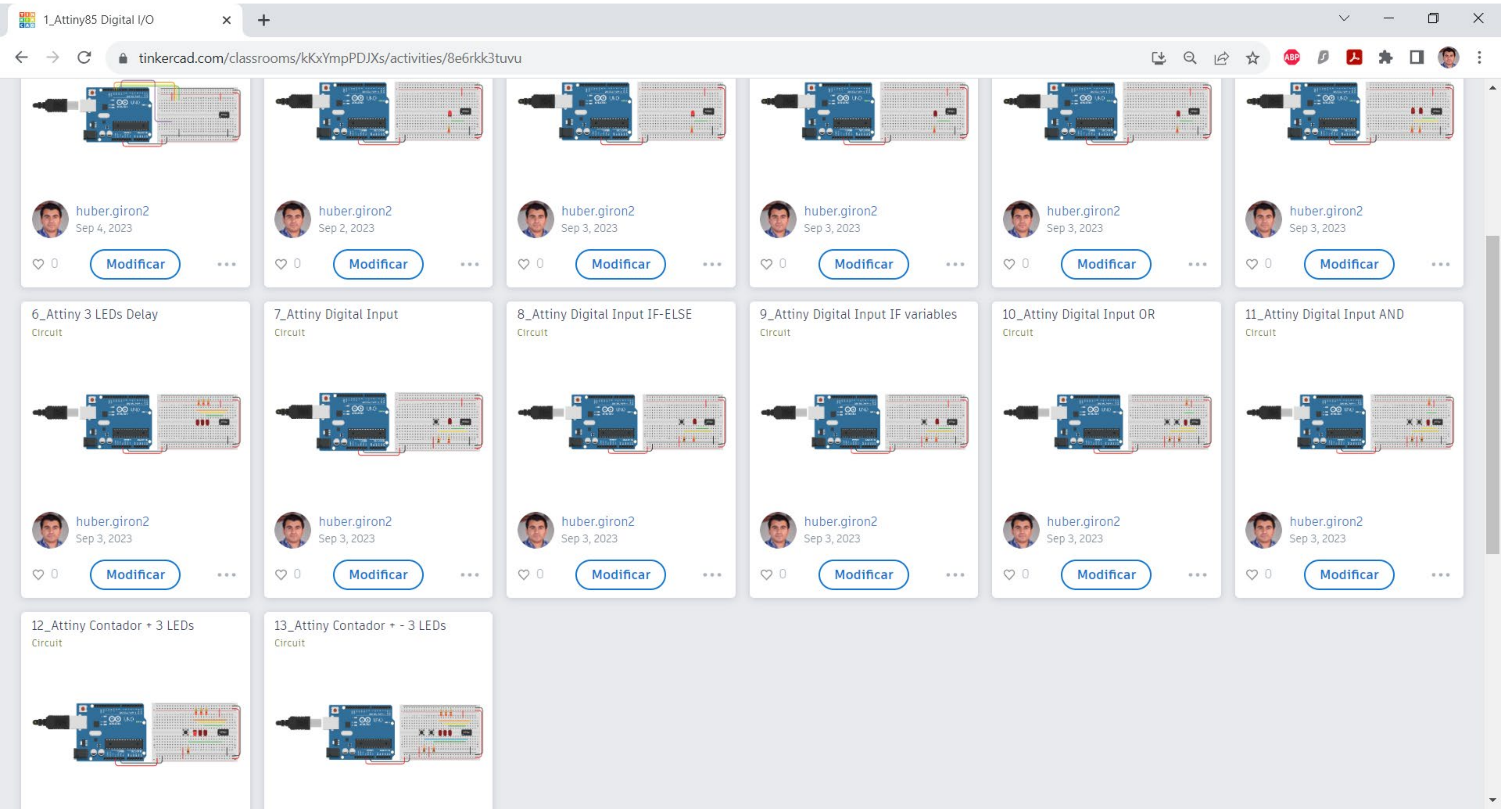
sketch_sep4a | Arduino IDE 2.2.1
File
Edit
Sketch
Tools
Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Configure and Upload
Upload Using Programmer Ctrl+Shift+U
Export Compiled Binary Alt+Ctrl+S
Optimize for Debugging
Show Sketch Folder Alt+Ctrl+K
Include Library
Add File...

```
{
  digitalWrite(3, HIGH);
  delay(100);
  digitalWrite(3, LOW);
  delay(100);
}
```


AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
ATTINY

6. Ejercicios

https://www.tinkercad.com/joinclass/YBTMC3KFN



huber.giron2
Sep 4, 2023
0 Modificar

huber.giron2
Sep 2, 2023
0 Modificar

huber.giron2
Sep 3, 2023
0 Modificar

huber.giron2
Sep 3, 2023
0 Modificar

huber.giron2
Sep 3, 2023
0 Modificar

huber.giron2
Sep 3, 2023
0 Modificar

6_Attiny 3 LEDs Delay
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

7_Attiny Digital Input
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

8_Attiny Digital Input IF-ELSE
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

9_Attiny Digital Input IF variables
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

10_Attiny Digital Input OR
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

11_Attiny Digital Input AND
Circuit
huber.giron2
Sep 3, 2023
0 Modificar

12_Attiny Contador + 3 LEDs
Circuit

13_Attiny Contador + - 3 LEDs
Circuit

Únete a **Curso Attiny85 Básico** con un vínculo o introduce este código de clase:

YBT MC3 KFN

Copiar vínculo   Copiar código

Instrucciones para estudiantes

Vínculo de clase:

1. Accede a la clase con este vínculo: https://www.tinkercad.com/joinclass/YBTMC3KFN
2. Introduce el **alias** asignado por el profesor.

Código de clase:

1. Ve a https://www.tinkercad.com/joinclass
2. Introduce el código de clase: **YBTMC3KFN**
3. Introduce el **alias** asignado por el profesor.