Install Prerequisites using Scoop

1. **Install Scoop (if you don't have it):**
    - Open a **PowerShell** window (press `Win + X` and choose "Windows PowerShell" or "Terminal").
    - Run the following command to install Scoop. **Say 'Yes' to all prompts** when it asks to install NuGet and change execution policy.

Cmd:
```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
Invoke-RestMethod -Uri https://get.scoop.sh | Invoke-Expression
```

- **Install the Arduino CLI and Git using Scoop:**

    - In the same PowerShell window, run this command. Scoop will download both programs, install them in a central location (`C:\Users\[YourName]\scoop\`), and add them to your system PATH.

Cmd:
```
scoop install arduino-cli git
```

- **Initialize the Arduino CLI:**

    - Now that `arduino-cli` is correctly installed and in your PATH, run these commands to set it up for the Arduino Mega (AVR boards):

Cmd:
```
arduino-cli core update-index
arduino-cli core install arduino:avr
```

    - **Verify the install** by checking the version:

Cmd:
```
arduino-cli version
```

That's it! The `arduino-cli` and `git` commands are now permanently available in any PowerShell or Command Prompt window on your system. This is a much cleaner and more maintainable setup than the manual method.

Continue with the Plan

Now you can perfectly continue from **Phase 2** of the previous plan:

- **Phase 2:** Install **Visual Studio Code** and the recommended extensions (C/C++, GitLens, etc.).
- **Phase 3:** Use your new `git` command to `init` your project repository.
- **Phase 4:** Use your new `arduino-cli` command to compile and upload sketches from the VS Code terminal.

For example, once your project folder (`MyCNC_Controller`) is set up and you have a `src/main.cpp` file, you can run the build command from the VS Code terminal (which will be PowerShell):

powershell
```
arduino-cli compile --fqbn arduino:avr:mega --build-path ./build ./src
```

This corrected method using **Scoop** is the modern, efficient, and correct way to manage these developer tools on Windows. It avoids manual downloads, PATH editing, and makes future updates a single command (`scoop update arduino-cli`).

Again, my apologies for the initial error, and thank you for catching it. You now have the correct path to set up a truly professional environment.