

ARDUINO CNC WITHOUT MEGA

Creating a CNC (Computer Numerical Control) machine using an Arduino without the Mega model is possible, but it requires careful planning due to the limited number of pins and resources on smaller boards like the Uno or Nano. Here's a step-by-step guide to help you build a basic CNC machine using an Arduino Uno or Nano:

Components Needed

1. **Arduino Uno or Nano** (or similar)
 2. **Stepper Motors** (NEMA 17 or similar)
 3. **Stepper Motor Drivers** (e.g., A4988 or DRV8825)
 4. **Power Supply** (12V or 24V, depending on motors)
 5. **CNC Shield** (optional, but highly recommended for easier wiring)
 6. **Mechanical Parts** (linear rails, belts, pulleys, lead screws, etc.)
 7. **Endstops** (for homing)
 8. **Spindle or Laser Module** (for cutting/engraving)
 9. **Jumper Wires and Breadboard** (if not using a CNC shield)
-

Step 1: Choose Your CNC Design

- Decide the type of CNC machine you want to build (e.g., 3-axis CNC router, laser engraver, plotter).
 - Design or download a mechanical structure that fits your needs.
-

Step 2: Wiring the Electronics

1. **Stepper Motors:**
 - Connect each stepper motor to its respective driver (A4988 or DRV8825).
 - Ensure the motor coils are correctly wired to the driver.
2. **Stepper Motor Drivers:**
 - Connect the drivers to the Arduino:
 - STEP pin to a digital pin (e.g., D2, D3, D4).

ARDUINO CNC WITHOUT MEGA

- DIR pin to a digital pin (e.g., D5, D6, D7).
 - ENABLE pin can be connected to a digital pin or left unconnected if not used.
 - Connect the driver's VDD and GND to the Arduino's 5V and GND.
 - Connect the driver's VMOT and GND to the external power supply.
 - 3. **Power Supply:**
 - Ensure the power supply matches the voltage rating of your stepper motors.
 - Do not power the motors directly from the Arduino.
 - 4. **Endstops:**
 - Connect endstops to digital pins (e.g., D8, D9, D10) for homing.
 - 5. **Spindle/Laser:**
 - Connect the spindle or laser module to a PWM-enabled pin (e.g., D11) for speed control.
-

Step 3: Install GRBL Firmware

GRBL is a popular open-source firmware for CNC machines that runs on Arduino.

1. Download the GRBL library from the official GitHub repository: [GRBL GitHub](#).
 2. Open the Arduino IDE and load the grblUpload example sketch.
 3. Upload the sketch to your Arduino.
-

Step 4: Configure GRBL

1. Connect your Arduino to your computer and open a serial monitor (115200 baud rate).
 2. Send the command \$\$ to view and modify GRBL settings.
 3. Adjust settings like steps per millimeter, acceleration, and max speed to match your machine.
-

Step 5: Control the CNC

ARDUINO CNC WITHOUT MEGA

1. Use CNC control software like:
 - **Universal G-Code Sender (UGS)**
 - **Candle** (for laser engravers)
 - **bCNC**
 2. Load your G-code file and start the machine.
-

Step 6: Test and Calibrate

1. Test each axis individually to ensure smooth movement.
 2. Calibrate the steps per millimeter for accurate movement.
 3. Perform a test run with a simple design.
-

Tips for Using Smaller Arduino Boards

- **Pin Limitations:** The Uno/Nano has fewer pins, so you may need to use a CNC shield to expand connectivity.
- **Performance:** For more complex projects, consider upgrading to an Arduino Mega or a dedicated CNC controller.
- **Power Management:** Ensure your power supply can handle the current requirements of all components.

المكونات المطلوبة:

1. Arduino Uno
2. على الأقل 2 - محركات خطوية (Stepper Motors)
3. دائرة تحكم بالمحركات الخطوية (مثل A4988 أو L298N)
4. مصدر طاقة مناسب للمحركات
5. أسلاك توصيل

الكود:

```
cpp
Copy
تعريف المنافذ المستخدمة للمحركات الخطوية
const int stepPinX = 2;
const int dirPinX = 3;
```

ARDUINO CNC WITHOUT MEGA

```
const int stepPinY = 4;
const int dirPinY = 5;

void setup() {
    // تعيين المنافذ كمنافذ إخراج
    pinMode(stepPinX, OUTPUT);
    pinMode(dirPinX, OUTPUT);
    pinMode(stepPinY, OUTPUT);
    pinMode(dirPinY, OUTPUT);
}

void loop() {
    // في اتجاه واحد X تحريك المحور
    digitalWrite(dirPinX, HIGH); // تحديد الاتجاه
    for(int x = 0; x < 200; x++) { // خطوة لتحريك المحرك 200
        digitalWrite(stepPinX, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPinX, LOW);
        delayMicroseconds(500);
    }
    delay(1000); // انتظر لمدة ثانية

    // في الاتجاه المعاكس X تحريك المحور
    digitalWrite(dirPinX, LOW); // تغيير الاتجاه
    for(int x = 0; x < 200; x++) {
        digitalWrite(stepPinX, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPinX, LOW);
        delayMicroseconds(500);
    }
    delay(1000); // انتظر لمدة ثانية

    // في اتجاه واحد Y تحريك المحور
    digitalWrite(dirPinY, HIGH); // تحديد الاتجاه
    for(int y = 0; y < 200; y++) { // خطوة لتحريك المحرك 200
        digitalWrite(stepPinY, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPinY, LOW);
        delayMicroseconds(500);
    }
    delay(1000); // انتظر لمدة ثانية

    // في الاتجاه المعاكس Y تحريك المحور
    digitalWrite(dirPinY, LOW); // تغيير الاتجاه
    for(int y = 0; y < 200; y++) {
        digitalWrite(stepPinY, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPinY, LOW);
        delayMicroseconds(500);
    }
}
```

ARDUINO CNC WITHOUT MEGA

```
}  
delay(1000); // انتظر لمدة ثانية  
}
```

شرح الكود:

- **stepPinX** و **stepPinY**: المنافذ المستخدمة لإرسال نبضات التحكم إلى المحركات الخطوية.
- **dirPinX** و **dirPinY**: المنافذ المستخدمة لتحديد اتجاه دوران المحركات.
- لإرسال عدد معين من النبضات (خطوات) لتحريك المحرك for يتم استخدام حلقة.
- `delayMicroseconds(500)` يستخدم للتحكم في سرعة المحرك. يمكنك تغيير هذه القيمة لتعديل السرعة.

