Logic gates in Quantum Computing

@CSIRO Co-learnium 16/12/21 noon

DANIEL BURGARTH, MQ

* Intro to QC for astronomers to engineers to management and finance
* 20+10 min
* over Zoom
* at RO

NO TIME FOR CONCEPTS AND IDEAS
ONLY DETAILS & EQUATIONS

Logic gates in normal computers

BITS: tiny switches that can be in two configurations $\{0, 1\}$

"lightswitch"

off → 0, on → 1

physical

GATES: elementary changes ("operations") applied to bits

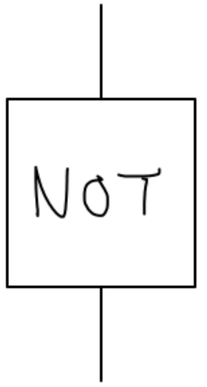
"flip switch"

Example: "NOT" gate $0 \to 1$ $1 \to 0$

Table:

x	not(x)	id(x)	off	on
0	1	0	0	1
1	0	1	0	1

Graphic:



are these all?

NOT MUCH OF INTEREST WITH 1 BIT

Multi - Bit Gates ...

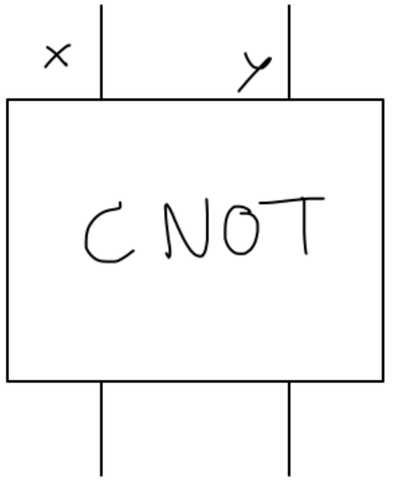
much more interesting

CNOT:

"control - not"

x	y	CNOT	
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

"if kitchen light on, flip bedroom light"



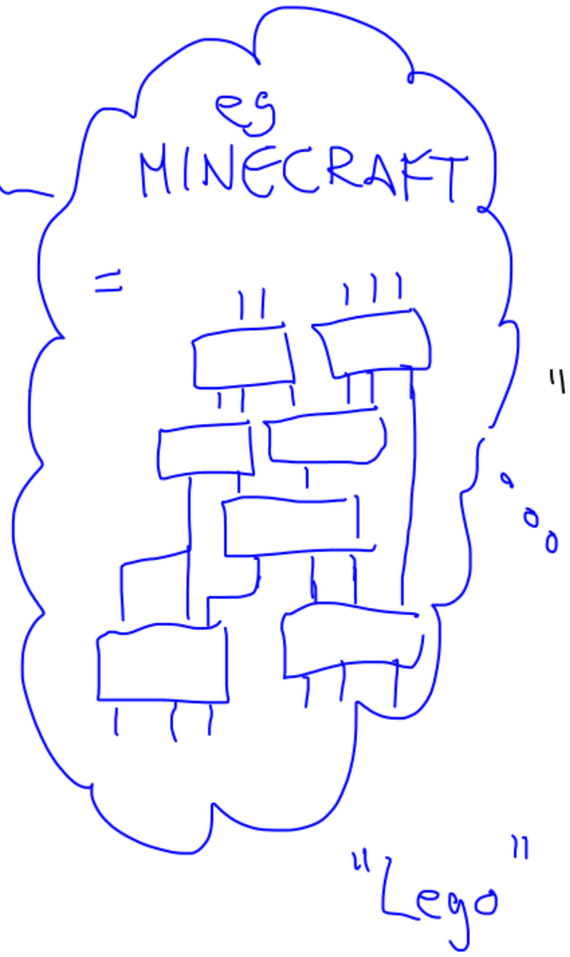
Multi - Bit Gates

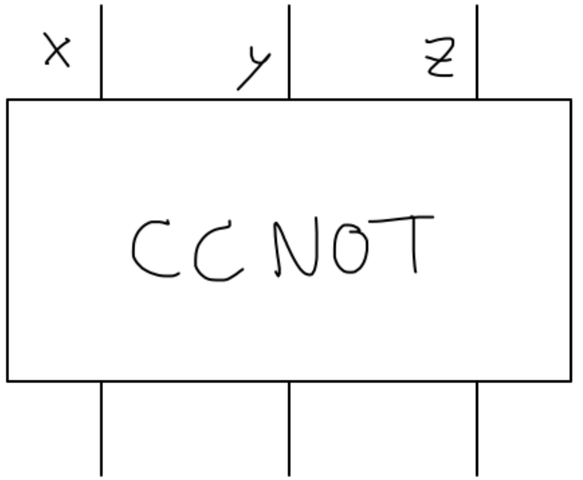
CCNOT:

"controlled-controlled not"

x	y	z	CCNOT
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

"if bitden and bathroom light on, flip bedroom light"





"UNIVERSAL":

ANY SOFTWARE CAN BE CREATED BY CCNOTS

Logic Quantum Gates

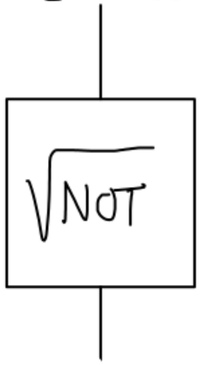
All gates are ultimately physics. Applying gates takes time.

What do you get when you stop "half way in" during NOT ("sparks!") ?

If bit is "small, isolated, cold":

BUT WHAT IS QUANTUM COMPUTING?

IMPOSSIBLE WITH NORMAL COMPUTERS !



a gate which gives you NOT when applied twice. see at MQ!

CCNOT + √NOT UNIVERSAL FOR QUANTUM COMPUTING

$NOT^2 = 1D^2 = 1D$

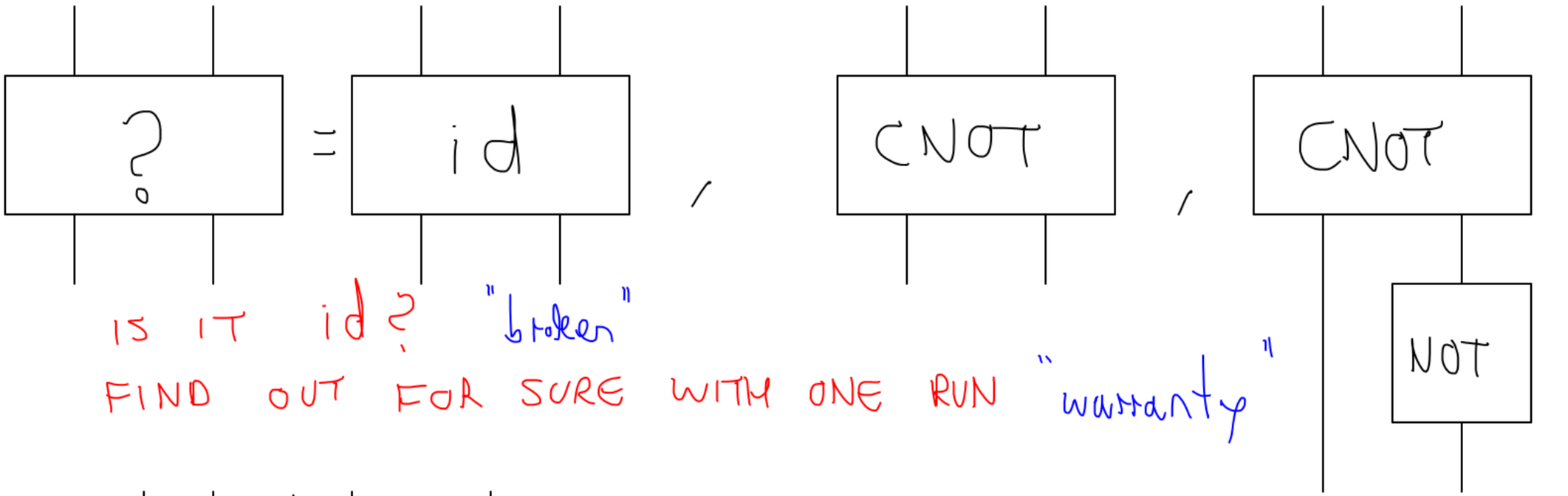
$OFF^2 = OFF$

$ON^2 = ON$

A BIT ON WHICH WE CAN APPLY √NOT IS CALLED QBIT

mystery box

The simplest Quantum Algorithm



IS IT id ? "broken"

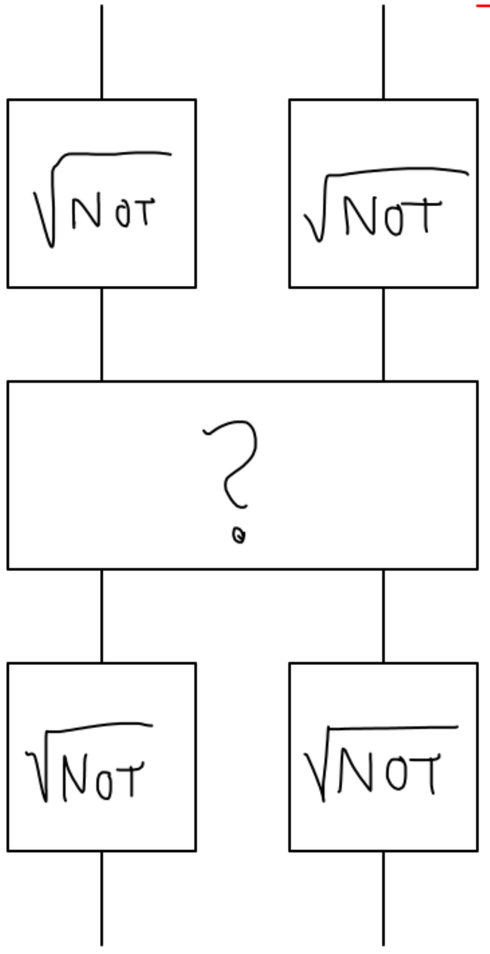
FIND OUT FOR SURE WITH ONE RUN "warranty"

x	y	id		CNOT		CNOT NOT	
0	0	0	0	0	0	0	1
0	1	0	1	0	1	0	0
1	0	1	0	1	1	1	0
1	1	1	1	1	0	1	1

EACH INPUT HAS AMBIGUITIES

IMPOSSIBLE WITH NORMAL COMPUTER

A quantum solution



INPUT : 0 1

? = id : 1 0 "broken"

? = CNOT : 0 0 } "not broken"

? = CNOT NOT : 0 0 } "not broken"

⇒ solved !

Quantum algorithms are faster at distinguishing certain things

: :

THIS IDEA GENERALIZES TO EG SHOR

ONLY THE DETAILS...

How does it work? (Heisenberg)

Idea: describe gates as arrays and bits as lists

$$0 \Rightarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad 1 \Rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\text{NOT} \Rightarrow \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

array multiplication

$$\text{NOT}(0) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 1$$

EQUIVALENT! BUT MORE MATH FREEDOM:

CAN TAKE $\sqrt{\ }$ OF ARRAY ("experiment")

2 bits: CNOT $\Rightarrow$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

NOW COMPUTE ARRAY SQUARE ROOT AND MULTIPLY OUT

$$00 \Rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad 01 \Rightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad 10 \Rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad 11 \Rightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Conclusions

* A quantum computer is a normal one + $\sqrt{NOT}$
* It is better at distinguishing stuff, sometimes
* Understanding it just requires array multiplication

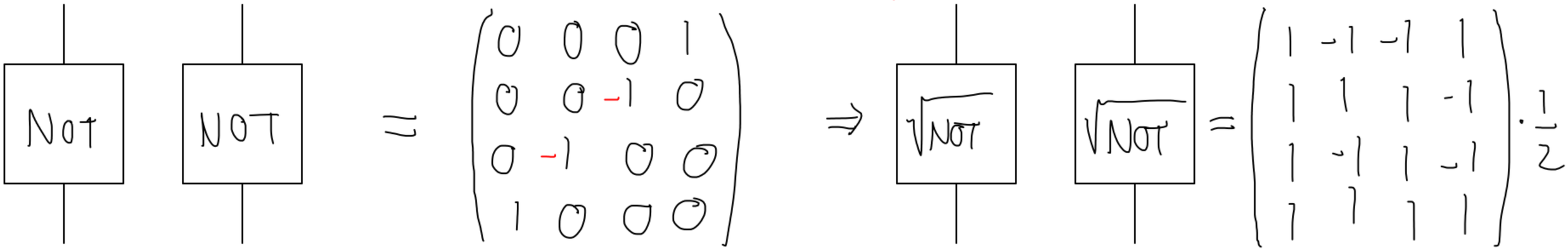
PS I cheated :

Cheating

It turns out that signs are important. We interpret eg. $\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$ as same.

Gate matrices can also have signs!

(chosen for convenience; have same effect)



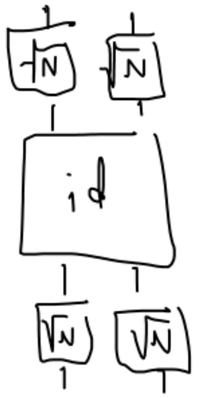
More details

$$\frac{1}{2}\begin{pmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\frac{1}{2}\begin{pmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix}\begin{pmatrix} 0 & 0 & 0 & 4 \\ 0 & 0 & -4 & 0 \\ 0 & -4 & 0 & 0 \\ 4 & 0 & 0 & 0 \end{pmatrix}\frac{1}{4}$$

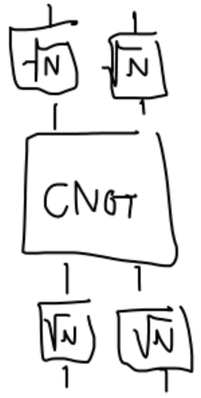
✓

⇒



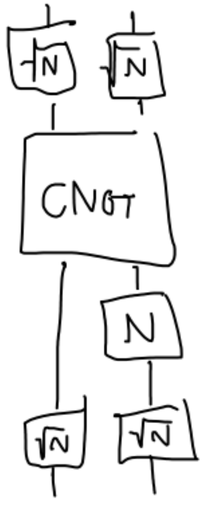
$$= \begin{pmatrix} 1 & & & \\ & -1 & & \\ & & -1 & \\ & & & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \end{pmatrix}$$

$\hat{=} 01 \qquad \hat{=} 10$



$$= \begin{pmatrix} -1 & & & \\ & -1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$\hat{=} 00$



$$= \begin{pmatrix} 1 & & & \\ & -1 & & \\ & & -1 & \\ & & & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$\hat{=} 00$