$Q|SI\rangle$ Cheat Sheet

Shusen Liu mailto:shusen.liu88@gmail.com

1 Setup

Choose a suitable setup file for downloading.

1. `QSI_Online_Setup` is a small setup file using Nuget to get all required VS libraries. It also includes inductive packages of Visual Studio 2017 and Matlab Runtime 2017 on condition that you do have NOT installed VS 2017 and Matlab yet. The packages require network to finish installation.
2. `QSI_Offline_Basic_Setup` is a medium setup file equipped with all essential VS libraries. It also includes inductive packages of Visual Studio 2017 and Matlab Runtime 2017 on condition that you do have NOT installed VS 2017 and Matlab yet. The packages require network to finish installation.
3. `QSI_Offline_Full_Setup` is a huge setup file equipped with all essential VS libraries and Visual Studio 2017 Full version and Matlab Runtime 2017.The packages do not require network.

The following is a quick table:

Network	VS& Matlab	Choice
Good	Installed	`QSI_Online_Setup`
Good	NOT Installed	`QSI_Online_Setup`
Average or Bad	Installed	`QSI_Offline_Basic_Setup`
Average or Bad	NOT Installed	`QSI_Offline_Basic_Setup`
None	Installed	`QSI_Offline_Basic_Setup`
None	NOT Installed	`QSI_Offline_Full_Setup`

2 Start

After installation of $Q|SI\rangle$, you could start the provided examples in the software packages.

Please notice that the examples are not unified into the pure second generation language. Some codes are programmed within engine layer (basic) language for testing and debugging reason. These code segments are only for preview purpose.

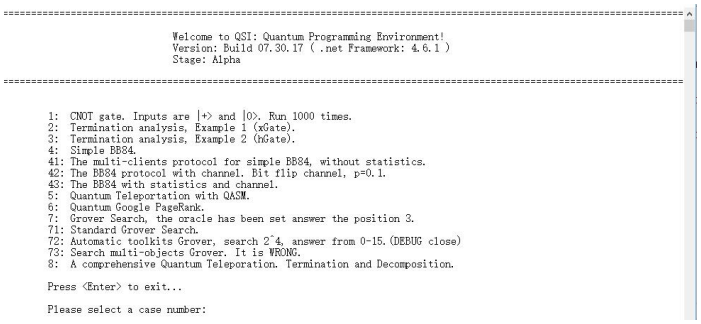

Figure 1: Welcome to $Q|SI\rangle$

1. Ensure the system environment has been installed correctly. The main program, Visual Studio and Matlab Runtime should be installed following the first step.
2. Menu → Debug → Start Debugging (or press F5).
3. Wait seconds. (It may consume more than 2 minutes to build the project at the first time of execution depending on your hardware.)
4. Eureka! You should find the display like Figure 1

There are two projects in the environment: QSI and UnitTest.

Project	Purposes	Howto
QSI	User Coding Area	Write Quantum Code in Test.cs and classical control code in Program.cs
UnitTest	Illustrating Examples	Study examples and languages

3 Examples

Examples can be found in Project UnitTest. All the examples can be traced to their corresponding source files.

Example	Function	Explain
CNOT gates	Generate Bell State	Using quantum language to construct algorithm
xGate	Termination Analysis	xGate, Termination
hGate	Termination Analysis	hGate, Almost sure termination
BB84	Protocol Simulation	Find behavior of BB84 under quantum channel
Teleportation	Using QIf & QWhile	Study Most powerful structures
Grover	Engine Layer Usage	Use engine layer to enrich flexible programming

4 Key Functions

You could start from QSI default and UnitTest. You may need other advanced features for customization the environment. Please use QSI as a template and configure it with classical codes. The key functions are provided in Section 3 of Manual.pdf and they can be used to adjust your program.

5 Contact

You can contact us using the Email:shusen.liu88@gmail.com. Any comments, suggestions and bugs are welcomed.