“A Practical Architecture for Reliable Quantum Computers”

written by Mark Oskin, Fredric T. Chong and Isaac L. Chuang

presented by Ketan Patel

March 11, 2002

Outline

- Overall Goal
- Quantum Error Correction
- Quantum Computer Architecture
- Conclusions

Goal

Provide a general-purpose architecture for quantum computation

- quantum storage
- quantum ALU
- data paths
- classical control circuits
- system integration

Important consideration: Reduce error-correction overhead

Quantum Error Correction

QEC can be used to combat the effects of decoherence and noisy gates

Single error correcting code decreases error prob. from $p \Rightarrow cp^2$

Recursively applying: $p \Rightarrow cp^2 \Rightarrow c(cp^2)^2 \Rightarrow \ldots \Rightarrow (cp)^{2^k}/c$
Error decreases exp. while increase in overhead is “only” poly.

Error Correction Overhead

Recursive QEC can introduce large overheads

For example (using the Steane $[7,1]$ code):

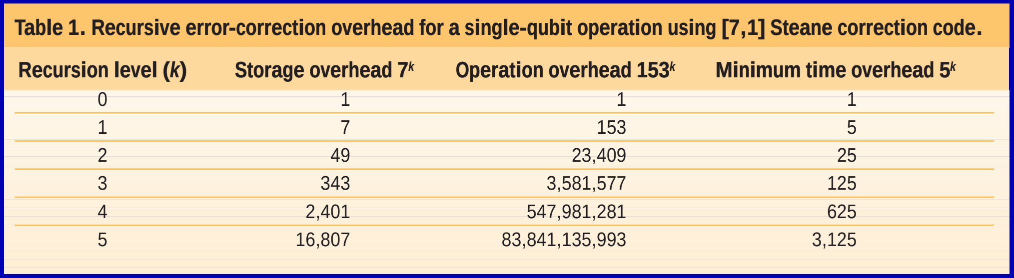


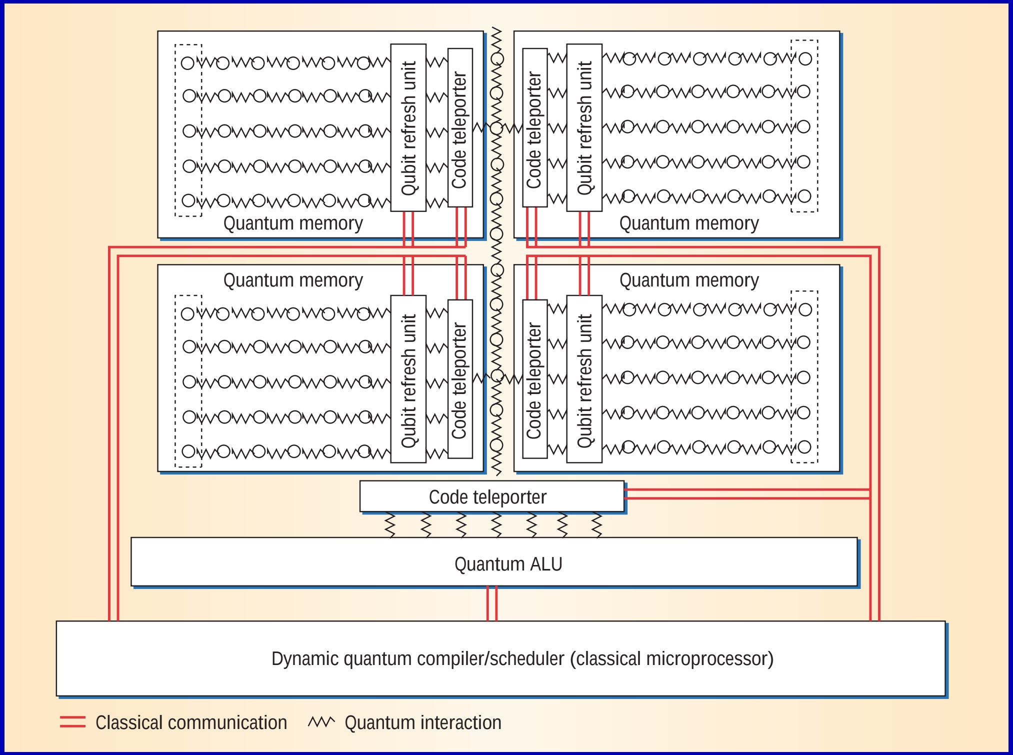
Table 1. Recursive error-correction overhead for a single-qubit operation using [7,1] Steane correction code.

Recursion level (*k*)	Storage overhead 7^k	Operation overhead 153^k	Minimum time overhead 5^k
0	1	1	1
1	7	153	5
2	49	23,409	25
3	343	3,581,577	125
4	2,401	547,981,281	625
5	16,807	83,841,135,993	3,125

Critical to only use as much error correction as necessary

Quantum Computer Architecture

- Classical Compiler/Scheduler
- Quantum ALU
- Quantum Memory
- Quantum Wires
- Code Teleporter



Compilers

Static Precompiler

- Generates code for target error rate on ideal quantum computer
- No knowledge of error model

Dynamic Compiler

- Produces instructions for FT-computation
- Dynamically determines necessary error correction

Dynamic Scheduler

Dynamically translates logical quantum ops ⇒ physical qubit ops

- Uses knowledge of input data size & physical qubit error rates
- Controls quantum ALU, code teleportation & memory refresh units

Critical to making architecture efficient—should be fast

Quantum ALU

Performs elementary ops fault-tolerantly on encoded states

- Hadamard
- Identity
- bit flip (X)
- phase flip (Z)
- combined bit & phase flip (Y)
- phase (S)
- $\pi/8$ (T)
- C-NOT (CNOT)

Specialized HW provides fresh ancilla

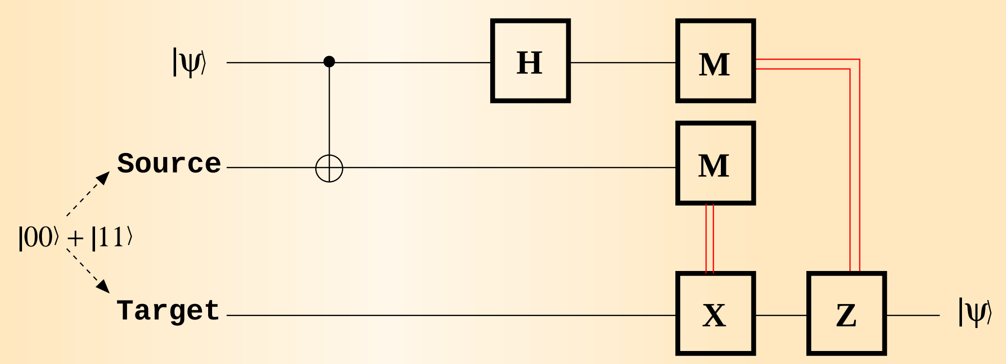
Quantum Memory

Key: Memory should be more reliable than computation
(Could make use of decoherence-free subsystems)

Logical qubits periodically “refreshed” with dedicated hardware

Quantum Wires

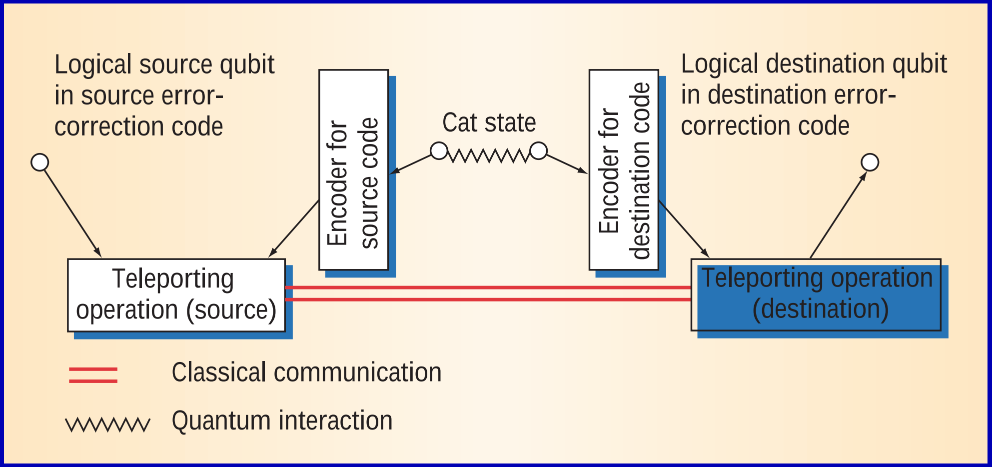
Use quantum teleportation to move qubits



No need to transmit qubits, only shared cat states and classical bits

Code Teleportation

Use teleportation to convert between codes

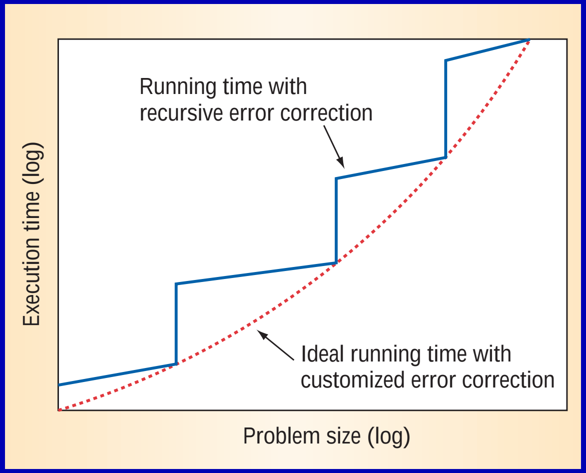


Can use space-efficient code for memory and operation-efficient code for computation

Error Correction Optimization

Recursive EC increases in steps

Leads to unnecessarily large overhead



Classical processor aggregates cost of EC over several ops

Conclusions

- Practical architecture will require error rates btwn 10^{-6}—10^{-9}
- Reliability of underlying technology crucial
- Error correction overhead is most pressing issue