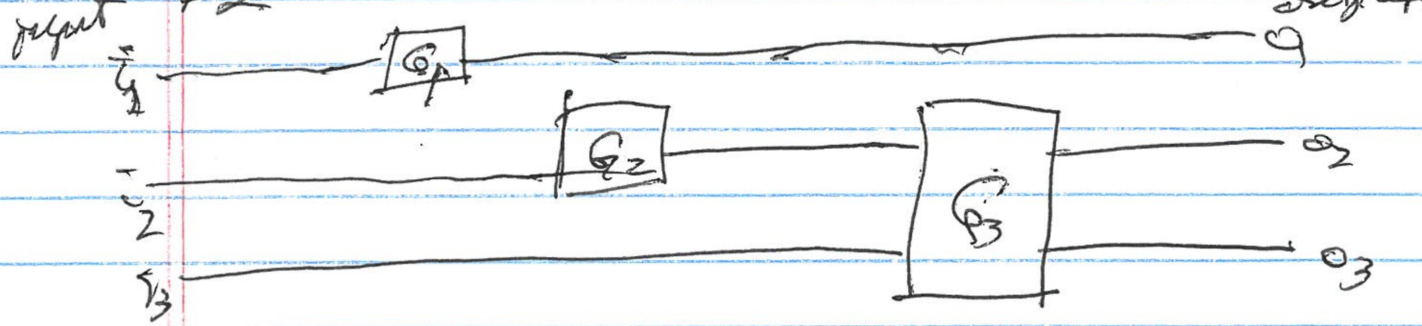
PGM: last time we covered truth tables; (how to [illegible]) & assume [illegible] to [illegible]

Circuit model of (classical) computation

circuit: an array (or network) of ~~gates~~ gates

restrict to acyclic (no loops), reversible circuits

Fig.



each line is a 'wire', each box represents a logic gate
time t proceeds from left to right

circuit complexity may be measured by the # of gates, width ('space') — that is the # of wires, and depth, being the # of time-slices
(above has depth 3, # gates 3, and width 3)

For deterministic circuits, each bit is just (in state) 0 or 1.

For a probabilistic circuit we assign probs. p_0 to be in state 0, p_1 to be in state 1.

So a single bit may be associated with a 2D vector $\begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$

operations (gates) on ~~state~~ vectors may be written as matrices.

E.g. logical NOT gate

we desire the action on basis states

$$NOT\begin{pmatrix}1\\0\end{pmatrix}=\begin{pmatrix}0\\1\end{pmatrix},\qquad NOT\begin{pmatrix}0\\1\end{pmatrix}=\begin{pmatrix}1\\0\end{pmatrix}$$

Thus the matrix repr of the NOT gate operator is

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

For more general inputs (states)

← gross typo!

$$NOT\begin{pmatrix}p_0\\p_1\end{pmatrix}=\begin{pmatrix}0&1\\1&0\end{pmatrix}\begin{pmatrix}p_0\\p_1\end{pmatrix}=\begin{pmatrix}p_1\\p_0\end{pmatrix}$$

Suppose next two wires

$$\text{———}\quad\begin{pmatrix}p_0\\p_1\end{pmatrix}$$

$$\text{———}\quad\begin{pmatrix}q_0\\q_1\end{pmatrix}$$

← probs for 2nd wire

a set of basis states is

1st	2nd
0	0
0	1
1	0
1	1

The combined state of the 2 qubits may be taken as

$$\begin{bmatrix} p_0 q_0 \\ p_0 q_1 \\ p_1 q_0 \\ p_1 q_1 \end{bmatrix} = \begin{pmatrix} p_0 \\ p_1 \end{pmatrix} \otimes \begin{pmatrix} q_0 \\ q_1 \end{pmatrix};$$

a first example of a tensor product of vectors. (much more on these products is to come)

Other examples:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes \begin{pmatrix} q \\ p \end{pmatrix}$$

$$= \begin{bmatrix} aq & bq \\ ap & bp \\ cq & dq \\ cp & dp \end{bmatrix}$$

If $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, $B = \begin{pmatrix} r & s \\ t & u \end{pmatrix}$

$$A \otimes B = \begin{pmatrix} a\begin{pmatrix} r & s \\ t & u \end{pmatrix} & b\begin{pmatrix} r & s \\ t & u \end{pmatrix} \\ c\begin{pmatrix} r & s \\ t & u \end{pmatrix} & d\begin{pmatrix} r & s \\ t & u \end{pmatrix} \end{pmatrix}$$

amounts that $(A_1 \otimes A_2)(B_1 \otimes B_2) = A_1 B_1 \otimes A_2 B_2$

Back to a 2-wire circuit

Consider action of controlled-NOT (or CNOT) gate.

Has 2 inputs - the control & target bits

The target bit is flipped only if the control bit is in the state 1

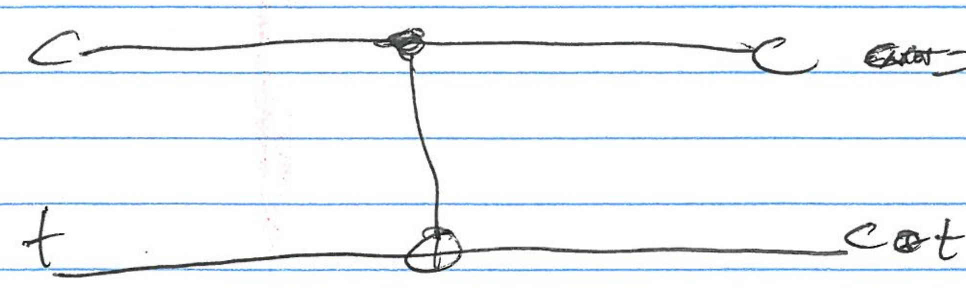
← need 2 addition, and the XOR gate

c	t	output c	output $c \oplus t$
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

(identity on first 2 states)

(swapping between the 2 states)

matrix repr. (obviously)



$$= \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

so if first bit is in state $\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$ & the 2nd bit in state $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, then $\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ 0 \\ \frac{1}{2} \\ 0 \end{pmatrix}$

& $CNOT \begin{pmatrix} \frac{1}{2} \\ 0 \\ \frac{1}{2} \\ 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

which is not a tensor product state - correlated bits in classical case - entangled in quantum setting

~~C~~NOT summary

CNOT : $(a,b) \to (a, a \oplus b)$

note if ~~the target bit~~ is 0, CNOT ~~[illegible]~~:

$(a, 0) \to (a, a)$

what's its inverse — for class to consider

(can use matrix ~~[illegible]~~ eg. & math partitions

& $\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{(ad-bc)} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

γ $\left(CNOT = \left(\begin{array}{c|c} I_2 & O_2 \\ \hline O_2 & NOT \end{array} \right) \quad \ni NOT^2 = I_2 \right)$

or else ~~[illegible]~~ 2 applications of CNOT:

$(a,b) \to (a, a\oplus b) \to (a, a \oplus (a \oplus b)) = (a, b)$

a	b	$a' = a$	$b' = a \oplus b$	$a \oplus (a \oplus b)$
0	0	0	0	0
0	1	0	1	1
1	0	1	1	0
1	1	1	0	1

i.e. $(CNOT)^2 = \not{I}$ $CNOT^{-1} = CNOT$,

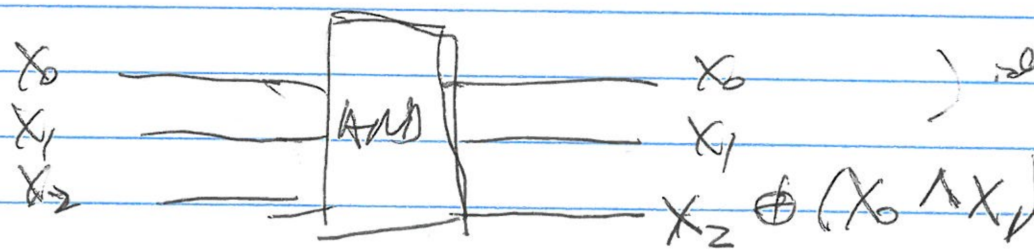
the CNOT gate is self inverse

Reversible computation - examples other than NOT & CNOT

(or write $X \wedge Y$)

is irreversible in this form

reversible version:

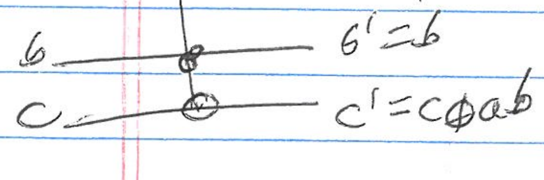


) identity on 1st 2 inputs

algebraic $X_2 \oplus (X_0 X_1)$

when (the extra bit) X_2 is set to 0, its output value gives AND

This is an instance of the 3-input, 3-output Toffoli gate (C^2-NOT gate)



a	b	c	a'	b'	c'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Identity on all the 6 first

If we put $a=1$, we get CNOT

(C^2-NOT or Toffoli gate is also self inverse)

taken from [illegible]

Supplement to in-class example

we had $a \text{ NAND } a = \text{NOT}(a \text{ AND } a)$
$= 1 - a^2 = 1 - a = \text{NOT}(a)$

now w/ truth table(s):

a	$a \wedge a$	$\text{NOT}(a \wedge a)$	$\text{NOT}(a)$
0	0	1	1
1	1	0	0

these two columns agree

~~Good morning~~ Bonjour tout le monde (everyone)

we had a long weekend, so perhaps we should review a bit first

a point concerning the circuit model
- using just 1 wire for now;

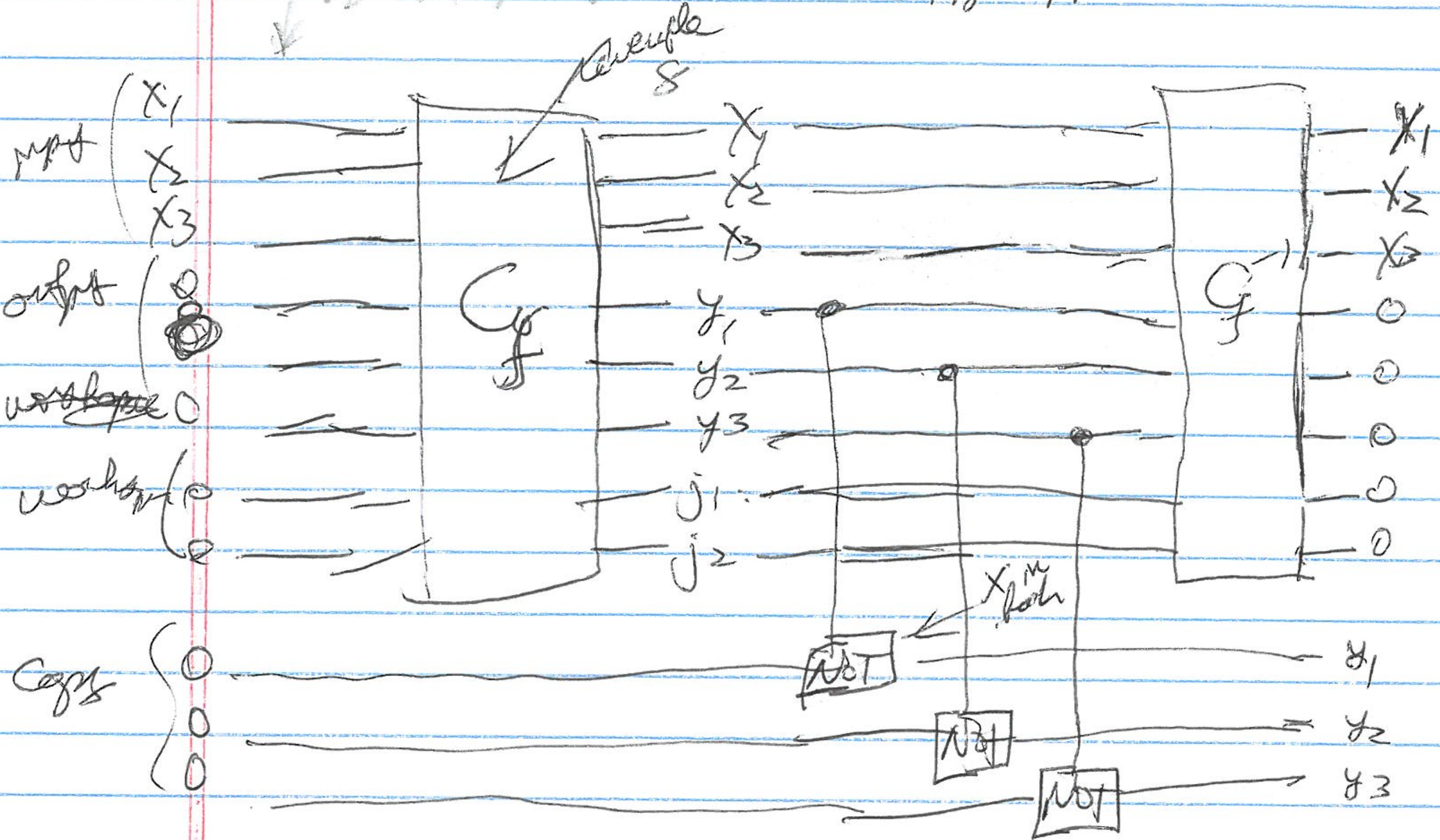
note the order of matrix operations on the input state:

$$G_3 G_2 \text{NOT} G_1 \begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$$

A reversible method for computing a fcn f

(In principle) can always find a reversible circuit for a computation (i.e. for a NAND circuit)

Fig 1.6 p[illegible]



where $f(x_1, x_2, x_3) = (y_1, y_2, y_3)$

recall CNOT gate w/ 0 as the target bit:

But so:

c	t	$c \oplus t$
0	0	0
1	0	1

← $c \oplus t$ then copies the value of the control ~~gate~~

Ex. 1.5.13 pp 14-15 Landauer principle [illegible] wasted!

8

Any irreversible fcn $f: \{0,1\}^m \to \{0,1\}^n$ can be embedded into a reversible fcn.

Define the fcn $\tilde{f}: \{0,1\}^{m+n} \to \{0,1\}^{m+n}$

step?

such that $\tilde{f}(x,y) = (x, [y + f(x)] \pmod{2^n})$

check check

$\{0,1\}^n \to \{0,1\}$

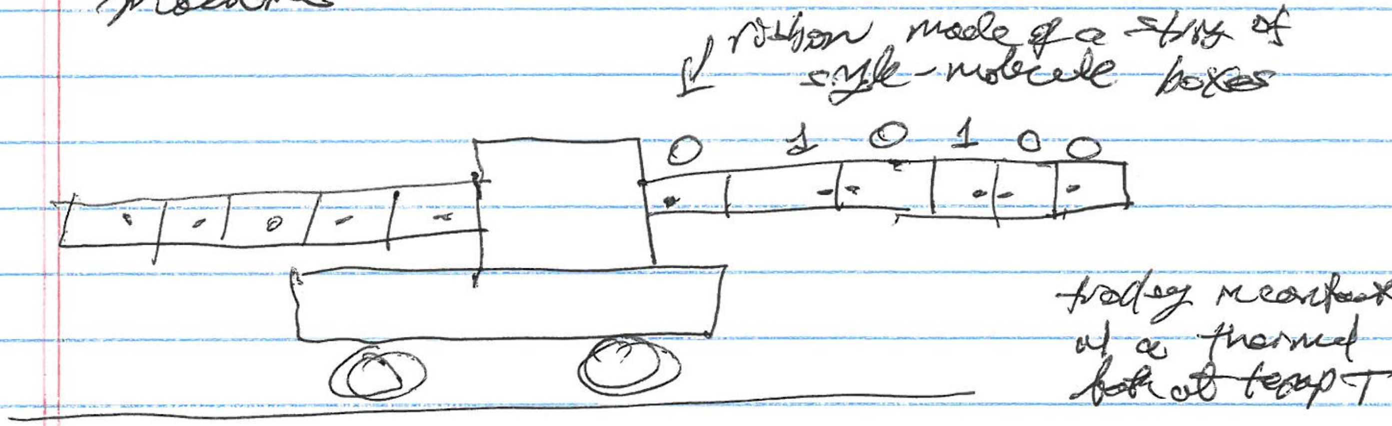
where x represents m bits, while y and $f(x)$ repr n bits. Since $\tilde{f}$ takes distinct inputs into distinct outputs, it is an invertible $(m+n)$-bit function.

(Pr. 61) (lower bd)

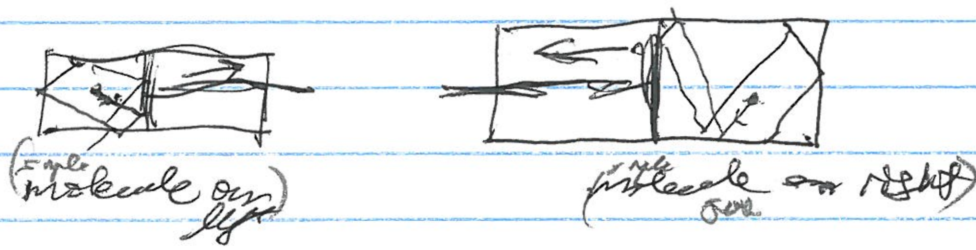
Landauer's principle. Each time a single bit of info is erased, the amount of energy dissipated into the environment is at least $k_B T \ln 2$, where $k_B \simeq 1.38 \times 10^{-16}$ erg/K $= 1.38 \times 10^{-23}$ J/K ($1J = 10^7$ erg) and T the (absolute) temp. of the (surrounding) environment. Equivalently, we say that the entropy of the environment increases by at least $k_B \ln 2$

Extracting work from information:
C. Bennett's example

showing that info may be used as fuel to move a machine



we can extract work to move the trolley by inserting a piston in the middle of each box



extracted work is $W = k_B T \ln 2$
For a N-bit ribbon, the total work is
$N k_B T \ln 2$ (and it may be used to displace the trolley). When the ribbon comes out of the trolley, the molecules can be anywhere inside the volume V.