Towards a Quantum Programming Language

Peter Selinger
(as presented by Brett Giles and Dana Harrington)
{danaha,gilesb}@cpsc.ucalgary.ca

University of Calgary

Linear Algebra Review

- Scalars: $\alpha,\ \beta,\ \lambda \in \mathbb{C}$
- Vectors: $u,\ v,\ w \in \mathbb{C}^n$ (Column Vectors)
- Matrices: $A,\ B,\ C \in \mathbb{C}^{n\times m}$
- Adjoint: $A^* = (\overline{a_{ji}})_{ij}$
- Trace: tr$(A) = \sum_i a_{ii}$
- Norm: $|A|^2 = \sum_{ij} |a_{ij}|^2$

Properties of Matrices

- A matrix $S \in \mathbb{C}^{n\times n}$is *Unitary* when $S^*S = I$. This can be used for a change of basis.
  $B = SAS^* \implies \text{tr}(B) = \text{tr}(A)$ and $|B| = |A|$
- A matrix A is *Hermitian* if $A = A^*$. Note that A is Hermitian iff $A = SDS^*$ for some unitary S and real diagonal D.
- A matrix A is *Positive Hermitian* if $u^*Au \geq 0 \; \forall u \in \mathbb{C}^n$
- We define a tensor product over complex matrices. For example:

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \otimes B = \left(\begin{array}{c|c} 0 & B \\ \hline -B & 0 \end{array}\right)$$

Hermitian Matrices

Lemma. *If* A *is Positive Hermitian, then* $|A| \leq tr(A)$

Definition. $D_n = \{A \in \mathbb{C}^{n \times n} | A$ *is Positive Hermitian and* $tr(A) \leq 1\}$.

Definition. *Define* $A \sqsubseteq B \iff A - B$ *is Positive Hermitian.*

Measurement

- One quantum bit:

$$\alpha\mathbf{0} + \beta\mathbf{1}$$

$P_0 = |\alpha|^2$, $\mathbf{0}$ → $\mathbf{0}$

$P_1 = |\beta|^2$, $\mathbf{1}$ → $\mathbf{1}$

- Two q-bits, measure FIRST one:

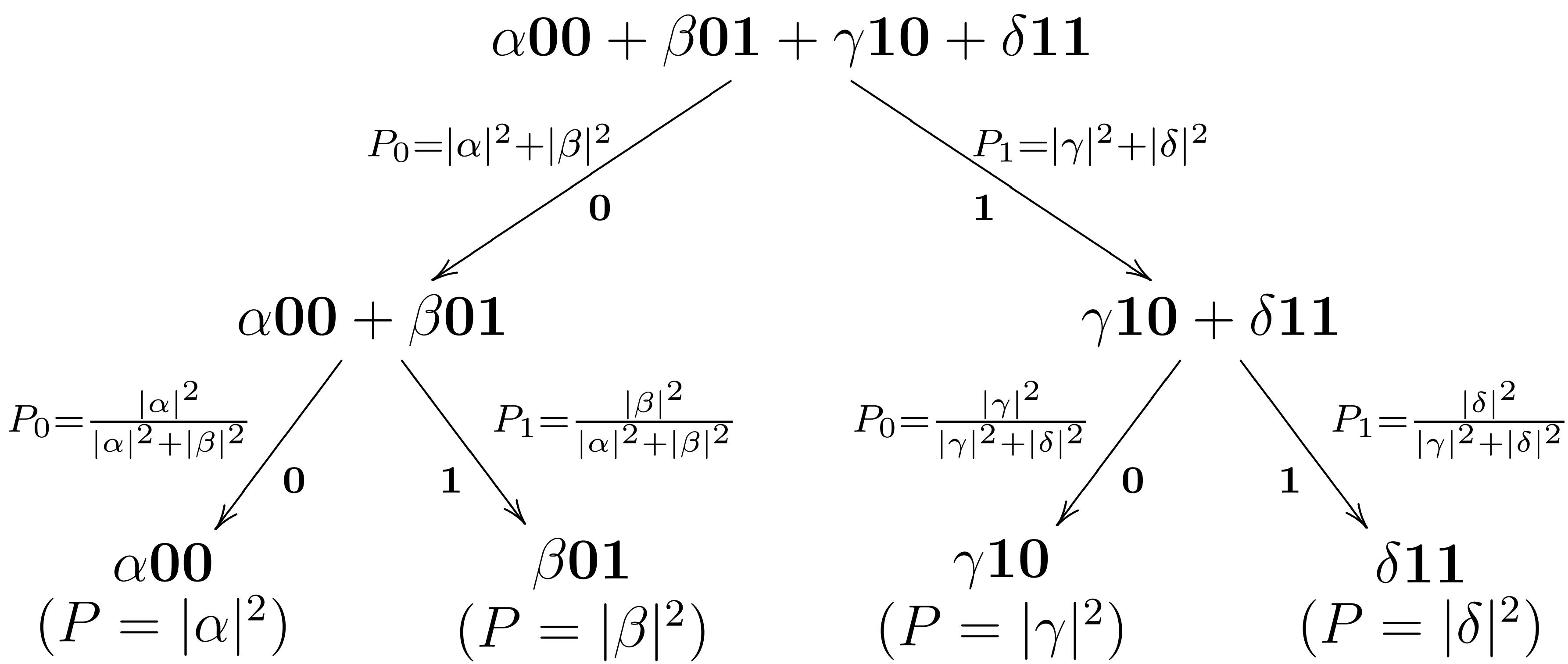
$$\alpha\mathbf{00} + \beta\mathbf{01} + \gamma\mathbf{10} + \delta\mathbf{11}$$

$P_0 = |\alpha|^2 + |\beta|^2$, $\mathbf{0}$ → $\alpha\mathbf{00} + \beta\mathbf{01}$

$P_1 = |\gamma|^2 + |\delta|^2$, $\mathbf{1}$ → $\gamma\mathbf{10} + \delta\mathbf{11}$

Measurement continued

Two q-bits, measure one, then the other:



Quantum Gates

$$N = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad N_c = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & N \end{array}\right)$$

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad H_c = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & H \end{array}\right)$$

$$V = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad V_c = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & V \end{array}\right)$$

$$W = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{i} \end{pmatrix} \qquad W_c = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & W \end{array}\right)$$

Mixed and Pure states

- *Pure state*: Quantum system is described by the state vector $u \in \mathbb{C}^{2^n}$.
- *Mixed state*: an outside observer has the viewpoint that the system is in state u_i with probability λ_i. Denoted as the mixed state:

$$\lambda_1\{u_1\} + \cdots + \lambda_m\{u_m\}, \qquad \sum_i \lambda_i = 1$$

- A unitary transformation is applied component wise to a mixed state.
- If we measure a qbit in state $\alpha\mathbf{0} + \beta\mathbf{1}$ but ignore the outcome, the system enters (from our view point) the mixed state $|\alpha|^2\{\mathbf{0}\} + |\beta|^2\{\mathbf{1}\}$

Density matrix notation

- Given a system in state u, we can represent it by the *Density Matrix* uu^*. Note that if $u = \gamma v,\ |\gamma| = 1$ we have $uu^* = \gamma\overline{\gamma}vv^* = vv^*$.
- eg. State of qbit $u = \frac{1}{\sqrt{5}}\mathbf{0} - \frac{2}{\sqrt{5}}\mathbf{1}$ is $uu^* = \begin{pmatrix} \frac{1}{5} & -\frac{2}{5} \\ -\frac{2}{5} & \frac{4}{5} \end{pmatrix}$
- A mixed state is the linear combination of the density matrices. eg., $\frac{1}{5}\{\mathbf{0}\} + \frac{4}{5}\{\mathbf{1}\}$ is

$$\frac{1}{5}\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \frac{4}{5}\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{5} & 0 \\ 0 & \frac{4}{5} \end{pmatrix}$$

Quantum operations on Density matrices - Measurement

Assume $u = \left(\frac{v}{w}\right)$, therefore $uu^* = \left(\begin{array}{c|c} vv^* & vw^* \\ \hline wv^* & ww^* \end{array}\right)$.

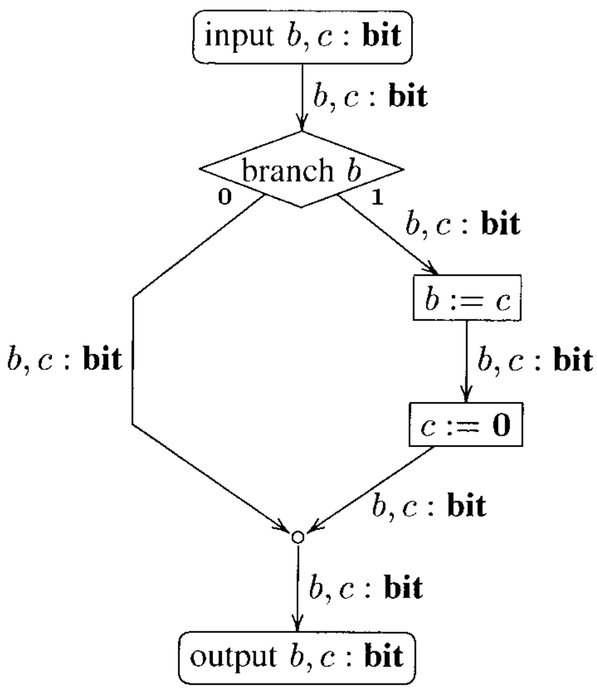
- Measuring the first qbit results in
  - $\left(\begin{array}{c|c} vv^* & 0 \\ \hline 0 & 0 \end{array}\right)$ with probability $|v|^2$.
  - $\left(\begin{array}{c|c} 0 & 0 \\ \hline 0 & ww^* \end{array}\right)$ with probability $|w|^2$.
- The probability that the matrix occurs is its trace.
- Mixed $\left(\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right) \mapsto \left(\begin{array}{c|c} A & 0 \\ \hline 0 & 0 \end{array}\right)$ or $\left(\begin{array}{c|c} 0 & 0 \\ \hline 0 & D \end{array}\right)$.

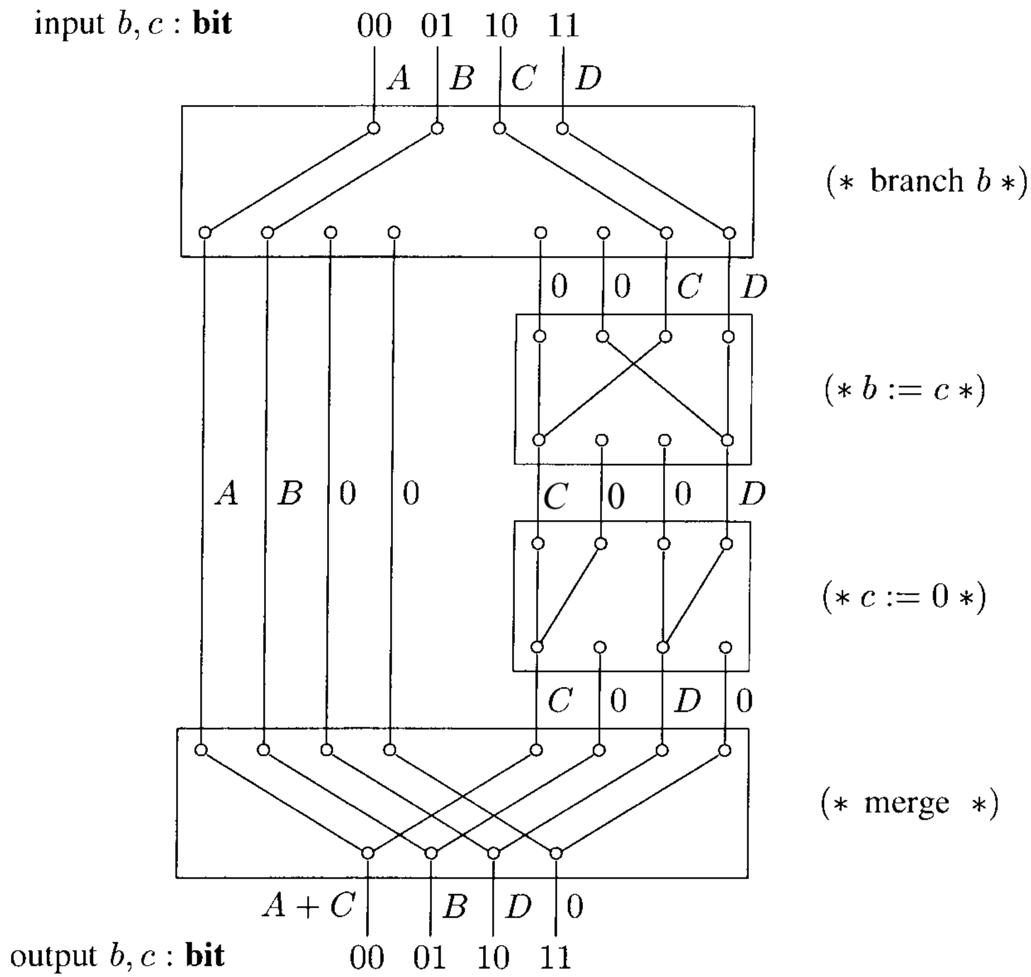
Quantum operations on Density matrices - Unitary transforms

- A transform S maps the pure state u to Su, therefore, the pure density matrix uu^* goes to Suu^*S^*.
- Extend this linearly to mixed states. A mixed density matrix A is taken to SAS^*.

As unitary transformations and measurements are our only interaction with a quantum state, there is no observable difference between two mixed states with the same density matrix.

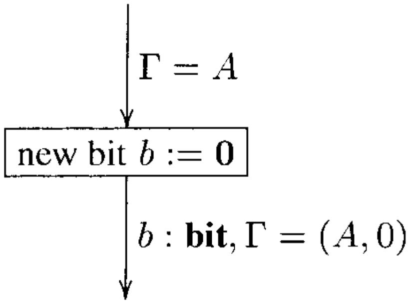
A Classical flow chart



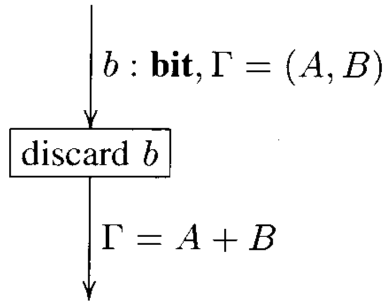


Rules for flow charts

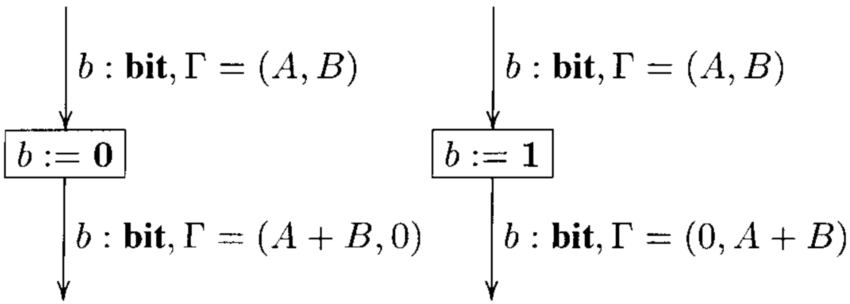
Allocate bit:



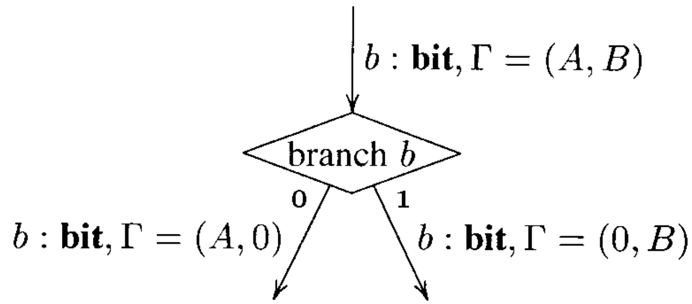
Discard bit:



Assignment:

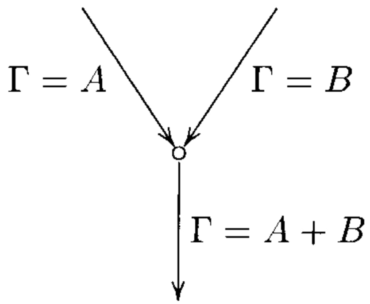


Branching:

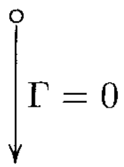


Rules for flow charts

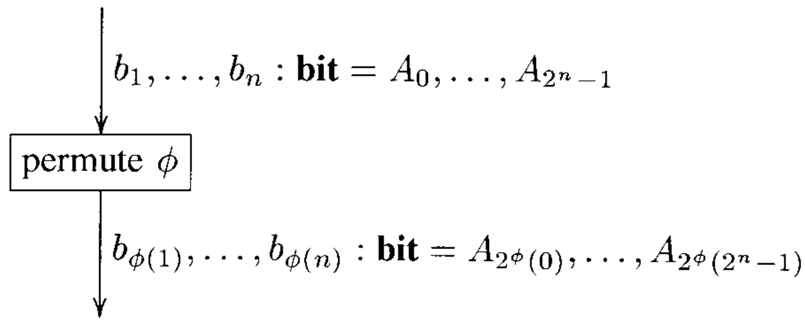
Merge:



Initial:



Permutation:



Example of permutation

$$\phi : 1 \mapsto 2,\ 2 \mapsto 3,\ 3 \mapsto 1$$

$$2^{\phi} : (x_1, x_2, x_3) \mapsto (x_3, x_1, x_2)$$

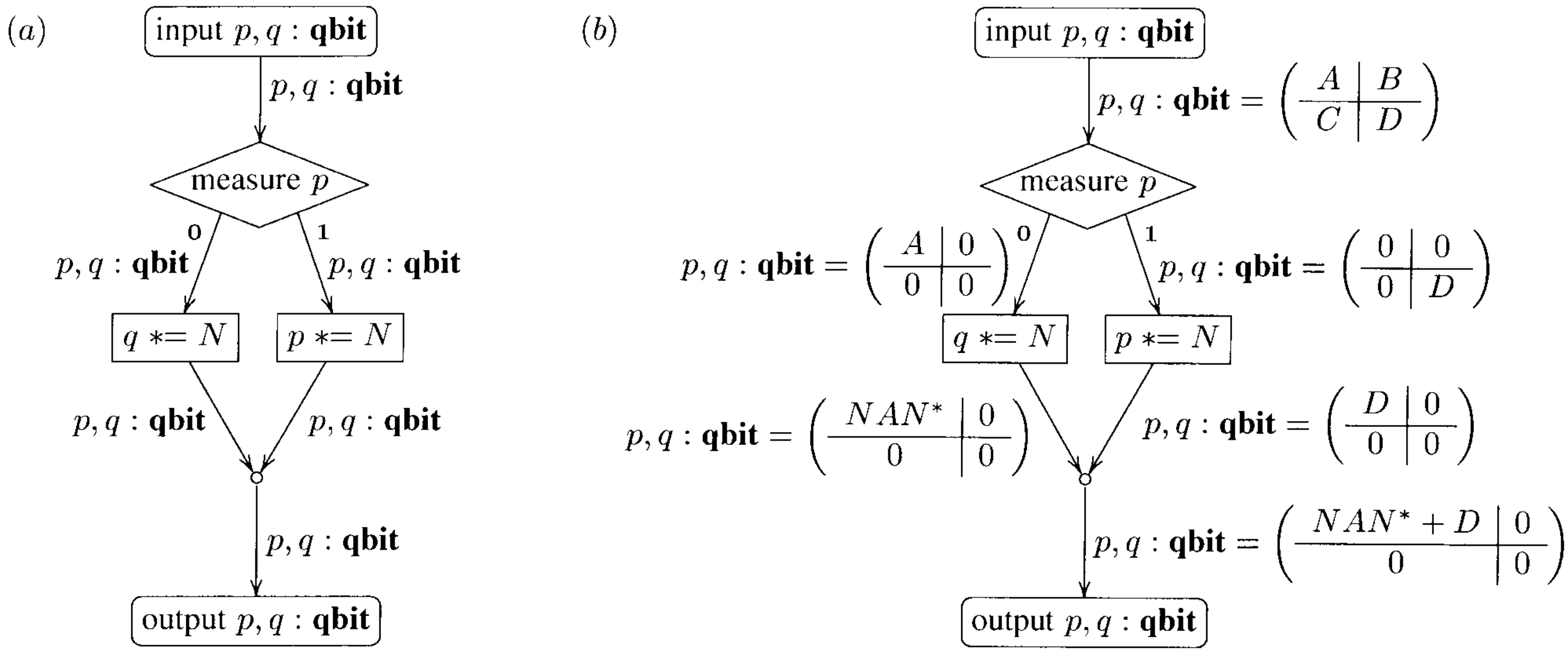
$$b_1, b_2, b_3 : \mathbf{bit} = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7)$$

$$\downarrow (\phi)$$

$$b_2 b_3 b_1 : \mathbf{bit} = (a_0, a_4, a_1, a_5, a_2, a_6, a_3, a_7)$$

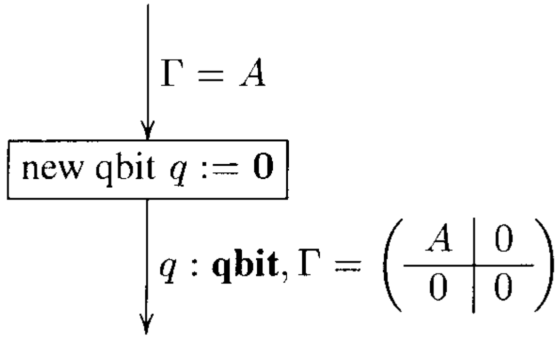
Before transform $P(011) = a_3$, transformed to 110 which still has probability a_3

A quantum flow chart

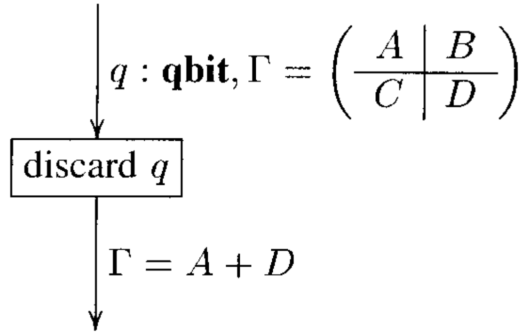


Rules for quantum flow charts

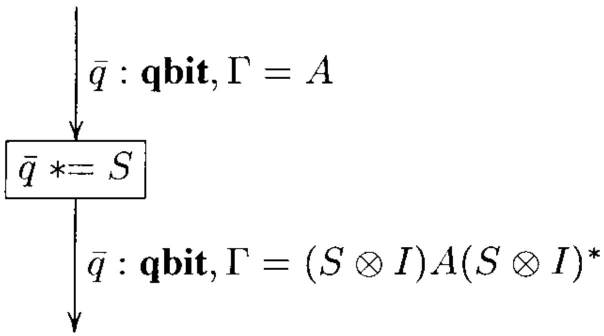
Allocate qbit:



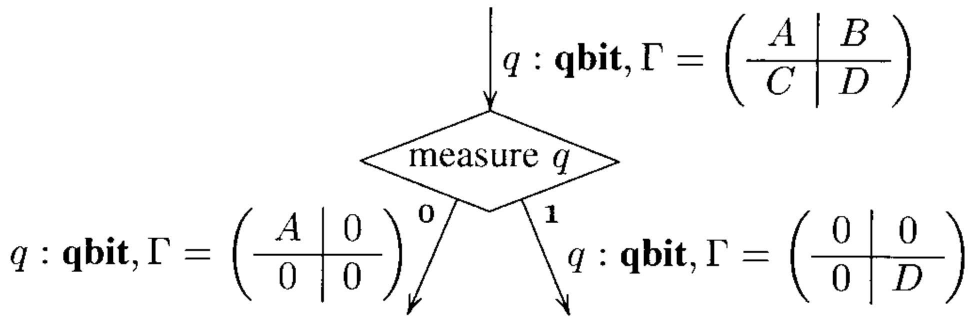
Discard qbit:



Unitary transformation:

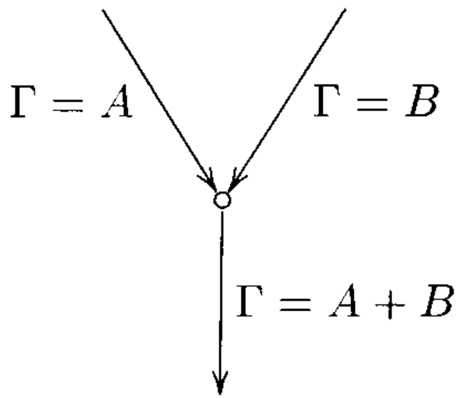


Measurement:

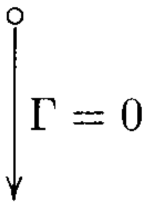


Rules for quantum flow charts

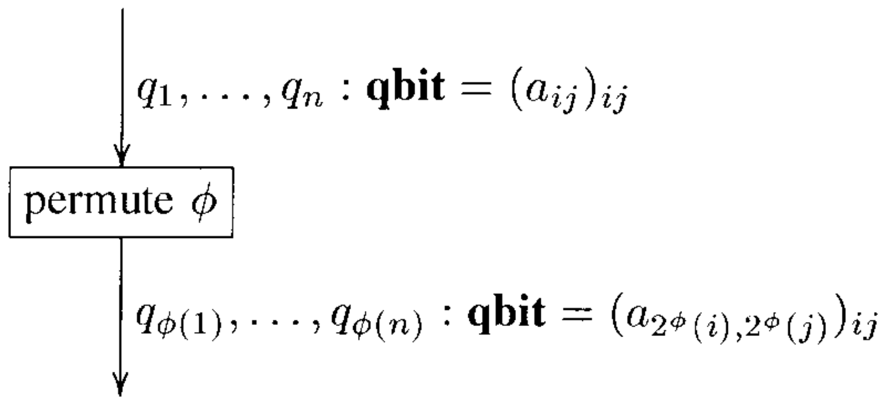
Merge:



Initial:



Permutation:



Implementation issues

(All assumptions...)

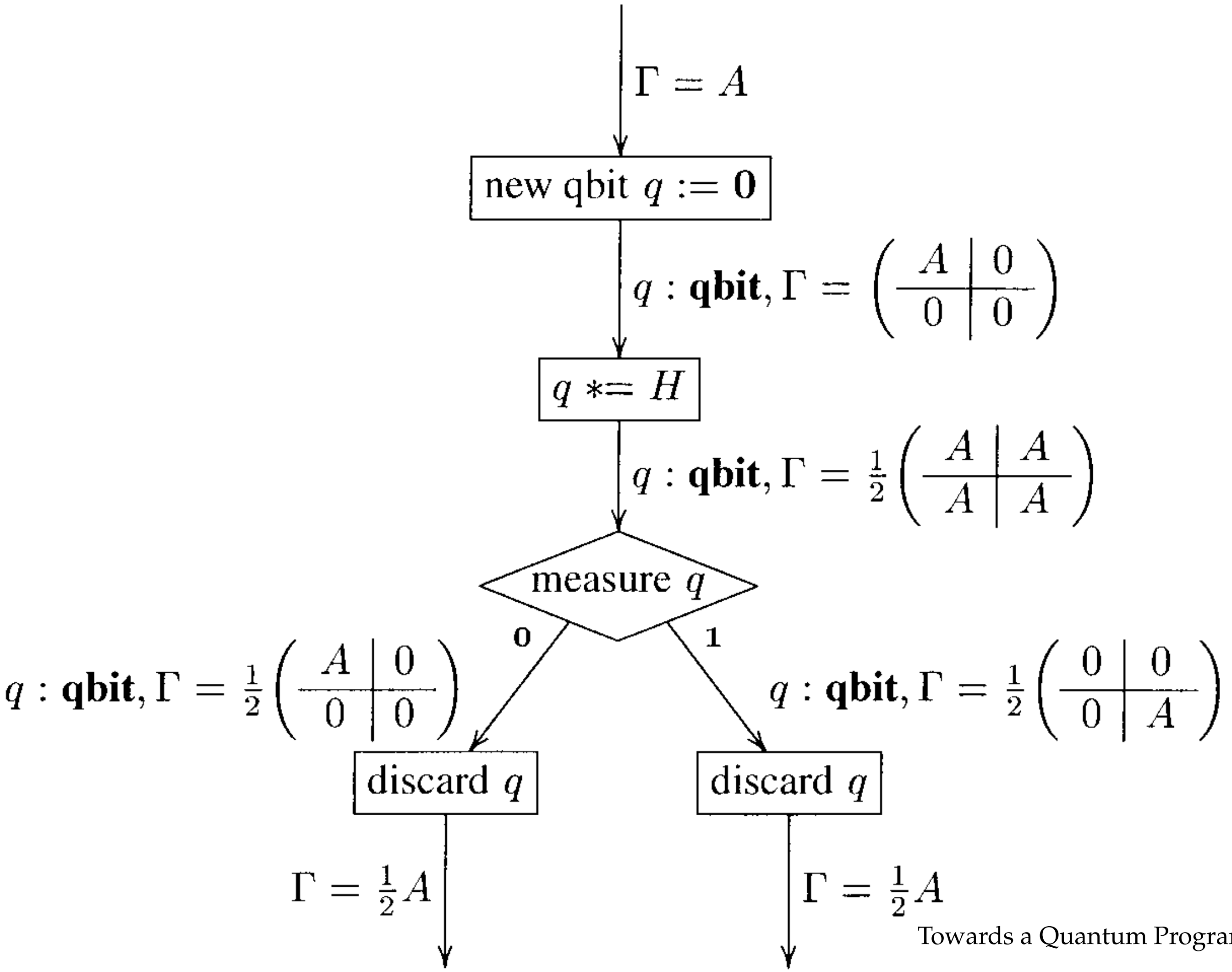
- Implement on QRAM type machine.
- OS provides basic services:
  - Allocation and deallocation of qbits.
  - Access control.
  - Actual manipulation of qbits.

Combining classical and quantum data

- Two types, **bit** and **qbit**, with typing contexts.
- Semantically, an edge labelled with n bits and m qbits can be replaced by 2^n edges each labeled with m qbits.
- The state for the above is a 2^n-tuple $(A_0, \ldots, A_{2^n-1})$ of density matrices each in $\mathbb{C}^{m \times m}$
- Extend the notions of trace, adjoints, unitary transform and norm via operation on the component and summing as needed.

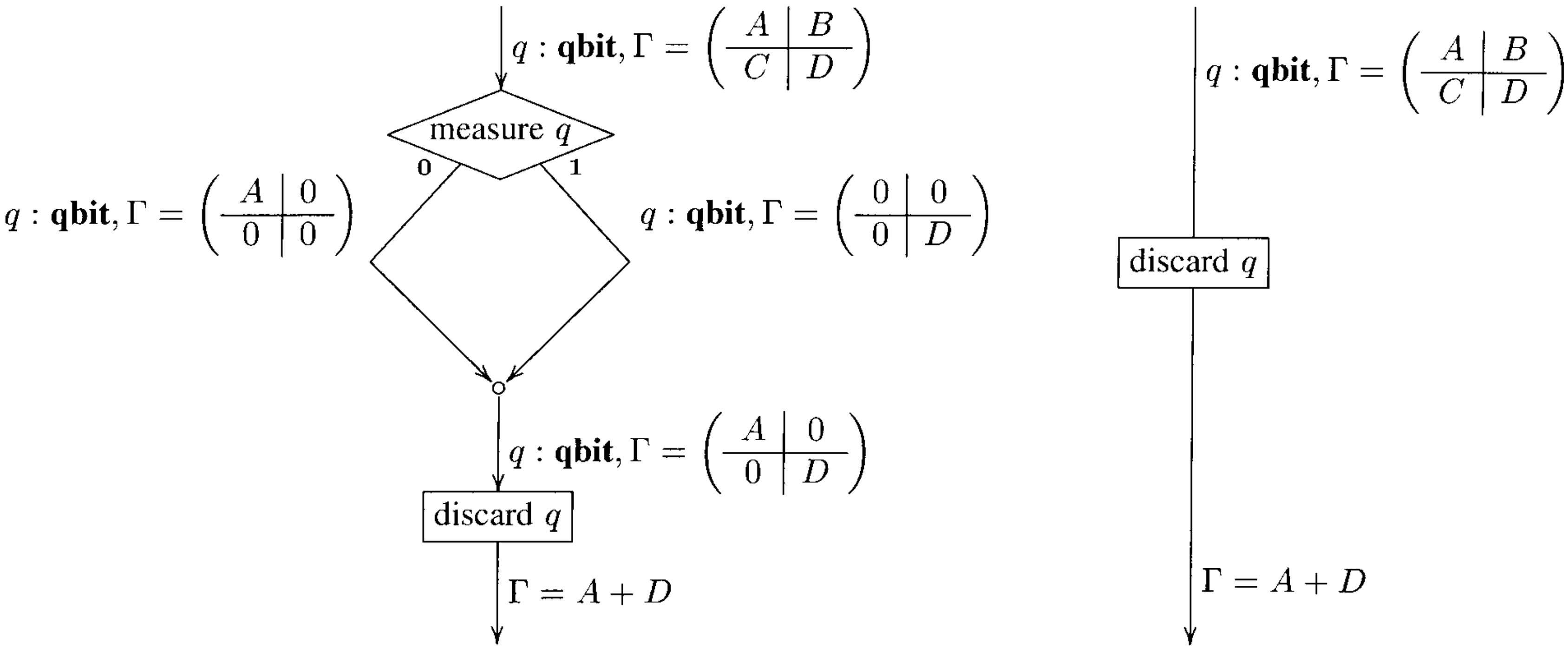
Examples of quantum flow charts

Fair Coin Toss



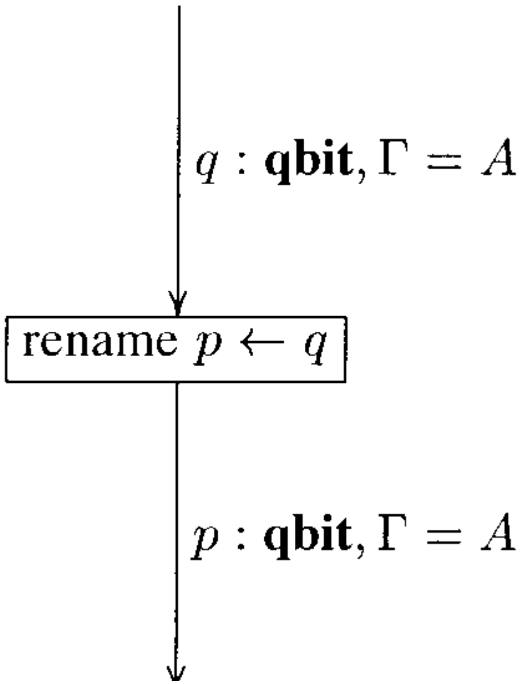
Examples of quantum flow charts

Measure ; Deallocate = Deallocate

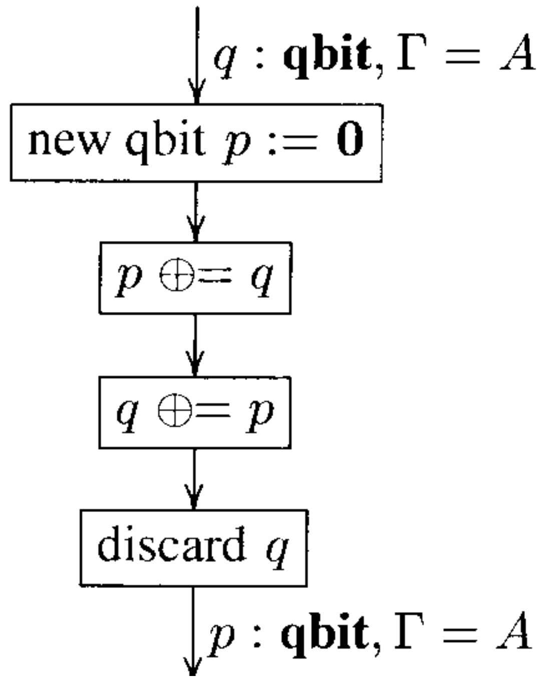


Examples of quantum flow charts

Rename of qbit

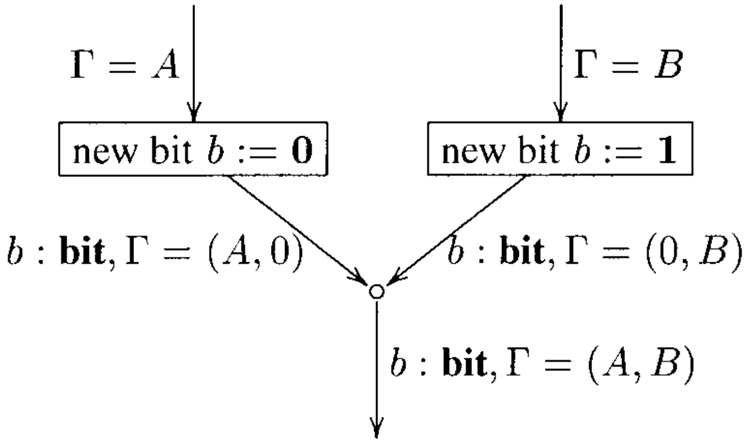


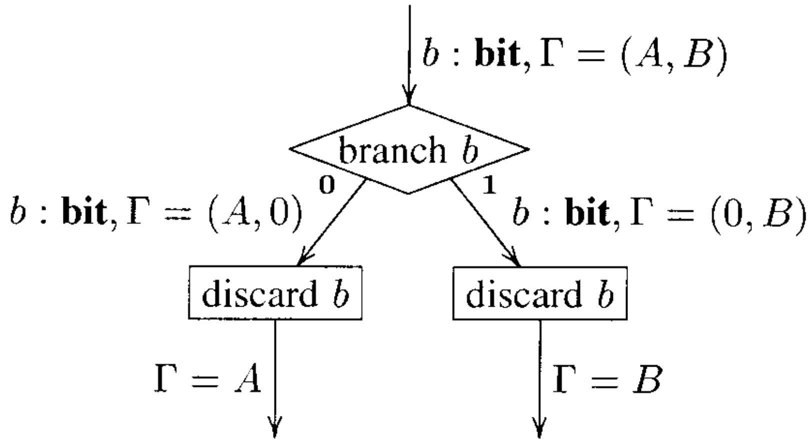
is definable as



Examples of quantum flow charts

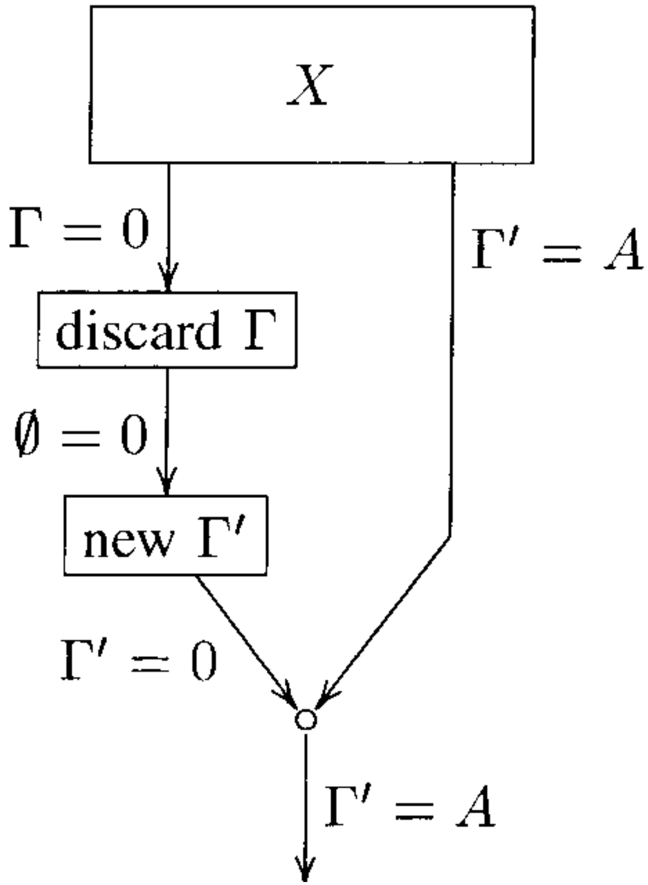
Classical Control





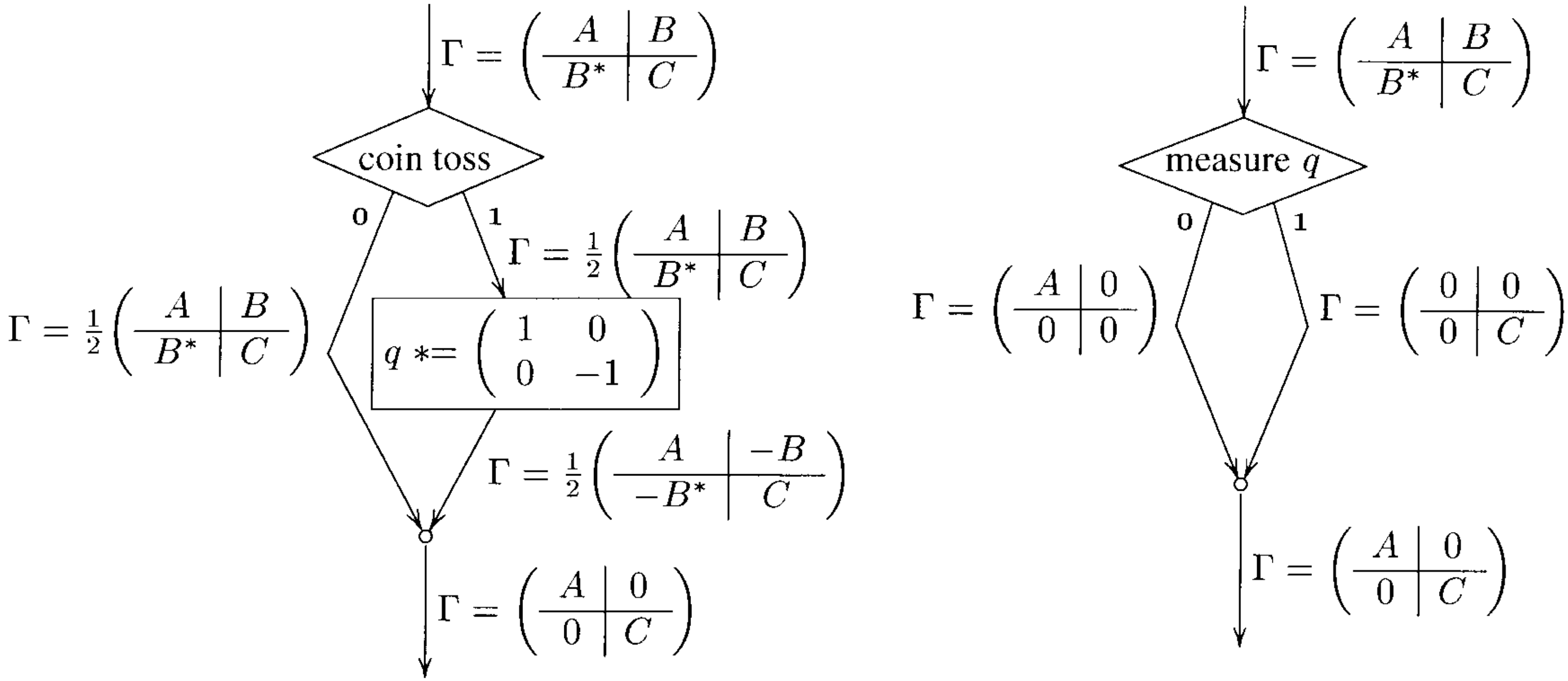
Examples of quantum flow charts

Unreachability $\Longrightarrow$ elimation of edge



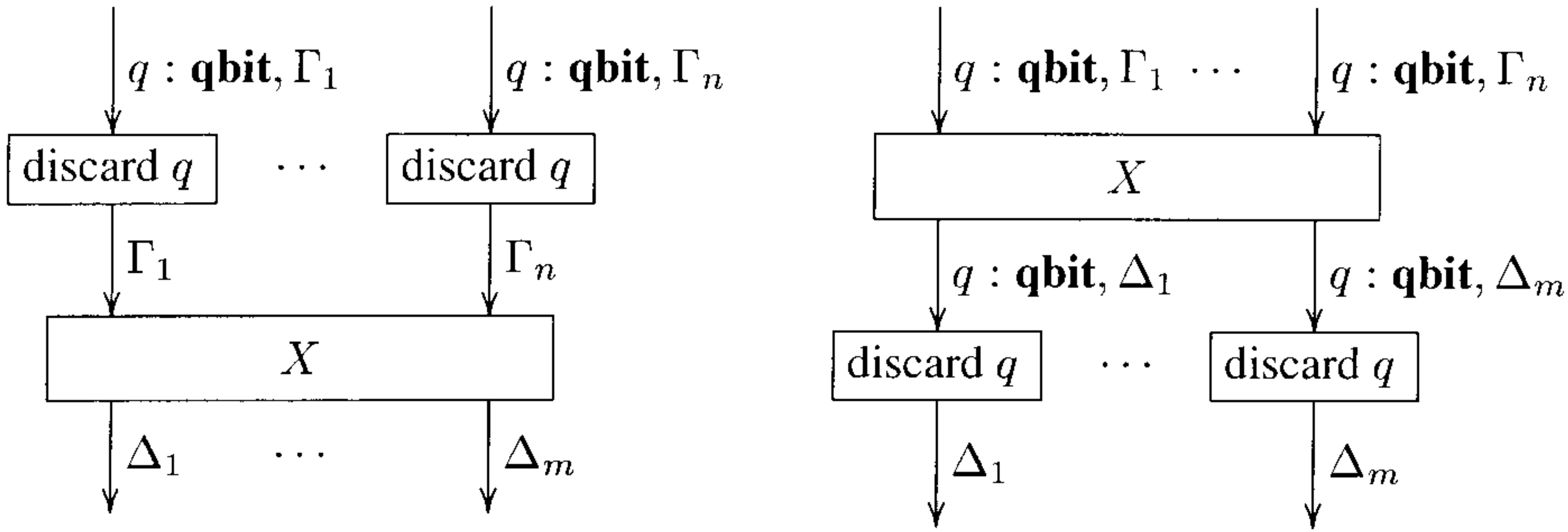
Examples of quantum flow charts

Collapse via coin toss



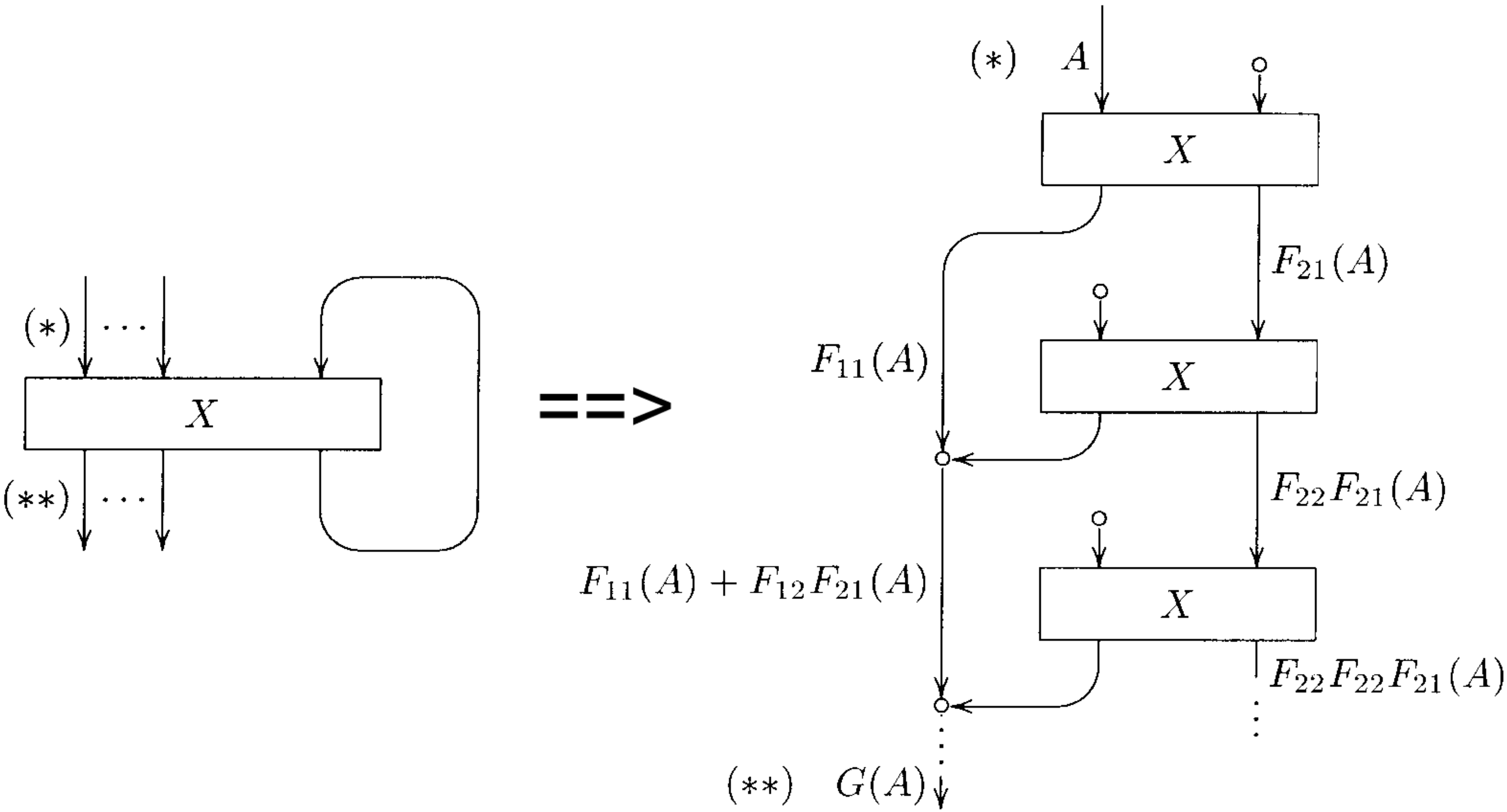
Examples of quantum flow charts

Postpone discard of qbit



Looping

Semantics of a loop = Infinite unwind



Loop semantics

- Given $A = (A_1, \ldots, A_n)$.
- Suppose semantics of X are $F(A_1, \ldots, A_n, B) = (C_1, \ldots, C_m, D)$.

Then

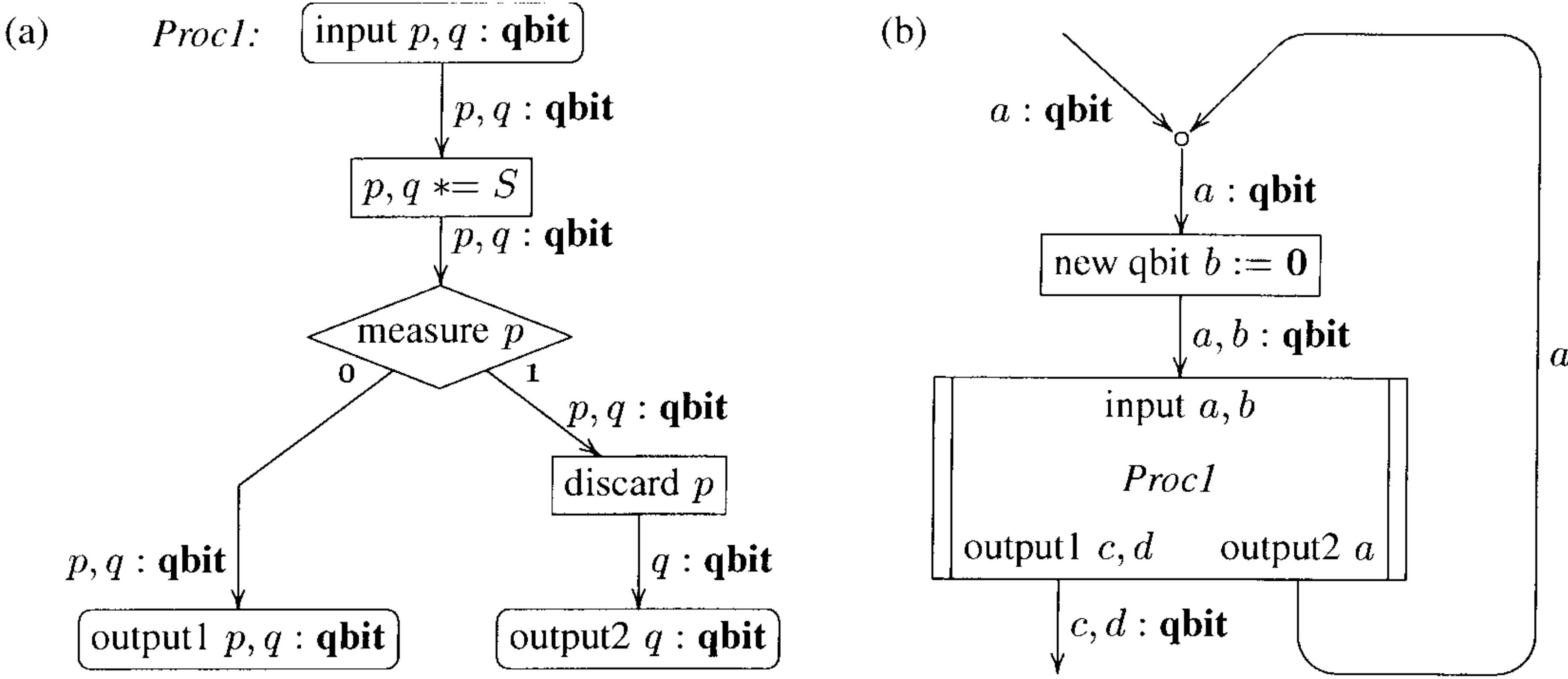
$$F(A, 0) = (F_{11}(A), F_{21}(A))$$
$$F(0, B) = (F_{12}(B), F_{22}(B))$$

and

$$G(A) = F_{11}(A) + \sum_{i=0}^{\infty} F_{12}(F_{22}^{i}(F_{21}(A)))$$

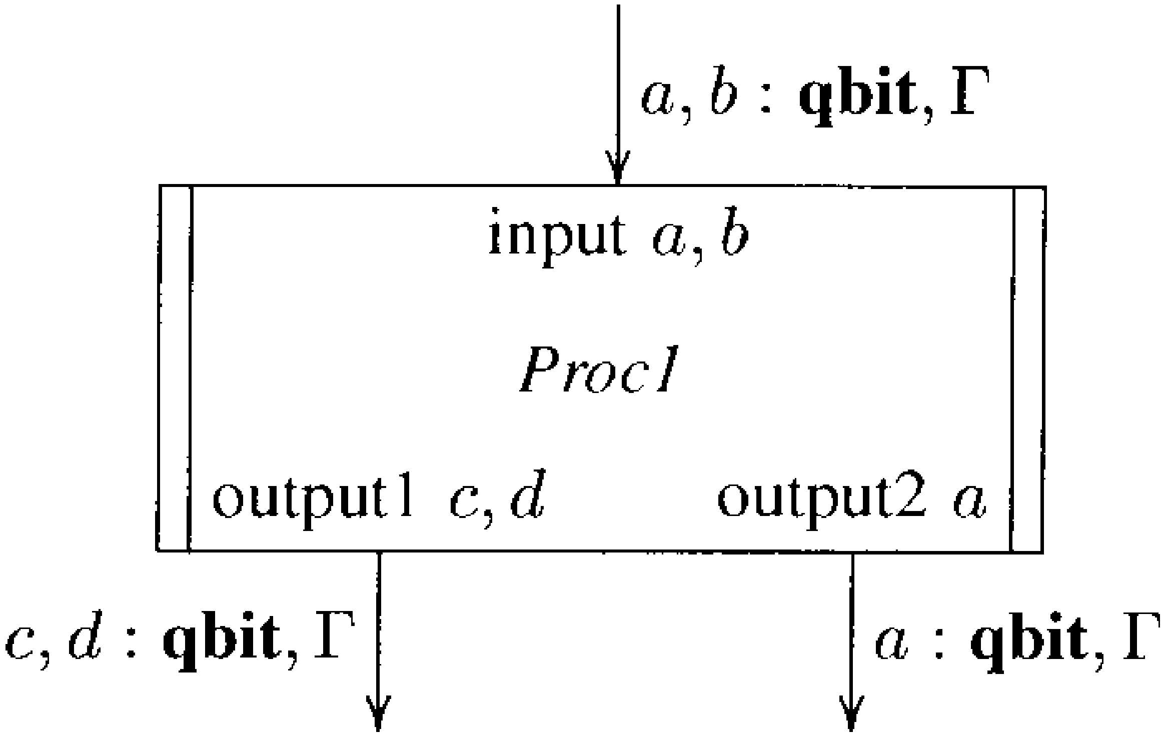
Procedures and calls (non-recursive)

Semantics = In-lining

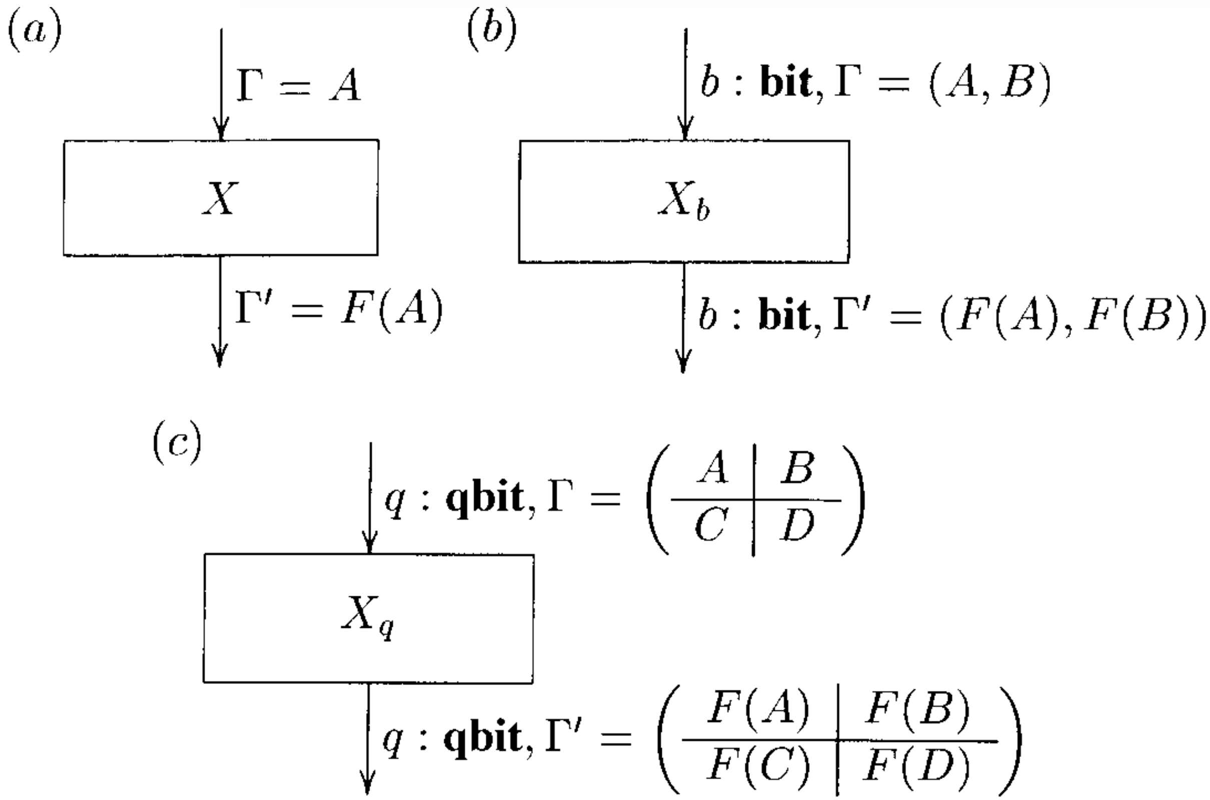


Procedures and calls

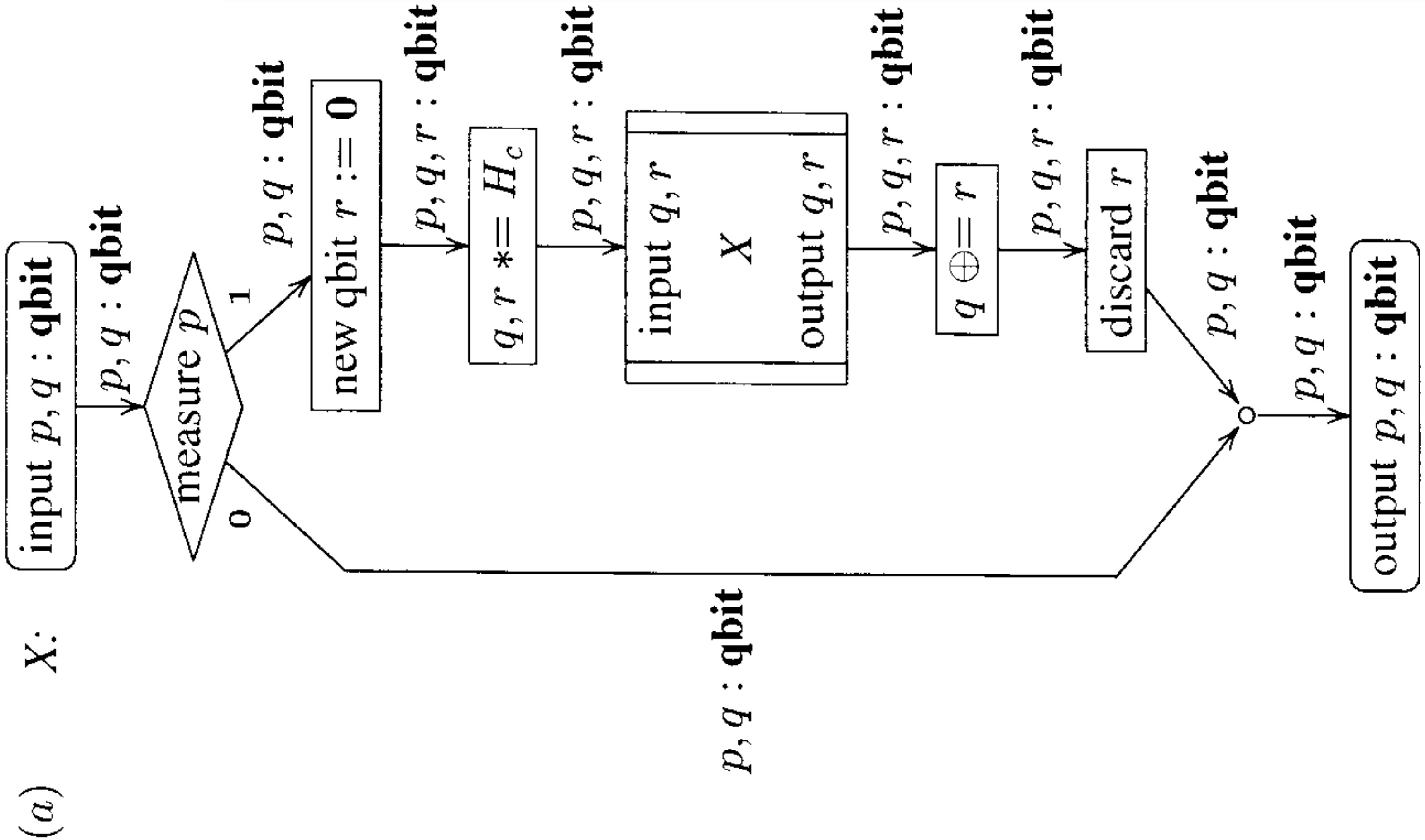
Execution in a context



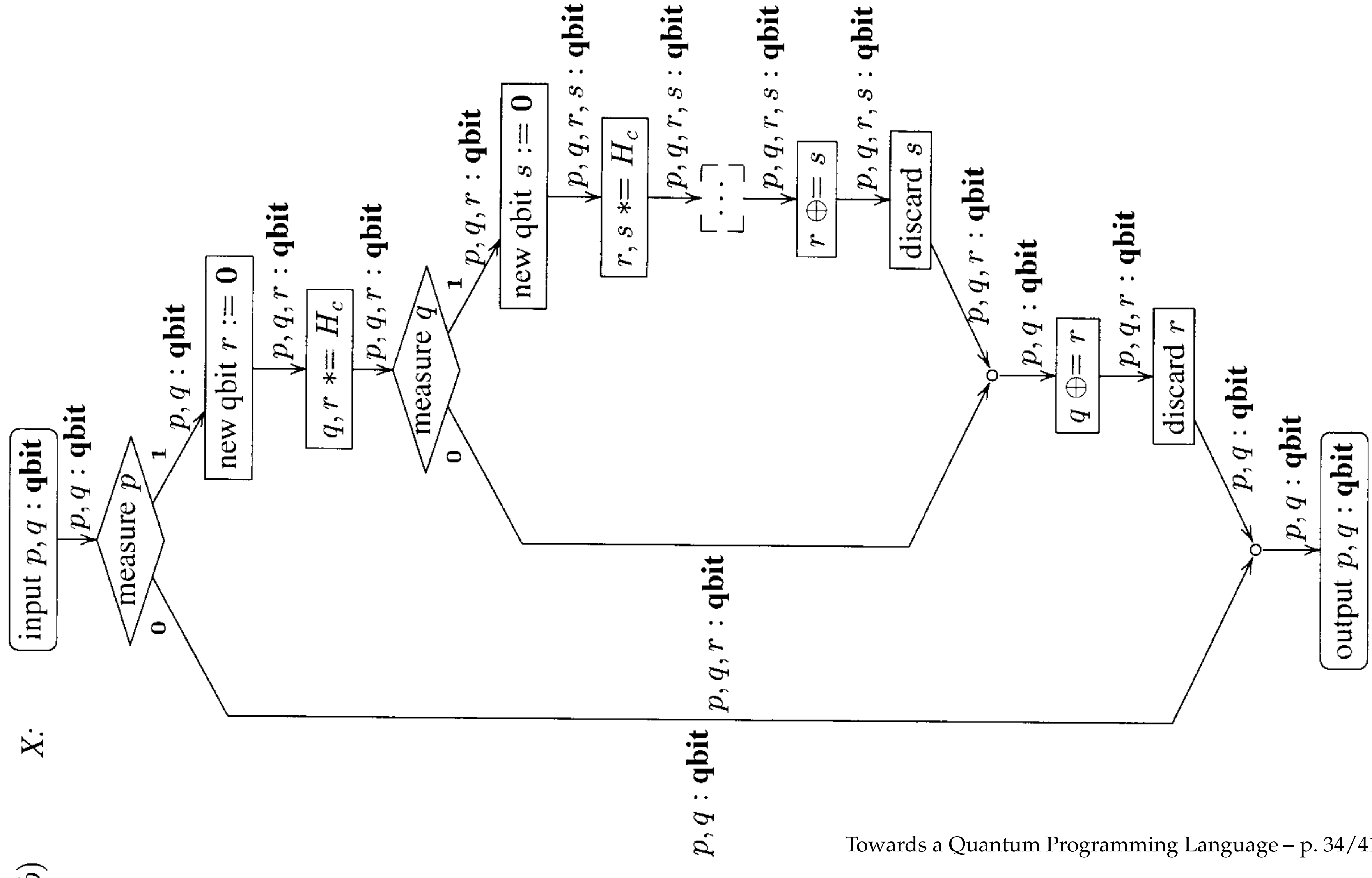
Weakening


(a)
Γ = A
X
Γ′ = F(A)
(b)
b : bit, Γ = (A, B)
X_b
b : bit, Γ′ = (F(A), F(B))
(c)
q : qbit, Γ = (A | B / C | D)
X_q
q : qbit, Γ′ = (F(A) | F(B) / F(C) | F(D))

Recursive Procedures



Recursive Procedures

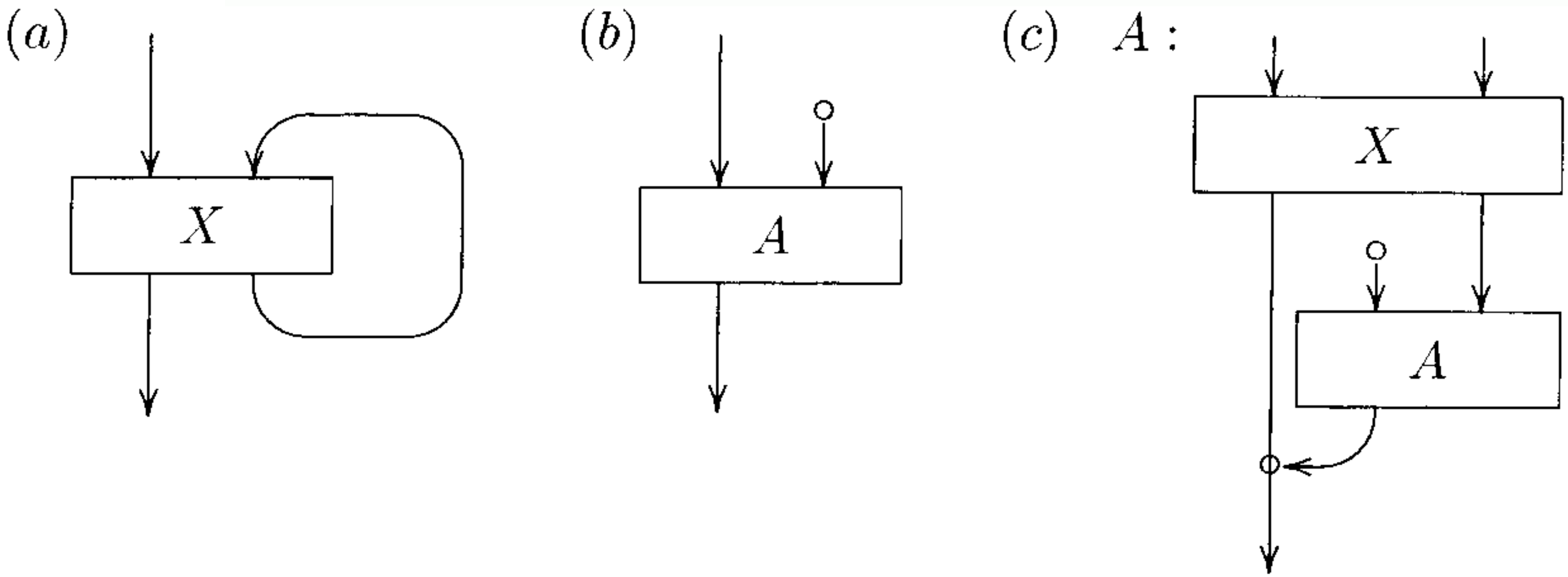


Semantics of Recursion

- $X(Y)$ is the flowchart with Y where the recursion occurred.
- Define Y_0 as a non-terminating program and then $Y_{i+1} = X(Y_i)$
- Let the semantics of $Y_i = F_i$. (Note $F_0 = 0$)
- The semantics of $X(Y)$ is a function Φ of the semantics of Y. ($F_{i+1} = \Phi(F_i)$)
- Then the semantics G of X is the limit of the F_i.

$$G = \lim_{i \to \infty} F_i.$$

Loops from Recursion



QPL

Terms $P, Q ::=$

$\textbf{new bit } b := 0 \mid \textbf{new qbit } q := 0 \mid \textbf{discard } x$

$\mid b := 0 \mid b := 1 \mid q_1, \ldots, q_n {*}= S$

$\mid \textbf{skip} \mid P; Q$

$\mid \textbf{if } b \textbf{ then } P \textbf{ else } Q \mid \textbf{measure } q \textbf{ then } P \textbf{ else } Q$

$\mid \textbf{while } b \textbf{ do } P$

$\mid \textbf{proc } X : \Gamma \to \Gamma' \ \{P\} \textbf{ in } Q \mid y_1, \ldots, y_m = X(x_1, \ldots, x_n)$

Block QPL

- Drop $\textbf{discard}\ x$.
- Add $\{P\}$ (Begin/end construction).
- Change: $\textbf{proc}\ X : \Gamma \rightarrow \Gamma\ \{P\}\ \textbf{in}\ Q$.

Extensions to type system

- Add tuples. i.e. $(x_1, \ldots, x_n)$.
- Add sums. i.e choice of n previously defined types.
- Infinite types require adaptation of the semantics.
- Structured types : add $\mathbf{case}$ construct, requires infinite types. For example, quantum list defined as: $L ::= I \oplus (\mathbf{qbit} \otimes L)$.

The Quantum Fourier Transform - rotate

```
proc rotate:
  (h:qbit,t:qbit list, n:int
          ->h:qbit,t:qbit list)
{case t of:
   nil -> {
      discard n ;
      t = nil}
   (x O* y) -> {
      x,h *= Rn ;
      n:= n+1 ;
      (h,y) = rotate (h, y, n);
      t = x O* y}
 } in...
```

The Quantum Fourier Transform - QFT

```
{proc qft:
   (l:qbit list
        ->l:qbit list) in
{case l of:
   nil -> {
       l = nil}
   (h O* t) -> {
       h *=H;
       new int n:= 2;
       (h,t) = rotate (h, t, n);
       t = qft(t);
       l= h O* t}
 }
```