# How to Setup and Debug PCHAIN on Windows (Go Language)

## 1. Install

a) Golang SDK for Windows x64 latest version

（ Recommend the latest one with at least 1.9 ）

https://golang.google.cn/dl/

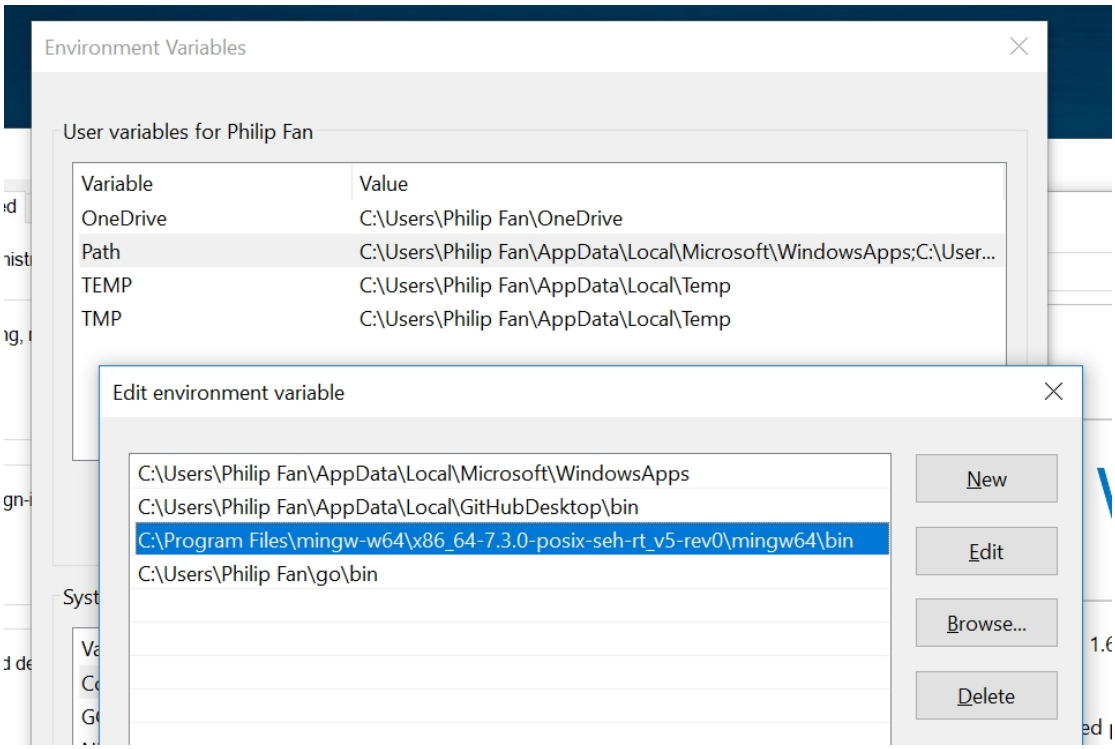
b) MingW GCC x64 Compiler

(It's used to compile Ethereum)

x86_64-7.3.0-release-posix-seh

https://sourceforge.net/projects/mingw-w64/files/Toolchains%20targetting%20Win32/Personal%20Builds/mingw-builds/installer/mingw-w64-install.exe/download

Path:



2. **Source code**

   git clone https://github.com/pchain-org/pchain.git

3. **Configure**

GoLand 2018

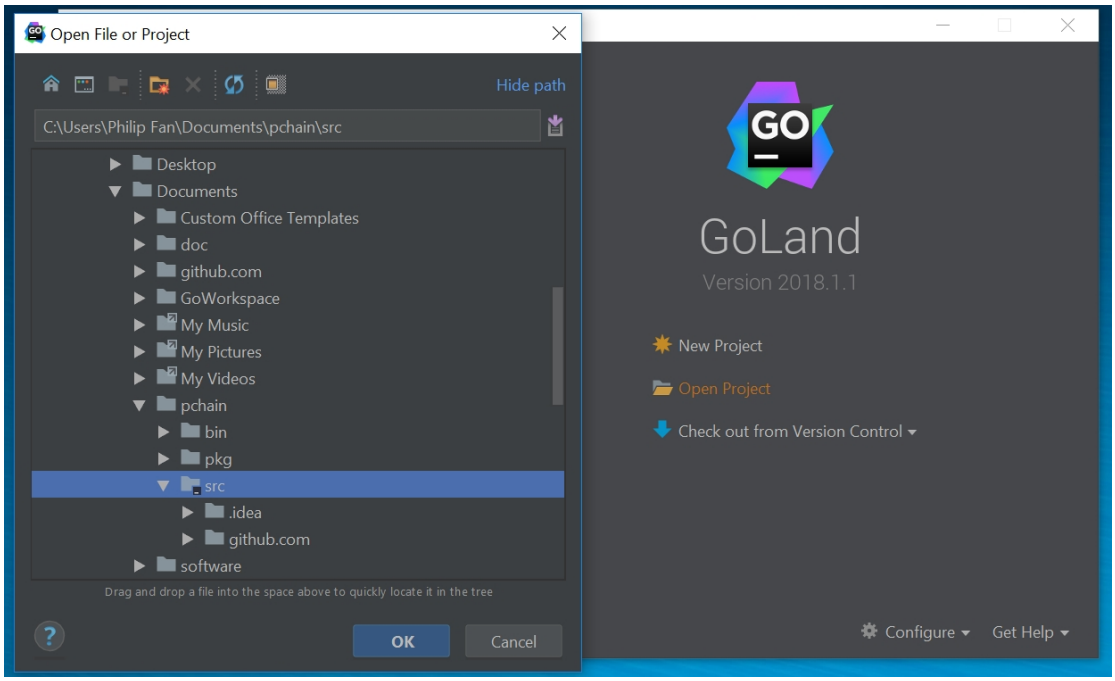
a) Create Go Workspace directory

Source code directory: pchain/src

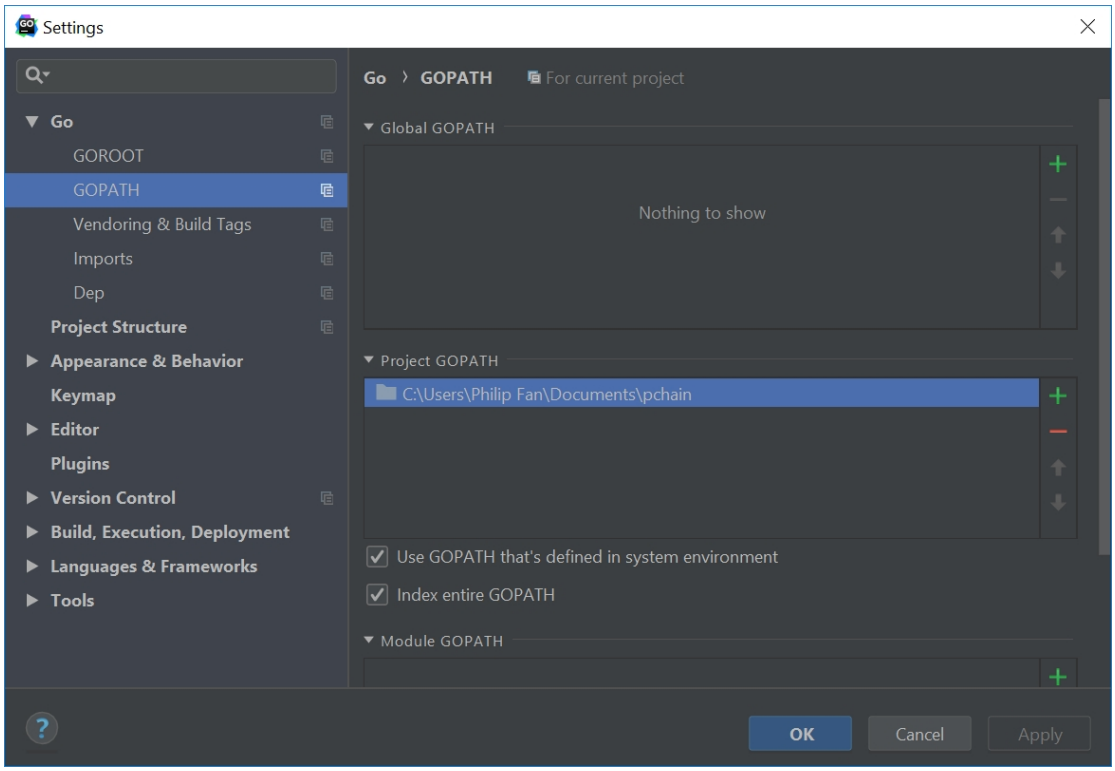
Executive document directory: pchain/bin

The document directory after compiling the source code: pchain/pkg

b) According to the readme, generate a directory named github.com\ under the src directory and transfer the source code to it. Please open this new directory with Goland and generate new projects.



c) In this project, please configure Project GoPATH as Root folder of current project so as to facilitate the code intelligent prompt, code compilation in the IDE and the automatic discovery of vendor directory.



4. **Build PCHAIN**

   a) **Setup Run Configuration before the first compilation.**

   Add New Configuration -> Go Build

   **Run kind**

   Please select Package ( or Directory based on the following different directories)
   **Package path**: github.com/pchain/cmd

   **Directory**: (There will be a window if you have selected Directory Compilation.)

   C:\Users\PhilipFan\Documents\pchain\src\

   **Output directory**: C:\Users\Philip Fan\Documents\pchain\bin

   **Run after build** (Do not select it.)

   **Go tool arguments**: -i -o "pchain.exe"

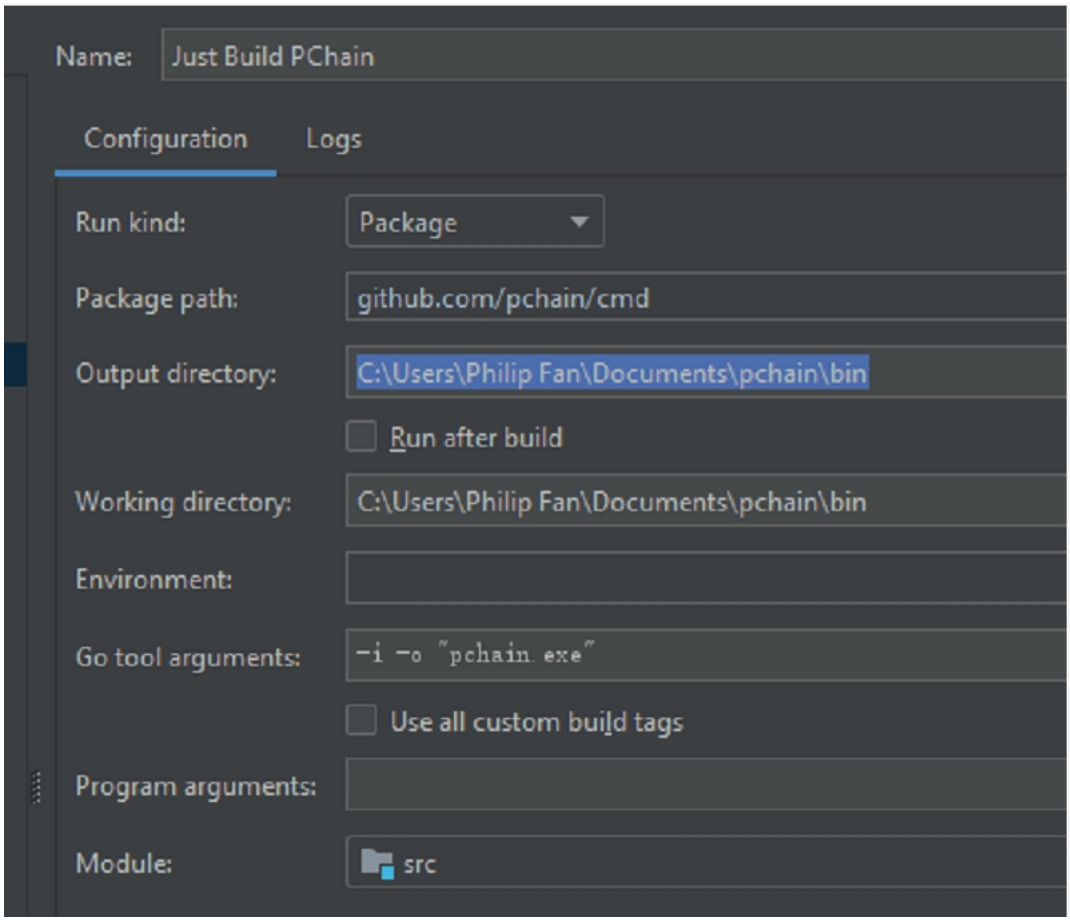
**After setup, execute this Run. You will see this kind of prompt below if you finish compilation successfully.**

GOROOT=C:\Go #gosetup

GOPATH=C:\Users\Philip Fan\Documents\pchain #gosetup

C:\Go\bin\go.exe build -i -o "C:\Users\Philip Fan\Documents\pchain\bin\pchain.exe" github.com/pchain/cmd #gosetup

**Compilation finished with exit code 0**

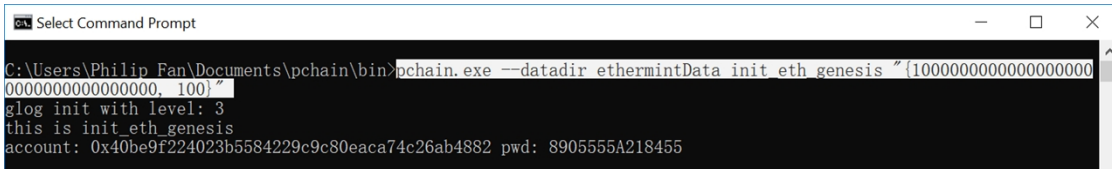



**b) Init PCHAIN**

Open Windows command and access to pchain bin directory

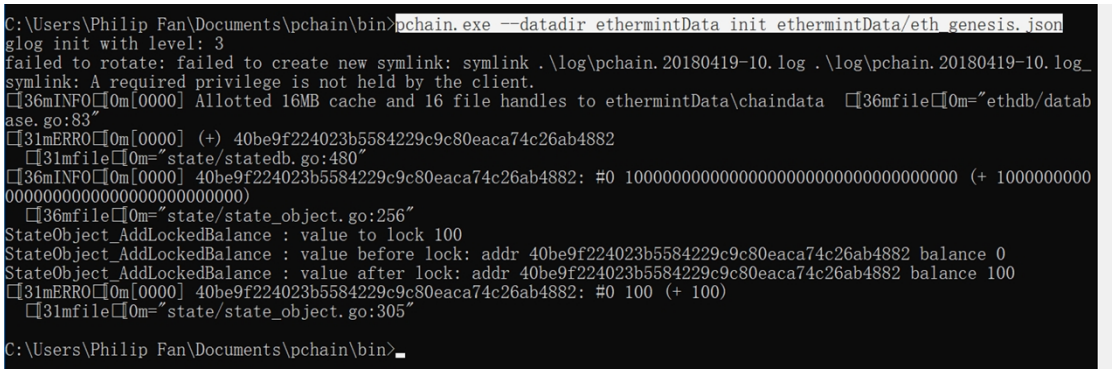
Init Ethereum

pchain.exe --datadir ethermintData init_eth_genesis "{1000000000000000000000000000000000000, 100}"




Init pchain genesis

pchain.exe --datadir ethermintData init ethermintData/eth_genesis.json




(The data directory can be modified as the case may be . The corresponding eth_genesis. Json path need to be modified as well.)

## 5. Debug PCHAIN

Generate a now Run/Debug Configuration inGoLand.

**Run kind**

Please select Package ( or Directory based on the following different directories)

**Package path**: github.com/pchain/cmd

**Directory**: (There will be a window if you have selected Directory Compilation.

C:\Users\PhilipFan\Documents\pchain\src

**Output directory**: C:\Users\Philip Fan\Documents\pchain\bin

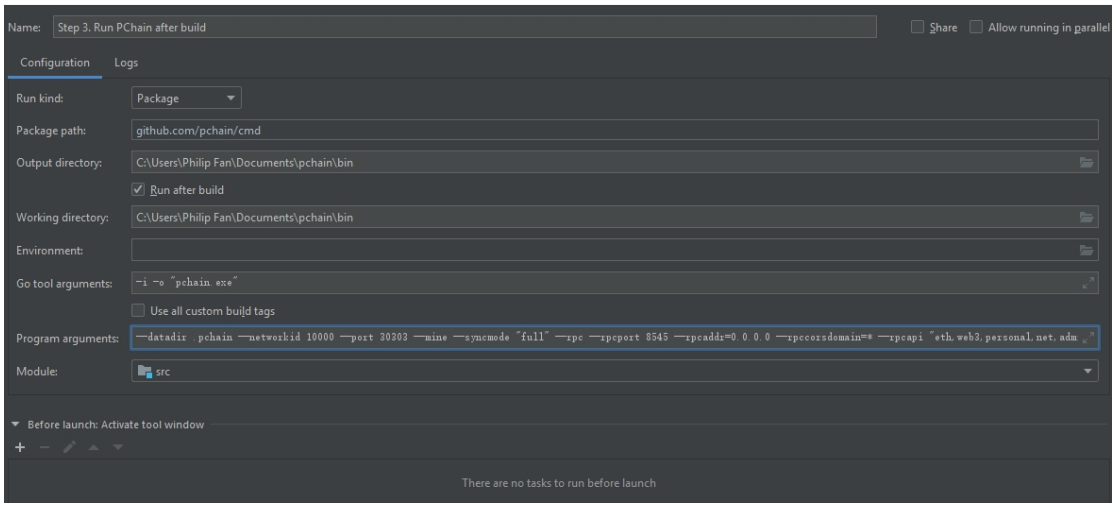
**Run after build**: ( Select i.)

**Working directory**: C:\Users\Philip Fan\Documents\pchain\bin (output directory)

**Go tool arguments**: -i -o "pchain.exe"

**Program arguments**: --datadir ethermintData --networkid 10000 --port 30303 --rpc --rpcport 8545 --rpcaddr=0.0.0.0 --rpccorsdomain=* --rpcapi "eth,web3,personal,net,admin,chain,tdm,miner,txpool" --verbosity=6

( The data directory remains the same as before, and the other port numbers can be modified as the case may be. )




After setup, please click“Ok” to save this configuration.

Once you have set a breakpoint in your code, select the configuration you just used and click

the Debug button.

**Enjoy your Debugging**!