



VIDI

P R O J E C T

X

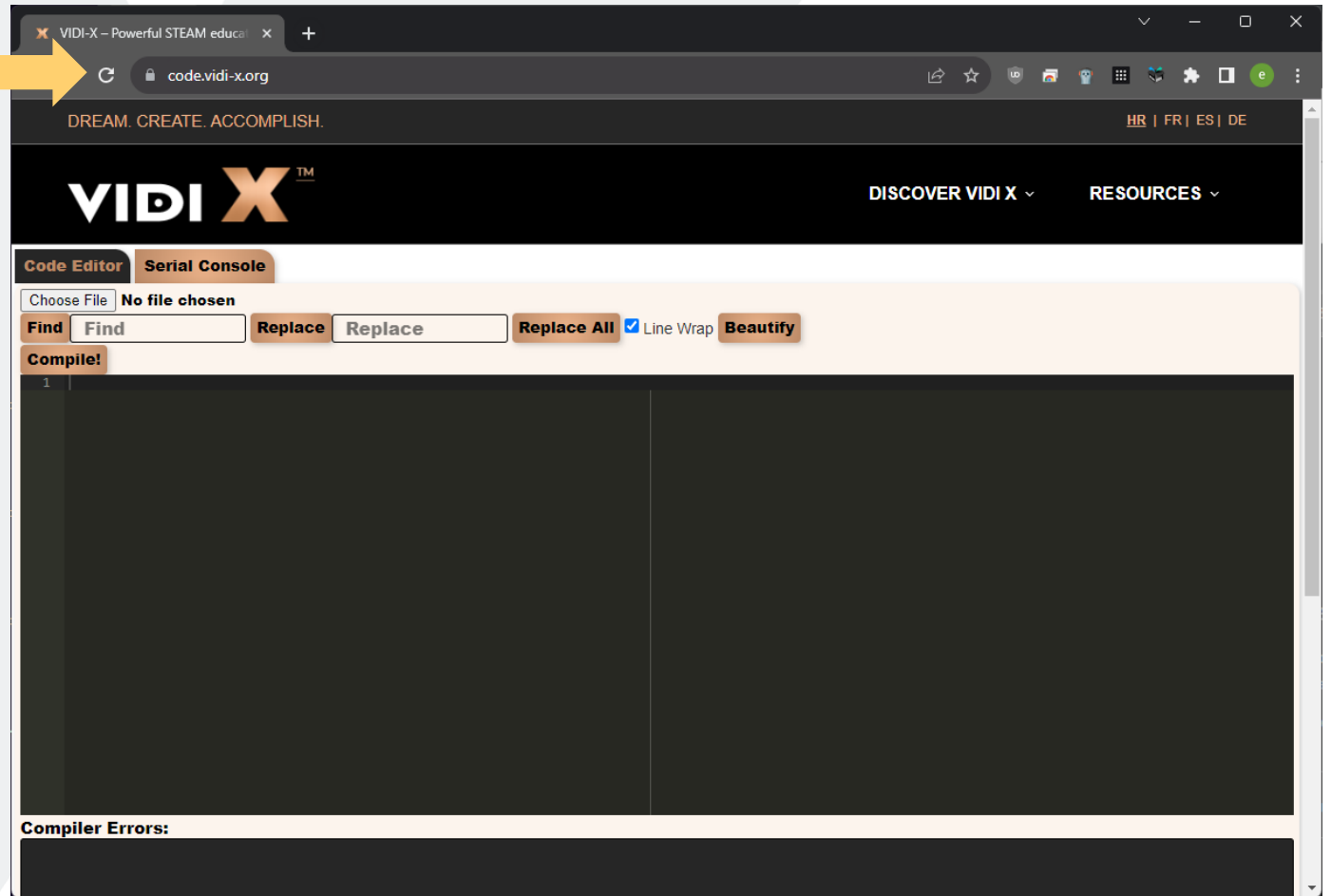


Code Editor Tutorial

1 Step 1

Go to the website
<https://code.vidi-x.org/>

At the same time, you can also plug-in your VIDI X microcomputer so that the computer has enough time to find and install the corresponding USB driver, while we go through the other steps.



2 Step 2

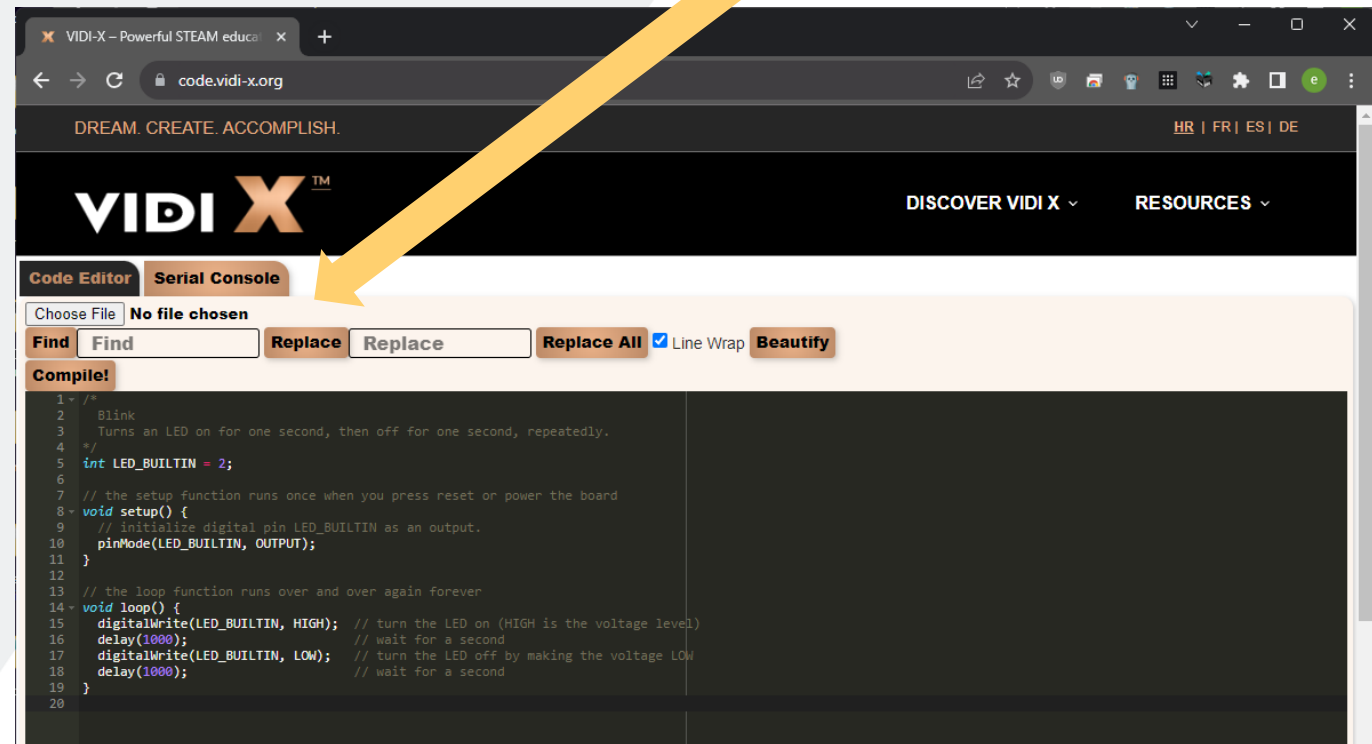
In the code input window, you can now type the code for the VIDI X microcomputer that you want to compile and transfer it compiled to the VIDI X microcomputer.

The easiest way to do this is with the Copy&Paste method, so mark the code from your lessons with the mouse and press CTRL+C to copy the code to the computer's clipboard.

Mark the window for entering the code with the left mouse click. Press CTRL+V to paste the code.

Now we have this situation:

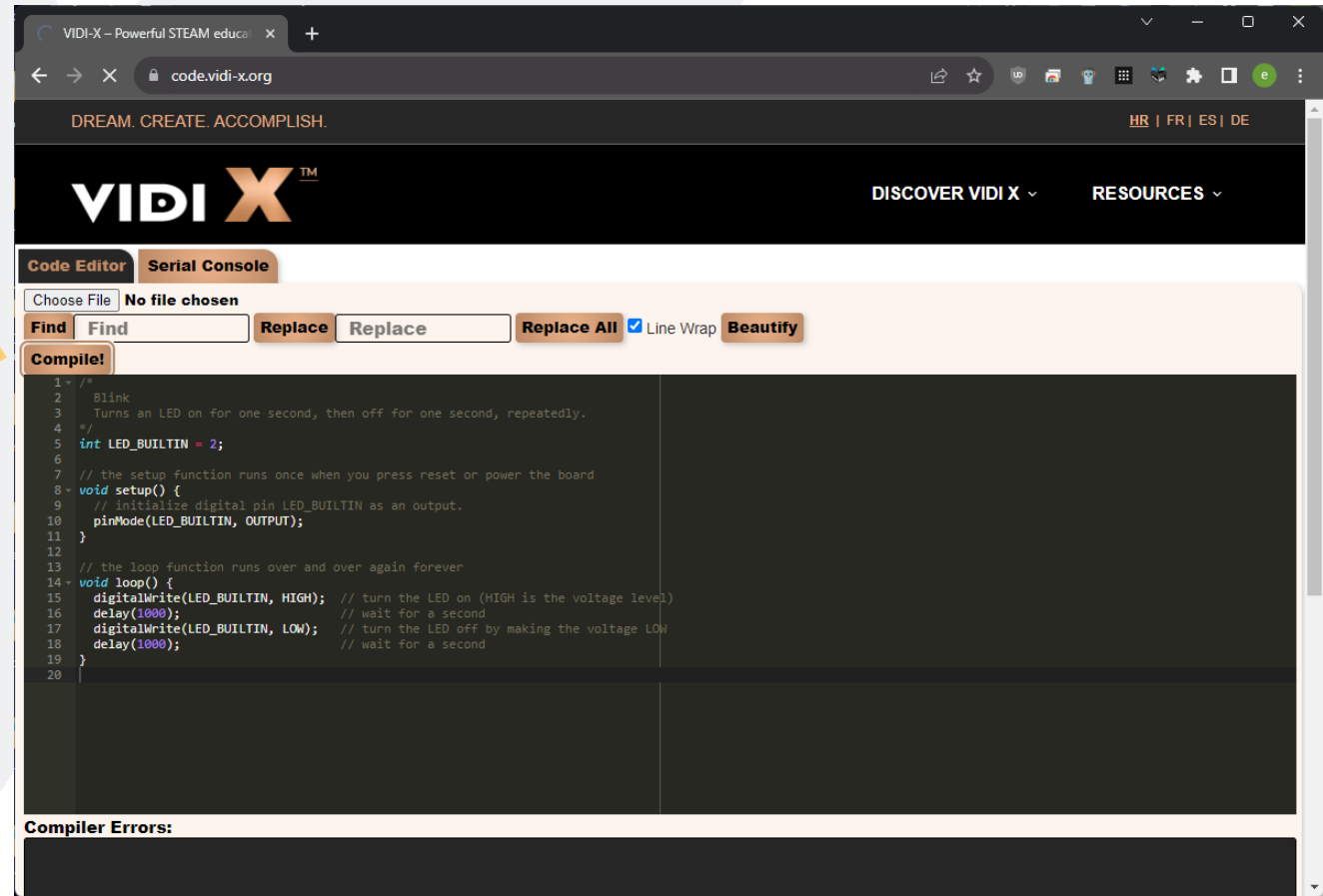
There is also a "Choose File" button to select *.ino files from your computer.



3 Step 3

Press the button "**Compile!**,"

In the title tab of the Internet browser, a rotating circular clip appeared, which, when in rotation, tells us that we should wait for the compilation process to finish.



Depending on the complexity of the code, this process may take some time.

After the compilation process was completed, we received a message about the size of the memory usage and other information.



The screenshot shows the VIDI X online compiler interface. The browser address bar displays 'code.vidi-x.org/compile'. The VIDI X logo is at the top left, with navigation links 'DISCOVER VIDI X' and 'RESOURCES' at the top right. The main area contains a code editor with the following code:

```
18 delay(1000); // wait for a second
19 }
20
```

Below the code editor, there is a section titled 'Compiler Errors:' which displays 'No Errors!'. Below that is a section titled 'Info about compile process:' which displays the following information:

```
1:-> Sketch uses 253097 bytes (19%) of program storage space. Maximum is 1310720 bytes.
2:-> Global variables use 21120 bytes (6%) of dynamic memory, leaving 306560 bytes for local variables. Maximum is 327680 bytes.
3:->
4:-> Used platform
5:-> esp32:esp32 2.0.11
```

The footer of the interface includes the VIDI X logo, navigation links 'HOME', 'DISCOVER VIDI X', 'LMS', and 'BLOG', and a copyright notice: 'ALL RIGHTS RESERVED / COPYRIGHT © VIDI X USA 2019 - 2023.' There are also links for 'SUPPORT' and 'PRIVACY POLICY'.

4

Step 4

We also have a new **"Upload!"** button, so press it!

VIDI X™

DISCOVER VIDI X ▾ RESOURCES ▾

Code Editor Serial Console

Choose File No file chosen

Find Find Replace Replace Replace All Line Wrap Beautify

Compile! Upload! ◀

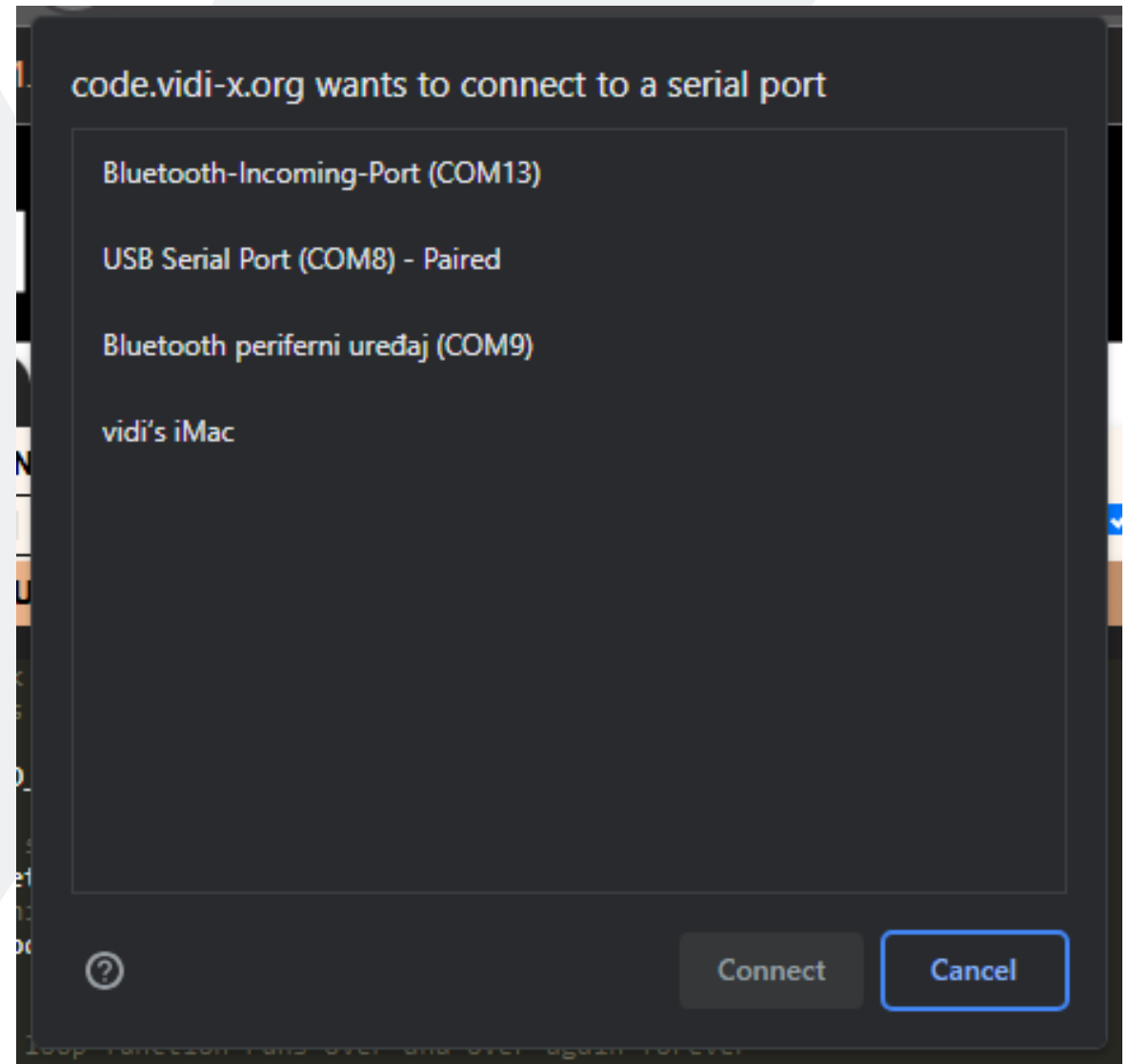
```
1 // This program turns an LED on for one second, then off for one second, repeatedly.
2
3 // Pin 2 has an LED on for one second, then off for one second, repeatedly.
4
5 int LED_BUILTIN = 2;
6
7 // the setup function runs once when you press reset or power the board
8 void setup() {
9   // initialize digital pin LED_BUILTIN as an output.
10  pinMode(LED_BUILTIN, OUTPUT);
11 }
12
13 // the loop function runs over and over again forever
14 void loop() {
15   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
16   delay(1000); // wait for a second
17   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
18   delay(1000); // wait for a second
19 }
20
```

Compiler Errors:

No Errors!

If a window similar to the image below appeared with a message that your device was not found, VIDI X is most likely not connected to your computer with a USB cable.

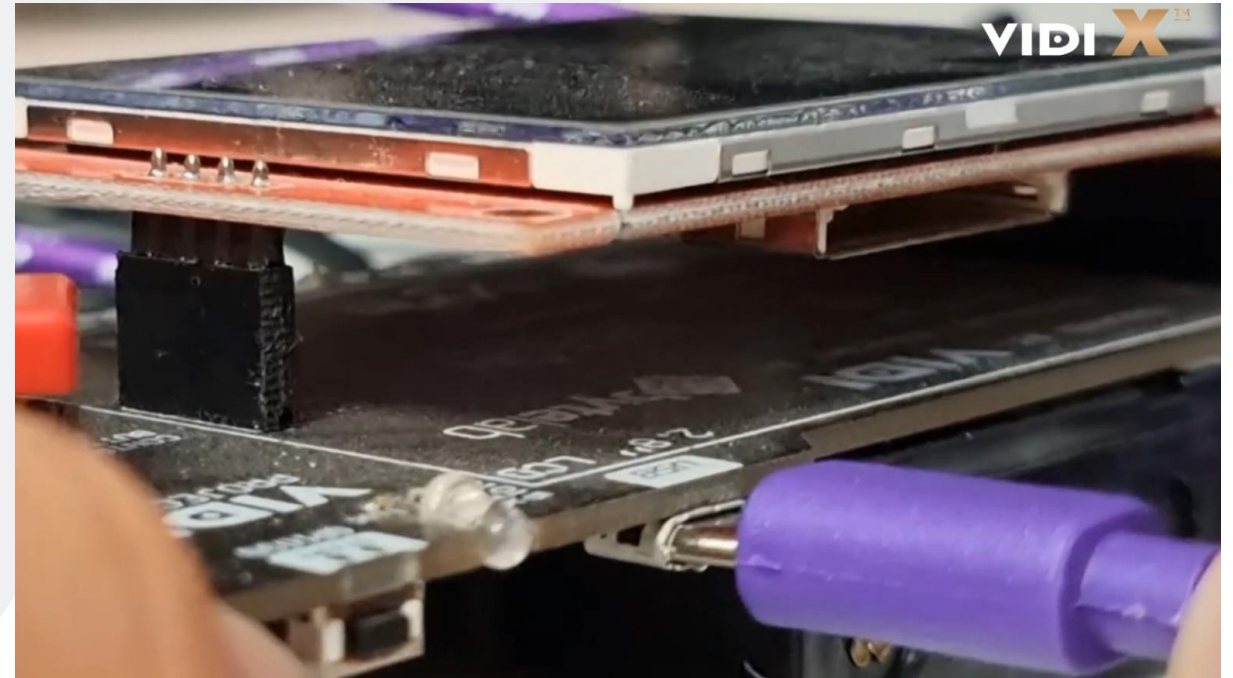
Check it and wait for a while so that the Windows operating system has enough time to find and install the corresponding driver.



5 Step 5

If you haven't already, now connect the VIDI X computer to your computer with a USB cable.

Most often, Windows will signal that a new USB device has been connected.

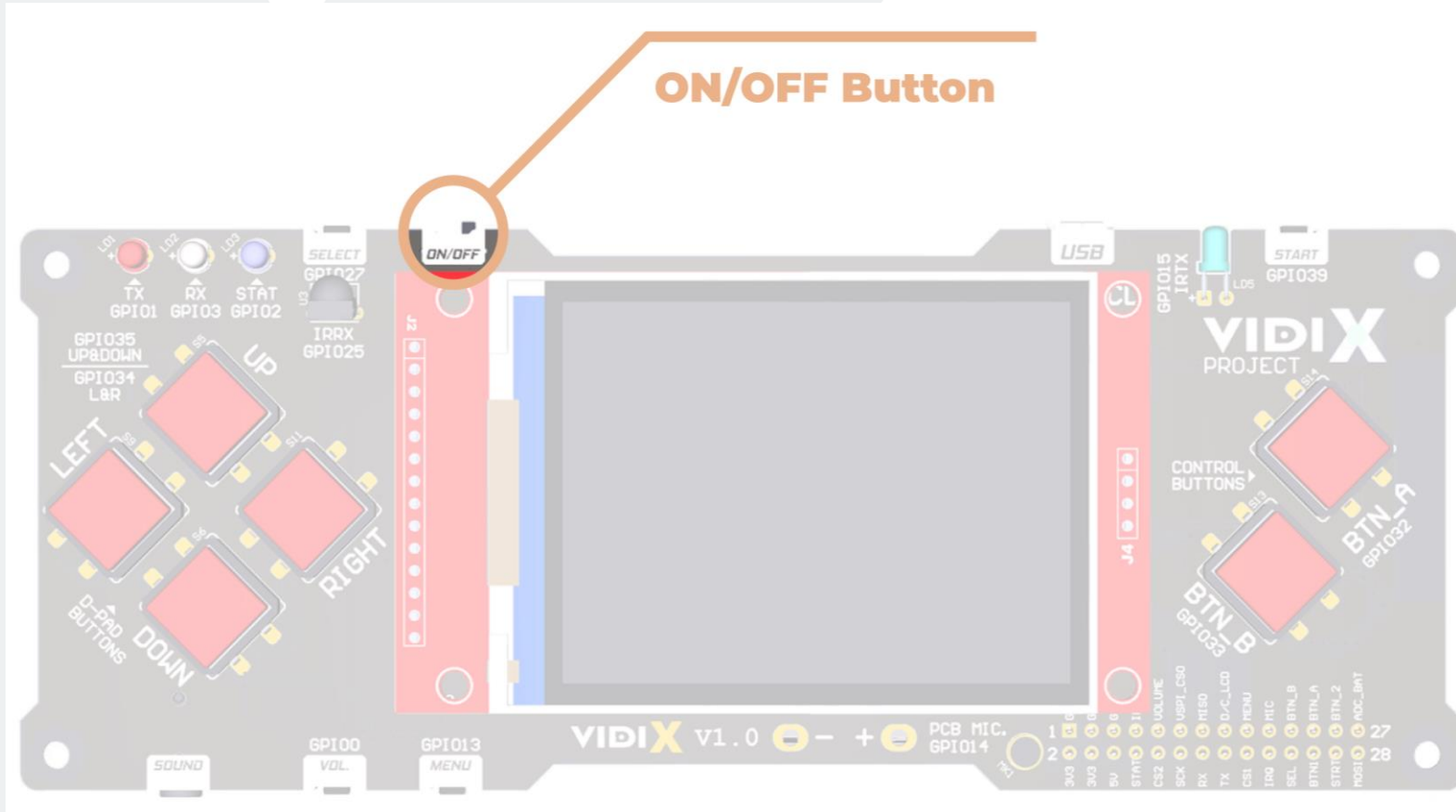


6 Step 6

Now the COM port of the connected VIDIX microcomputer has appeared in the newly opened window, as in the picture above.

Highlight it by left-clicking on it.

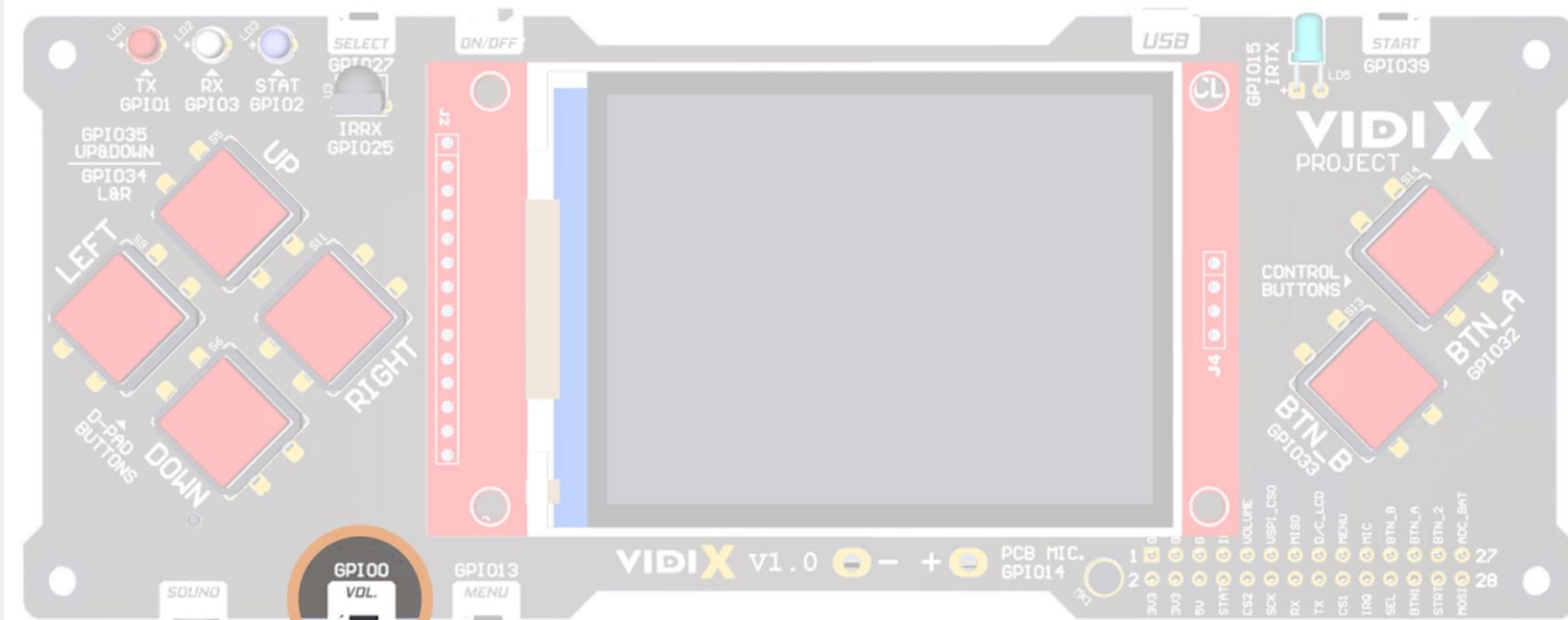
Before you click the "**Connect**" button turn off VIDIX, using the on/off button



7 Step 7

Press and hold the "Vol." button.

You will find the volume button on the left side of the screen, on the bottom edge of the VIDI X microcomputer, also marked as GPIO0.



VOL./BOOT Button
GPIO0

8 Step 8

Click "Upload!"



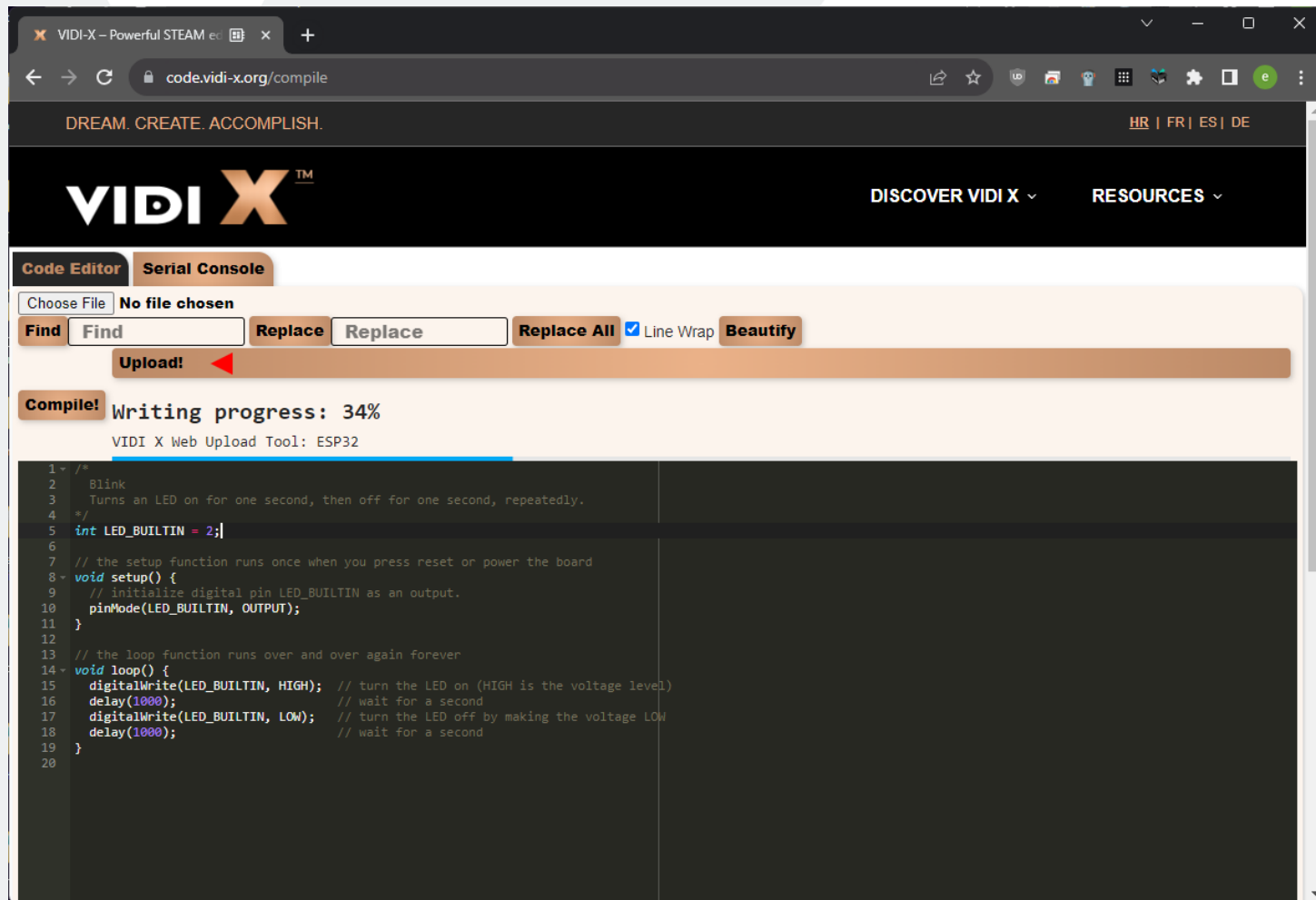
The screenshot shows the VIDI X web interface. At the top, there's a navigation bar with the VIDI X logo and links for 'DISCOVER VIDI X' and 'RESOURCES'. Below this is a toolbar with 'Code Editor' and 'Serial Console' tabs. The 'Code Editor' tab is active, showing a 'Choose File' button (with 'No file chosen' text), 'Find', 'Replace', 'Replace All', 'Line Wrap' (checked), and 'Beautify' buttons. A prominent 'Upload!' button is highlighted with a red arrow. Below the toolbar, there's a 'Compile!' button and the text 'Initializing...'. The main area contains a code editor with the following code:

```
1 /*  
2  * Blink  
3  * Turns an LED on for one second, then off for one second, repeatedly.  
4  */  
5 int LED_BUILTIN = 2;  
6  
7 // the setup function runs once when you press reset or power the board  
8 void setup() {  
9   // initialize digital pin LED_BUILTIN as an output.  
10  pinMode(LED_BUILTIN, OUTPUT);  
11 }  
12  
13 // the loop function runs over and over again forever  
14 void loop() {  
15  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
16  delay(1000); // wait for a second  
17  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
18  delay(1000); // wait for a second  
19 }  
20
```

9 Step 9

While holding the "Vol." button, turn on the VIDIX microcomputer by setting the ON/OFF button to the left "ON" position.

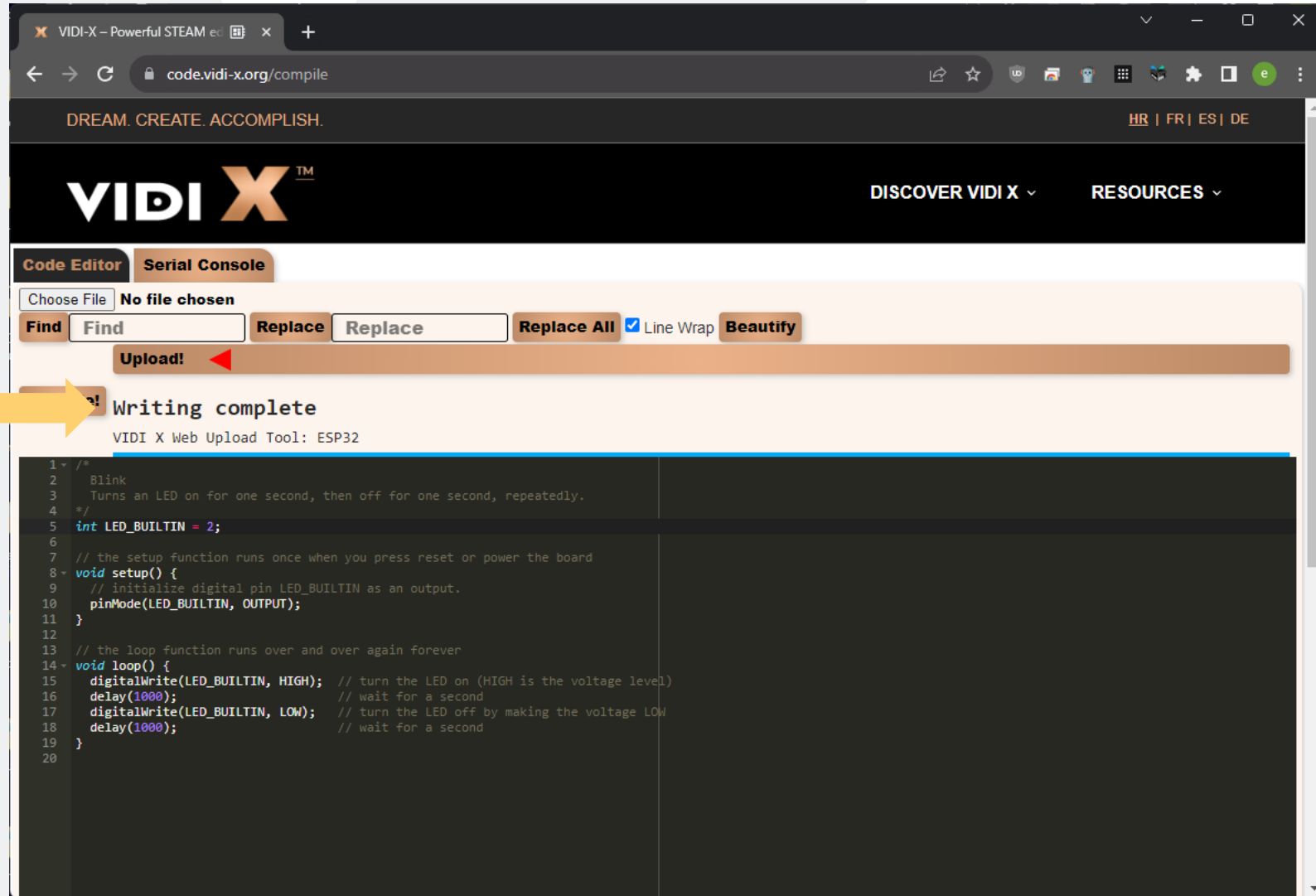
When the message "Initializing.." disappears, you can release the button "Vol." and start uploading the code.



The screenshot shows the VIDIX web upload tool interface. At the top, there's a navigation bar with the VIDIX logo and links for "DISCOVER VIDIX" and "RESOURCES". Below that, there are tabs for "Code Editor" and "Serial Console". The "Code Editor" tab is active, showing a code editor with a file named "No file chosen". There are search and replace fields, an "Upload!" button, and a "Compile!" button. The "Compile!" button is highlighted, and a progress bar below it shows "Writing progress: 34%". The code editor contains the following code:

```
1- /*
2-  * Blink
3-  * Turns an LED on for one second, then off for one second, repeatedly.
4-  */
5- int LED_BUILTIN = 2;
6-
7- // the setup function runs once when you press reset or power the board
8- void setup() {
9-   // initialize digital pin LED_BUILTIN as an output.
10-  pinMode(LED_BUILTIN, OUTPUT);
11- }
12-
13- // the loop function runs over and over again forever
14- void loop() {
15-  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
16-  delay(1000); // wait for a second
17-  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
18-  delay(1000); // wait for a second
19- }
20-
```

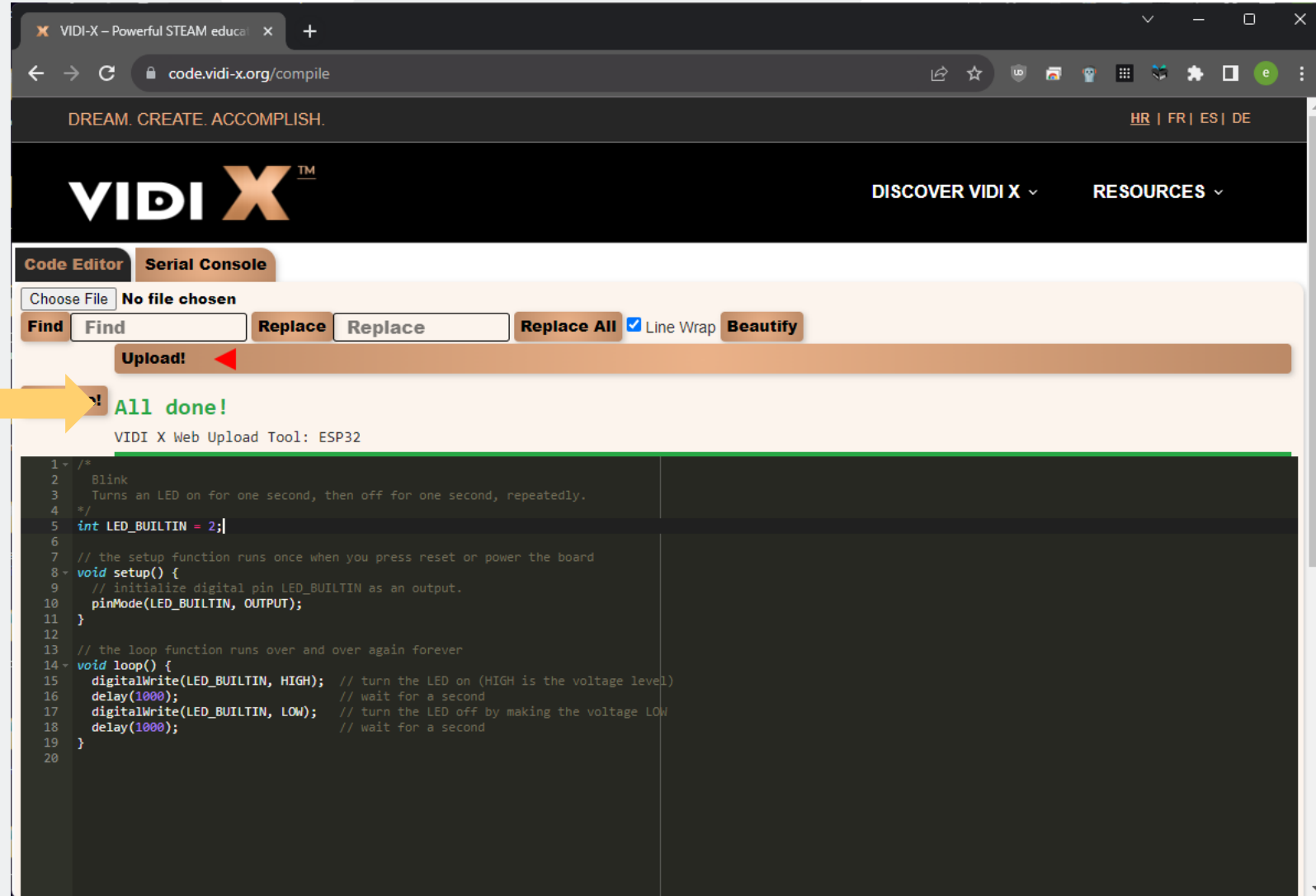
When the code was successfully transferred, we received a message confirming it.



The screenshot shows the VIDI X web interface. At the top, there is a navigation bar with the VIDI X logo and links for "DISCOVER VIDI X" and "RESOURCES". Below this is a "Code Editor" section with tabs for "Code Editor" and "Serial Console". The "Code Editor" tab is active, showing a code editor with a "Choose File" button and "No file chosen" text. There are also "Find", "Replace", "Replace All", "Line Wrap", and "Beautify" buttons. A prominent orange message bar with a red arrow pointing left contains the text "Writing complete" and "VIDI X Web Upload Tool: ESP32". Below the message bar, the code editor displays the following C++ code:

```
1 /*  
2  Blink  
3  Turns an LED on for one second, then off for one second, repeatedly.  
4  */  
5  int LED_BUILTIN = 2;  
6  
7  // the setup function runs once when you press reset or power the board  
8  void setup() {  
9    // initialize digital pin LED_BUILTIN as an output.  
10   pinMode(LED_BUILTIN, OUTPUT);  
11 }  
12  
13 // the loop function runs over and over again forever  
14 void loop() {  
15   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
16   delay(1000); // wait for a second  
17   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
18   delay(1000); // wait for a second  
19 }  
20
```

When the code was successfully transferred, we received a message confirming it.



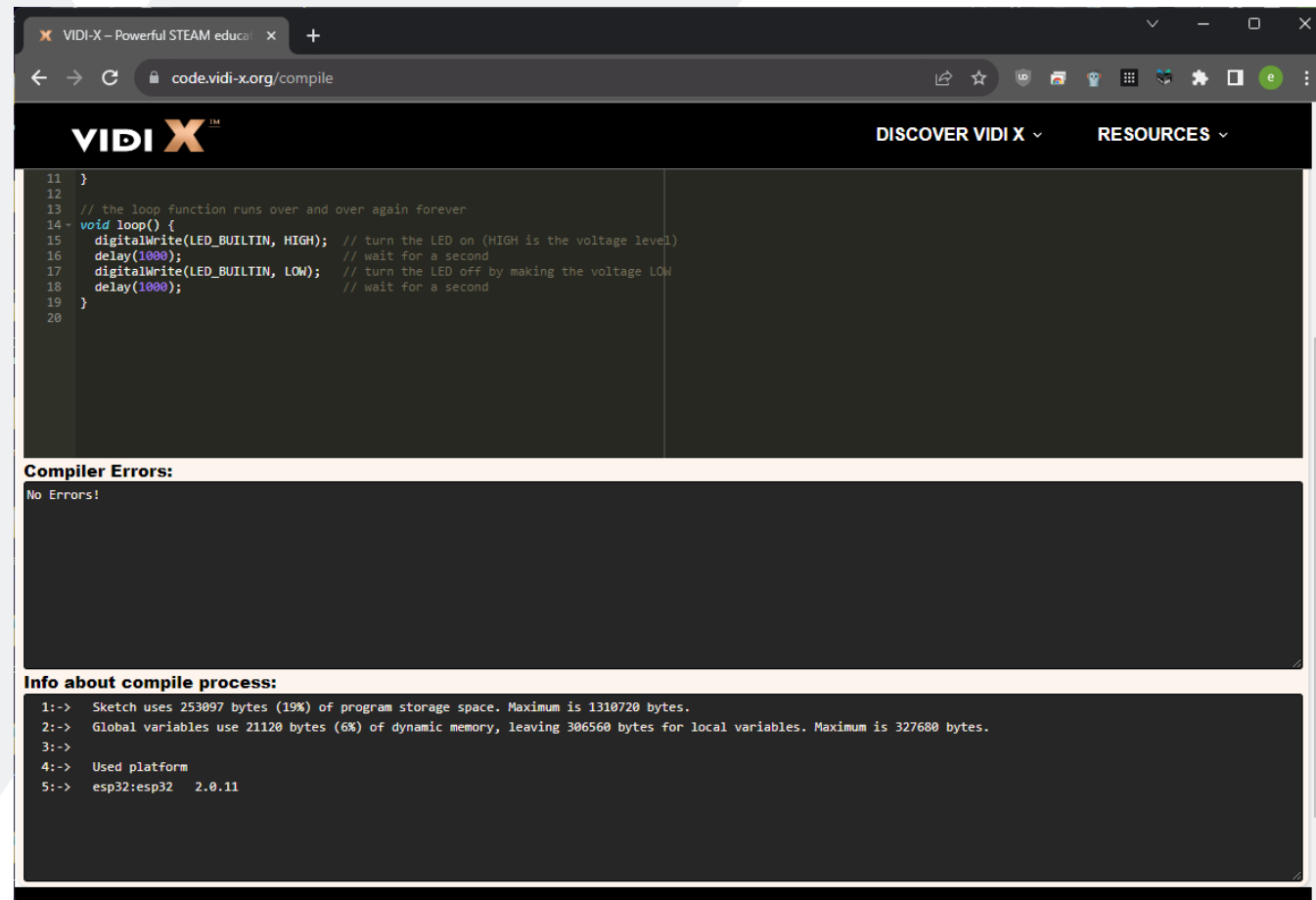
The screenshot shows the VIDI X web interface. At the top, there is a navigation bar with the VIDI X logo and links for "DISCOVER VIDI X" and "RESOURCES". Below this is a "Code Editor" section with tabs for "Code Editor" and "Serial Console". The "Code Editor" tab is active, showing a file upload area with a "Choose File" button and "No file chosen" text. Below the upload area are buttons for "Find", "Replace", "Replace All", "Line Wrap" (checked), and "Beautify". A prominent orange bar with a red arrow points to a green confirmation message that says "All done!". Below this message, it says "VIDI X Web Upload Tool: ESP32". The bottom part of the screen shows a code editor with C++ code for a blink function on an ESP32. The code is as follows:

```
1 /*
2  Blink
3  Turns an LED on for one second, then off for one second, repeatedly.
4  */
5  int LED_BUILTIN = 2;
6
7  // the setup function runs once when you press reset or power the board
8  void setup() {
9    // initialize digital pin LED_BUILTIN as an output.
10   pinMode(LED_BUILTIN, OUTPUT);
11 }
12
13 // the loop function runs over and over again forever
14 void loop() {
15   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
16   delay(1000); // wait for a second
17   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
18   delay(1000); // wait for a second
19 }
20
```

10 Step 10

Now release the "Vol." button and you will see the countdown continue to 100% where the code upload process is complete.

Your VIDI X microcomputer automatically runs the code you compiled!



The screenshot shows a web browser window with the URL `code.vidi-x.org/compile`. The page features the VIDI X logo and navigation links for "DISCOVER VIDI X" and "RESOURCES". The main area contains a code editor with the following C++ code:

```
11 }
12
13 // the loop function runs over and over again forever
14 void loop() {
15   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
16   delay(1000); // wait for a second
17   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
18   delay(1000); // wait for a second
19 }
20
```

Below the code editor, the "Compiler Errors:" section displays "No Errors!". The "Info about compile process:" section provides the following details:

```
1:-> Sketch uses 253897 bytes (19%) of program storage space. Maximum is 1310720 bytes.
2:-> Global variables use 21120 bytes (6%) of dynamic memory, leaving 306560 bytes for local variables. Maximum is 327680 bytes.
3:->
4:-> Used platform
5:-> esp32:esp32 2.0.11
```


Code

This tutorial uses the code shown below.

```
/*Blink
 Turns an LED on for one second, then off for one second, repeatedly.
*/
int LED_BUILTIN = 2;
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

 **VIDI X**
PROJECT

