

Programming Quantum Computers in .NET using Q#

Rolf Huisman Software Development Enablement Engineer Rolf.Huisman@InfoSupport.com @rlrhuisman





Agenda

Why do you need it ? What is it? How can I program it ?

How would a Quantum Computer improve your life ?









Internet Of Stuffed Animals (IOSA)



Stuffed animal problem

Attic

Bedroom

Living Room













Stuffed animal problem

Attic

Search Path Optimization

Living Room



Same problem as other fields



Picture by Geralt / CC0 Creative Commons

Traveling Salesman problem



NP Hard







Add more hardware

No problem,

Can scale using cloud providers



Right ?







The power of exponents

O(2n*2ⁿ)

Using 1000 machines, only makes it 1000 times quicker,

if you're lucky

| o | •• | 0 0 0 0 | ::: | | | | 128 |
|-----|------|------------|------|------|------|-------|-------|
| 256 | 512 | 1024 | 2048 | 4096 | 8192 | 16384 | 32768 |
| 65K | 131K | 262K | 524K | 1M | 2M | 4M | 8M |
| 16M | 33M | 67M | 134M | 268M | 536M | 1G | 26 |
| 46 | 8G | 17G | 34G | 68G | 137G | 274G | 549G |
| | | | | | | | |
| | P | | | | | | |
| | | | | | | | |

And there are more fields that have this issue



Artificial Intelligence



Weather Prediction



DNA Simulations

New plan Quantum computing The computer that uses Qubits

New plan

- Quantum computing
- The computer that uses Qubits

Qubit can take value |0>, |1>, or **both at once**, a phenomenon known as **superposition**



A Computer with N Qubits, can be in a superposition of 2ⁿ logical states.

Speed

A Computer with N Qubits, can be in a superposition of 2ⁿ logical states.

So can work with these at the same time.

Multi-Billion Investments



What does a Quantum Computer look like ?



Pictures by Lars Plougmann / CC BY-SA 2.0

IBM Q

| Backend: QS1_1 (20 Qubits) | ACTIVE AVAILABLE TO HUBS, PARTNERS, AND HI | | | | | |
|--|--|---------------|---------------|----------------------|--------------------|--|
| 9 -9-9 9-79 | | QO | Q1 | Q2 | Q3 | |
| | Frequency (CHu) | 4.84 | 448 | 4.86 | 5.0 | |
| 0-0-0 | T1 (ps) | 71.96 | 61.08 | 73.91 | 68. | |
| | T2 (ps) | 33.97 | 28.54 | 25.02 | -36. | |
| | Gate error (10 ⁻¹) | 2.01 | 2.56 | 6.64 | 7.2 | |
| | Readout error (10 ⁻¹) | 7.70 | 27.80 | 19.60 | 20.4 | |
| | MultiQubit gate error (10-7) | CH0_1 2.67 | CKL_0 2.67 | 602,1 3.34 | CK3 7.01 | |
| 0-0-0-0 | | CHD_5 | CK1_2 | CH216 | 71 | |
| Last Calibration: 2018-03-01 20:24:56 | | | 00.6 | 02.7 | | |
| | | | 3,90 | 4.20 | | |
| | | | CK1_7 | | | |
| Backend: ibmgx5 (16 0abits) | | | | | - | |
| | | | | | | |
| | | QO | Q1 | Q2 | Q3 | |
| | Frequency (SHc) | 5.26 | 5.40 | 5.28 | 5.0 | |
| | T1 (rd) | 39.30 | 36.40 | 42.30 | 51. | |
| | 12 (90 | 34.00 | 59.90 | .ə.r.,ıq | - 1 -1-1 | |
| Last Calibration: 2018-03-01 13:35:31 | Gate error (10 5) | 2.17 | 3.36 | 3.66 | 1.6 | |
| Fridge Temperature: 0.0133661 K | Readout error (10 ⁻¹) | 6.32 | 6.17 | 3.91 | 6.8 | |
| More details | MultiDubit date error (UP.7) | | CX1_0 | 602,3 | | |
| | | | 5.00 | 2.41 | 0.0 | |
| | | | 419 | | 4.3 | |

https://quantumexperience.ng.bluemix.net



MERS OF THE ISH O NETWORK

78.55

1.95

1.90

Q6 4.89 87.03

38:12

14.90

CHA_1 3.90 CHA_2 7.05

CX6_11 3.67

06

5.31 50.50

47.50 80.40

CHS_4 CX6_5

1.81

CKS_T

CH6_11

CKS_10 CK6_5 2.34 2.72

CH5_11 CH6_7 2.75 3.67 CH6_10

AVAILABLE ON GIVENT

05

3.63

\$ 55

54.20

6.60

"Quantum Accelerator card"



QC Picture by by Lars Plougmann / CC BY-SA 2.0 Card Picture by Dave Fischer / CC BY-SA 3.0





















The Qubit in more detail



Bloch Sphere

Picture by IBM / Licensed Use

The Qubit in more detail



Bloch Sphere

Picture by IBM / Licensed Use

M (Measurement Gate)



Picture by IBM / Licensed Use


X Gate (Not)



Picture by IBM / Licensed Use



H Gate (Hadamard)



Picture by IBM / Licensed Use

One Qubit Gates

- X-Gate (Bit-flip, Not Gate) Y-Gate
- Z-Gate (Phase flip)

H-Gate (Hadamard, Creates Superposition) S-Gate, T-Gate (Phase shift)

Multiple Qubit Gates CNOT (Controlled Not Gate)



Multiple Qubit Gates

CNOT (Controlled Not Gate)





Multiple Qubit Gates

CNOT (Controlled Not Gate)





Multiple Qubit Gates CNOT (Controlled Not Gate)





Multiple Qubit Gates CNOT (Controlled Not Gate)





How can I program for this ?

Microsoft Q#

Revealed at Ignite 2017 Microsoft Quantum (former Station Q) Previously known as F# extension LIQUi >

Microsoft Quantum labs and locations





Santa Barbara









Microsoft Q#

| | 21.11 | | | | | | 1 | × |
|-----------|---|--------|----------------------|---------------------------|-----------|---------------------------|---|------|
| | Recent | JNET F | amework 4.6.1 • | Sort by: Default | • # E | Search (Ctri+E) | | p. |
| | Installed | 1 | Console App (NET | Framework) | Visual C# | Type: Visual C# | | |
| - | Wisual C Windows Classic Desktop Web | - | | | Visual C# | A project for testing qui | | ons. |
| teb | .NET Core | - 2511 | Class Library (NET I | Framework) | Visual C# | | | |
| | Cloud Extensibility | 9 | ASP.NET Core Web | | Visual C# | | | |
| | Test | | ASP NET Web Appli | lication (JNET Framework) | Visual C# | | | |
| 1 | Visual F# COL Secure | 5î | Shared Project | | Visual C# | | | |
| 0; | R b Bathan | - | Class Library (Legac | | Visual C# | | | |
| t. Iri | Python Other Project Types | 0 | Azure Cloud Service | e | Visual C# | | | |
| .1e * | Online | 0 | Q# Application | | Visual C# | | | |
| | | 0 | Q# Library | | Visual C# | | | |
| | Not finding what you are looking for? Open Visual Studio Installer | ð | Q# Test Project | | Visual C# | | | |

DEMO



Grover (and other amplifications)



Grover execution: informally \sqrt{states}

ReflectionOracle user defined type

Namespace: Microsoft.Quantum.Canon

Represents a reflection oracle O, where the inputs are

- The phase φ by which to rotate the reflected subspace.
- The qubit register on which to perform the given reflection.

newtype ReflectionOracle = ((Double, Qubit[]) => () : Adjoint, Controlled);

AmpAmpByOracle function

Namespace: Microsoft.Quantum.Canon

Standard Amplitude Amplification algorithm

Q#

OF

function AmpAmpByOracle (nIterations : Int, stateOracle : StateOracle, idxFlagQubit : Int) : (Qubit[] => () : Adjoint, Controlled)

Copy Copy

Dy Copy

Reversing on the real device Real vs Simulated for x + y = 1



Quantum State: Computation Basis



Stuffed animal problem

Attic

Bedroom

Living Room





Using the gates to describe



Optimized Traveling Salesman Problem 5 Cities:

| RpI:-0.238 | - | | | | | | | | Rp22.0.048 | Pp22.0.048 | Ry22.0.048 | - Rp22.0.048 | - Ry22.0.048 | Haman | - RpZ-0.03453 | - RpX:-0.258 | 7- |
|-------------|------------|------------|------------|------------|--------------|------------|--------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|---------------|--------------|----|
| RpI:-0.258 | | | | | | | Ry22.0.048 | PpZZ-0.048 | Rp22-0.048 | | Rp22.0.048 | - | | Rp22.0.048 | - RpZ-0.03395 | - RpX:-0.258 | 7- |
| RpI:-0.238 | 0 | | | | | Ry22.0.048 | Ap22.0.048 | | Rp22.0.048 | - | | Rp22-0.048 | - Ap2Z-0.048 | RpZ-0.03327 | ApX:-0.258 | | - |
| RpX:-0.238 | | | | | Rp22.0.048 | Ry22.0.048 | | Pp22.0.048 | | APRZ.0.048 | Ry22.0.048 | Rp22.0.048 | RyZ:-0.03521 | - RoX -0.258 |] | | _ |
| RpX:-0.238 | | | | | Pp22.0.048 | Ry22.0.048 | - Ry22.0.048 | - | Rp22.0.048 | | Ry22.0.048 | RpZ:-0.036 | - BpX-0.238 | | 94 | | - |
| RpX:-0.238 | | | | Ry22.0.048 | - Rozz.0.048 | H | | APTE O Del | Rp22.0.048 | Pp22.0.048 | RpZ:-0.01529 | RpX:-0.258 | | | | | - |
| Rp.E:-0.238 | | 3 1 3 | Rp22.0.048 | Ap22.0.048 | <u> </u> | Ap22.0.048 | Ap22.0.048 | Pp72:0.048 | Rp22.0.048 | - RpZ-0.0348 | ApX:-0.258 | | | | | | |
| RpX:-0.258 | | Rp22:0.048 | | | Rp22-0.048 | Rp22.0.048 | - Ry22.0.048 | Rp22:0.048 | RpZ-0.03432 | - RpX-0.258 | | | | | | | |
| RpX:-0.258 | | | MART O DAG | Ry22.0.048 | Rp22-0.048 | Ry22.0.048 | - Ry22.0.048 | RpZ-0.03495 | RpX:-0.238 | | | | | | | | _ |
| RpX:-0.238 | Apr5.0.048 | Rp22.0.048 | 8422.0.048 | Ry22.0.048 | Rp22-0.048 | Ry22.0.048 | RpZ-0.03438 | ApX:-0.238 | | | | | | | | | - |

Details: Phys. Rev. A **95**, 032323 – Published 22 March 2017

Oracle Function

(8) 11>

One Stair

Oracle (State)



Two Stairs

Many hours and code lines later

| | Traditional | Quantum |
|-------------|-----------------------|------------------|
| Naive | O(<i>n</i> !) | $O(\sqrt{n!})$ |
| With effort | O(2n*2 ⁿ) | $O((\sqrt{n})!)$ |

Stuffed animal solution

Attic

Bedroom

Living Room



Picture by Rolf Huisman / CC0 Creative Commons

Low-Tech Solution



Conclusions

Quantum Computing

- 1. Its not simple
- 2. But its powerful
- 3. It will change the world
- 4. Is currently unstable
- 5. Is cool to program for
- 6. One can use existing tooling
- 7. And it works on real hardware \odot



So be part of the quantum future !

Questions ?

Rolf Huisman Rolf.Huisman@infosupport.com @rlrhuisman





Microsoft Q# (Operators)



Microsoft Q# (Driver)

| N | Quantur | nMiner - Microsoft Visual Studio |
|------|----------|---|
| File | Edit | View Project Build Debug Team Tools Test R Tools Analyze |
| 2. | | |
| 2 | .01 | To + a a a Debug + Any CPO + Runner |
| Ī | Tests.qs | Operation.gs Driver.cs + X IbmQx5.cs |
| B | C# Runne | r |
| × | 1 | Dusing Microsoft Quantum Simulation Core: |
| ē | 2 | using Microsoft Quantum Simulation Simulators: |
| at E | 3 | using System: |
| Đ. | 4 | Losing Systemy |
| ore | 5 | - namespace Quantum, Runner |
| | 6 | 1 |
| | | 0 references Rolf Huisman, 1 hour ago 1 author, 4 changes |
| | 7 | public class Driver |
| | 8 | { |
| | | O references Rolf Huisman, 1 hour ago 1 author, 4 changes |
| | 9 | <pre>static void Main(string[] args)</pre> |
| | 10 | |
| | 11 | <pre>//var sim = new IbmQx5();</pre> |
| | 12 | <pre>using (var sim = new QuantumSimulator())</pre> |
| | 13 | |
| | 14 | = for (Int 1 = 0; 1 < 1000; 1++) |
| | 15 | une casult - Madamas Dum(sim) Pasults |
| | 17 | Console Writeline(result): |
| | 18 | console.writeLine(result); |
| | 19 | |
| | 20 | Console Read ine(): |
| | 21 | } |
| | 22 | |
| | 23 | |

Microsoft Q# (Emulated Run)

| Tests.qs | Operation.qs | Driver.cs # + × IbmQx5.cs # |
|----------|------------------------|------------------------------------|
| C Runner | | |
| 1 | Eusing Microsoft | .Quantum.Simulation.Core; |
| 2 | using Microsoft | .Quantum.Simulation.Simulators; |
| 3 | using System; | |
| 4 | SIG PORTO CONSTRUCTION | |
| 5 | Enamespace Quant | um.Runner |
| 6 | { | C:\Program Files\dotnet\dotnet eve |
| 1.1 | O references P | olf Hulan |
| 7 | public class | s Drilone |
| 8 | 1 Australia | One |
| 9 | E static | void t |
| 10 | 1 | Zero |
| 11 | 1/1 | one sila |
| 12 | E usi | |
| 13 | 1 | Zero |
| 14 | | for Zano |
| 15 | | Zero |
| 16 | | 2000 |
| 17 | | One |
| 18 | | } Zero |
| 19 | } | One |
| 20 | Con | sole. One |
| 21 | } | One |
| 100 % + | | Zero |
| Autos | | Zero |
| history | | Zero |
| Name | | Zero |
| | | Zero |
| | | One |
| | | One |
| | | Zero |
| | | Zero |
| | | One |
| | | One |
| | | One |
| | | Zero |
| | | - |

Microsoft Q# (Unit Test)

| QuantumMiner - Microsoft Visu File Edit View Project Bulk | ial Studio d Debug Team Tools Teit R Tools Analyze Window Help |
|---|---|
| 0-0 8-0 8-0 | - C - Debug - Any CPU - Runner Runner - 🗯 👘 🐄 🐄 🤫 🐄 |
| Text Explorer • • • • × • Itt • 曰 Search P • Run All Run. • Paylist: All I • Passed Tests (2) • AllocateQubitTest 18 ms • HadamarTest 87 ms | <pre>Testags * X Operation.gs Onveros RumQuS.cs i namespace Quantum.RummerTest 2 () 3</pre> |

Microsoft Q# (VSTS Build)

| 🗘 Crypto 🗸 Das | hboards Code Work Build and Release Test Will 🔘 | | | | | |
|----------------------------------|--|--|--|--|--|--|
| Builds Releases Packages Library | Task Groups Deployment Groups* | | | | | |
| Build 20180228.1 | Cryptominer / Build 20180228.1 / Phase 1 | | | | | |
| Phase 1 | 🖋 Edit build definition 🛛 Cancel 😸 Queue new build. 🔹 🔒 Eliminitat at logs as op | | | | | |
| 🗸 ivitalize Agent | Build Started | | | | | |
| ✓ Initialize Job ✓ Get sources | Phase 1 A Running for 113 seconds (Hosted Agent) | | | | | |
| V Use Nuclet 4.4.1 | Console Timeline Code coverage* Tests | | | | | |
| Nuclet restore | Linger and Lowell Line, and you the distance any timelity you have in this | | | | | |
| @ Build solution | state without impacting any branches by performing monther checkout. If you want to create a new branch to retain commits you create, you may do to (new or later) by using -b with the checkout command again. Example: git checkout -b come branch-name: #000 is new at Mikebol And project files. | | | | | |
| & Test Assemblies | | | | | | |
| & Publish symbols path | | | | | | |
| 🔗 Publish Artifact | | | | | | |
| 🧬 Port Icb Cleanup | Starting: Use Nutet 4.4.1 | | | | | |
| | Task : Nulet Tool Installer Description : Acquires a specific version of Nulet from the internet or the tools cache and adds it to the FAIM. Use this step to c Version : Ficrosoft Corporation Selp : [More Information](https://go.microsoft.com/Fulime//limid=852538) | | | | | |
| | Dewloading: https://dist.muget.org/win-x00-communilize/v4.4.1/muget.exe Caching Soci: Mudet 4.4.1 x04 Using Version: 4.4.1 Found tool in cache: NuGet 4.4.1 x04 Using tool path: D:va_toolVMudet:4.4.1\x04 Propending PATH environment variable with directory: D:va_toolVMudet:4.4.1\x04 | | | | | |
| | Finishing: Due Nudet 4.4.1 Starting: Nudet restore | | | | | |

Microsoft Q# (VSTS Test)

| Cryptominer / Build 20180228.3 | |
|--|--|
| P Edit Suid Administra 😸 Queue new Suid. * 🚊 Devented al tops in siz 🔒 Result Indefentery 🍸 Release | - |
| Duild succeeded | |
| Build 20180228.3 R Ran for 3.3 minutes (Hosted VS2017), completed 2.2 minutes ago | |
| ummary Tenetine Code coverage* Tests | |
| Build details Onfortion Organisme settl Source Sour | Test Results Test Results Table tests Pass percentage Run duration Image: State of the sta |
| Directory Gruit/Tur' is empty Nothing will be added to build artiflect drop: Associated changes No changes associated with this build. | Deployments |
| Work items linked to associated changes | |

Production (IBM Q)

Backend: QS1_1 (20 Qubiti)



| | QO | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 |
|--|---|--|--|--|--|--|--|
| Frequency (CHu) | 4.84 | 4.48 | 4.86 | 5.03 | 5.01 | 4991 | 4.89 |
| T2 (µs) | 33.97 | 28154 | 25.02 | 36.77 | 28.43 | \$1,32 | 38.12 |
| Gate error (10 ⁻¹) | 2.01 | 2.54 | 6.64 | 7.27 | 1.96 | 1.95 | 1.96 |
| Readout error (10") | 7,70 | 27,80 | 19.60 | 20.65 | 9.55 | 8.90 | 14.90 |
| fultiQubit gate error (10-7) | CH0_1 2.67 | CK1_0 2.67 | 3.34 | 7.06 | 7.06 | CR5_0 1.73 | 3.90 |
| | CHD_5 1.73 | 6X1,2 7.64 | CH2_6 7.86 | 603,9 7.33 | 6.95 | 2.72 | CX6_2 7.05 |
| | | 00114 | 002,7 | | 00429 | CK5_10 | 016_5 |
| | | 3070 | 4.20 | | 3.54 | 2124 | 2.12 |
| | | CALLET | | | | 2.75 | 3.67 |
| | | 0.14 | | | | | |
| | | | | | | | C06,10 3.11 |
| | | 2.74 | | | | | 046_31 3.11 046_31 3.67 |
| | | | | EACTION | | DARLE OF | 046_3 3.11 046_3 3.67 |
| | QQ | 01 | Q2 | 03 | Q4 | Q5 | 044_31 3.11 044_11 3.67 001940 |
| Frequency (DHz) | Q0 5.26 | Q1 5.40 | Q2 5.28 | Q3 5.00 | Q4 4.10 | Q5 5.15 | 044_3 3.11 044_1 3.67 (01940 0 0 5.31 |
| Frequency (1040) T1 (pi) T2 (pi) | Q0 5.26 39.30 34.50 | Q1 5.40 35.40 53.60 | Q2 5.28 42.30 57.70 | Q3 5.08 51.80 73.50 | Q4 4.98 53.30 54.20 | Q5 5.15 43.50 47.50 | Q6 5.31 Q6 5.31 50.50 90.40 |
| Frequency (DHu) T1 (µ) T2 (µ) | Q0 5.26 39.30 34.60 | Q1 5.40 35.40 55.60 | Q2 5.28 42.30 57.70 | Q3 5.08 51.80 73.50 | Q4 33.30 54.20 | Q5 5.15 43.50 47.50 | CK4_1 3.11 CK4_1 3.67 Q6 5.31 50.50 80.40 |
| Frequency (BHs) T1 (µ) T2 (µ) T2 (µ) Gate error (10 ⁻¹) Readout error (10 ⁻¹) | Q0 5.26 39.30 34.60 2.17 6.32 | Q1 5.40 35.40 53.60 3.36 6.17 | Q2 5.28 42.30 57.70 3.46 3.91 | Q3 5.08 51.80 73.50 1.60 6.82 | Q4 4.98 53.39 54.20 1.35 6.60 | Q5 5.15 43.50 47.50 2.43 5.35 | CX4_1 3.11 CX4_1 3.67 Q6 5.31 50.50 50.40 1.81 4.42 |
| Frequency (UHs) T1 (ps) T2 (ps) Gate error (10 ⁻¹) Readout error (10 ⁻¹) Autojubit gate error (10 ⁻¹) | Q0 5.26 39.30 34.60 2.17 6.32 | Q1 5.40 35.40 53.60 3.36 6.17 CKL 0 5.00 | Q2 5.28 42.30 57.70 3.46 3.91 cm2_3 3.47 | Q3 5.08 51.80 73.50 1.60 6.82 cc1,4 3.26 | Q4 4.98 33.30 54.20 1.35 6.60 | Q5 5.15 43.50 47.50 3.43 5.35 cx3.4 5.57 | CK4_1 3.11 CK4_1 3.67 Q6 5.31 50.50 00.40 1.81 4.42 CK4_5 5.23 |
| Frequency (UH4) T1 (pa) T2 (pa) T2 (pa) Gate error (10 ⁻¹) Readout error (10 ⁻¹) MultiQubit gate error (10 ⁻¹) | Q0 5.26 39.30 34.60 2.17 6.32 | Q1 5.40 35.40 55.60 3.36 6.17 CXL0 5.00 CXL2 4.19 | Q2 5.28 42.30 57.70 3.46 3.91 cs2,3 3.47 | Q3 5.08 51.80 73.50 1.60 6.82 C03.4 3.26 C03.4 4.37 | Q4 4.98 33.30 54.20 1.35 6.60 | Q5 5.15 43.50 47.50 3.63 5.35 cas,4 5.57 | C04_1 3.11 C04_1 3.67 Q6 5.31 50.50 80.40 1.81 4.42 C04_5 5.23 C04_7 2.90 |



Backend: ibmqx5 (16 Qubits)



Last Calibration: 2018-03-01 13:35:31 Fridge Temperature: 0.0133661 K

More details

Production (Work in progress)

| Operation.qs | Driver.cs = + × IbmQx2.cs = | |
|--------------|---|---|
| C Runner | | + 🔩 Quantum.Runner.Driv |
| 1 | ⊡using Microsoft.Quantum.Simulation.Core; | |
| 2 | using Microsoft.Quantum.Simulation.Simulators; | |
| 3 | using System; | |
| 4 | | C:\Program Files\dotnet\dotnet.exe |
| 5 | Enamespace Quantum.Runner | |
| 6 | { | QUASM file |
| | Oreferences Rolf Huisman, Less than 5 minutes ago 1 autho | include delibi.inc; |
| 7 | public class Driver | dreg d[2]; |
| 8 | | creg c[5]; |
| 0 | static upid Main(string[] acrs) | |
| 10 | static void hain(scring[] args) | measure q[0] -> c[0] |
| 11 | ares = new string[] { "" }: | Processing |
| 12 | IOperationFactory factory; | One |
| 13 | if (args.Length != 0) | one |
| 14 | { | OUASM file |
| 15 | <pre>factory = new IbmQx2();</pre> | include "gelib1.inc": |
| 16 | } | areg a[5]: |
| 17 | else | creg c[5]; |
| 18 | { | H q[0] |
| 19 | factory = new QuantumSimulator | (measure q[0] -> c[0] |
| 20 | } | |
| 21 | Contraction in the second s | Processing |
| 22 | for (int i = 0; i < 10; i++) | One |
| 23 | { | and the second se |
| 24 | var result = Hadamar.Run(facto | CQUASM file |
| 25 | Console.WriteLine(result); | include "gelib1.inc"; |
| 26 | } | qreg q[5]; |

Production (Work in progress)



Bitcoin Mining 101



Bitcoin Mining 101



SHA 256



Reduced Adder





x1
Quantum Adder



Quantum Adder





Adder Algorithm



Inverted Adder Algorithm



Reverse Adder

Reset(qubits[1]);
Reset(qubits[2]);

return (a,b, error);

operation ReverseAdder (first: Bool, second: Bool) : (Result, Result, Result) body mutable a = Zero; mutable b = Zero; mutable error = Zero; using (qubits = Qubit[3]) let outA = qubits[0]; //end garbage ? = 2(One, One, Zero) let outB = qubits[1]; //also inB = 2(Zero, Zero, One) let inA = qubits[2]; ? = 2(One, One, Zero) //Data Consc ? = 2(Zero, Zero, One) if(first){ X(outA); } //Invese A to expect the first bit to be one if(second){ X(outB); } //Inverse B to expect the last bit to one = 2(Zero, Zero, One) //Rev // Put waste into superposition for = 1(Zero, One, Zero) H(inA); = 1(One, Zero, Zero) = 1(One, Zero, Zero) //perform addition in reverse CNOT(inA, outB); //XOR with B in out A = 1(Zero, One, Zero) let inB = outB; = 1(Zero, One, Zero) CCNOT(inA, inB, outA); //And a and b in outB = 0(Zero, Zero, Zero) //Read the outputs = 0(Zero, Zero, Zero) set a = M(inA); set b = M(inB); = 0(One, One, One) set error = M(outA); ? = 0(One, One, One) //Cleanup + ? = 0(Zero, Zero, Zero) Reset(qubits[0]);

Reverse Adder

operation ReverseAdder (first: Bool, second: Bool) : (Result, Result, Result) body mutable a = Zero; mutable b = Zero; mutable error = Zero; using (qubits = Qubit[3]) let outA = qubits[0]; //end garbage let outB = qubits[1]; //also inB let inA = qubits[2]; //Data if(first){ X(outA); } //Invese A to expect the first bit to be one if(second){ X(outB); } //Inverse B to expect the last bit to one // Put waste into superposition H(inA); //perform addition in reverse CNOT(inA, outB); //XOR with B in out A let inB = outB; CCNOT(inA, inB, outA); //And a and b in outB //Read the outputs set a = M(inA);

set b = M(inB); set error = M(outA);

//Cleanup

Reset(qubits[0]); Reset(qubits[1]); Reset(qubits[2]);

return (a,b, error);

| 1 | ? + ? = 2(One, One, Zero) |
|----------------|-----------------------------|
| } | ? + ? = 2(One, One, Zero) |
| Conso //Rev | |
| for | ? + ? = 1(Zero, One, Zero) |
| { | ? + ? = 1(One, Zero, Zero) |
| 1 | ? + ? = 1(One, Zero, Zero) |
| | ? + ? = 1(Zero, One, Zero) |
| | ? + ? = 1(Zero, One, Zero) |
| | ? + ? = 0(Zero, Zero, Zero) |
| | ? + ? = 0(Zero, Zero, Zero) |
| | |
| | ? + ? = 0(Zero, Zero, Zero) |

Reverse Adder





Dirty Adder (Not Reduced)



Tolifo Gate





Adder (Not Reduced)

