2^n Uses for a Live/Dead Cat

College of Engineering

Electrical and Computer Engineering

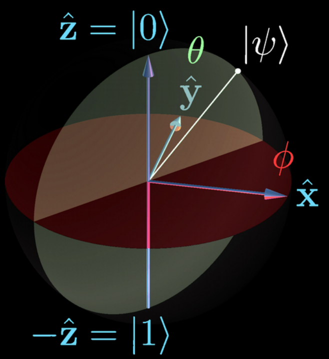
Aggregate.Org
UNBRIDLED COMPUTING

SC19
Denver, CO | hpc is now.

University of Kentucky

2^n Uses for a Live/Dead Cat

Quantum Computing



$$|\psi\rangle = \cos(\theta/2)|0\rangle + e^{i\phi}\sin(\theta/2)|1\rangle$$
$$= \cos(\theta/2)|0\rangle + (\cos\phi + i\sin\phi)\sin(\theta/2)|1\rangle$$

where $0 \leq \theta \leq \pi$ and $0 \leq \phi < 2\pi$

- Superposition: 0, 1, or 2^n probability amplitudes
- Entanglement can link *n* Qubit values together
- Qubit values can interfere

2^n Uses for a Live/Dead Cat

SEE WHAT'S *Wildly Possible*

Parallel Bit Pattern Computing

- A new model: **Pattern Bits**, not Qubits
- *n*-way entangled **`pbit`** is 2^n bits (1 of 2^{2^n} **patterns**)
- A **regular expression** encodes a **`pbit`** value
- A **conventional gate** can produce **2^n results!**

Aggregate.Org UNBRIDLED COMPUTING

SC19 Denver, CO hpc is now.

University of Kentucky

2^n Uses for a Live/Dead Cat

SEE WHAT'S *Wildly Possible*

Parallel Bit Pattern Computing

- **KREQC**: Kentucky's Rotationally Emulated Quantum Computer
- **16-`pbit`** HW executing 7 "Q"-bit adder per Cuccarro et al, **`arXiv:quant-ph/0410184v1`**

input: $A_i = a_i \quad B_i = b_i \quad Z = z \quad X = 0$

output: $A_i = a_i \quad B_i = s_i \quad Z = z \oplus s_n \quad X = 0$

circuit:

for $i = 1$ **to** $n - 1$: $B_i \oplus= A_i$

$X \oplus= A_1$

$X \oplus= A_0 B_0 \; ; \; A_1 \oplus= A_2$

$A_1 \oplus= X B_1 \; ; \; A_2 \oplus= A_3$

for $i = 2$ **to** $n - 3$:

$\quad A_i \oplus= A_{i-1} B_i \; ; \; A_{i+1} \oplus= A_{i+2}$

$A_{n-2} \oplus= A_{n-3} B_{n-2} \; ; \; Z \oplus= A_{n-1}$

$Z \oplus= A_{n-2} B_{n-1}$; **for** $i = 1$ **to** $n - 2$: Negate B_i

$B_1 \oplus= X$; **for** $i = 2$ **to** $n - 1$: $B_i \oplus= A_{i-1}$

$A_{n-2} \oplus= A_{n-3} B_{n-2}$

for $i = n - 3$ **down to** 2:

$\quad A_i \oplus= A_{i-1} B_i \; ; \; A_{i+1} \oplus= A_{i+2}$; Negate B_{i+1}

$A_1 \oplus= X B_1 \; ; \; A_2 \oplus= A_3$; Negate B_2

$X \oplus= A_0 B_0 \; ; \; A_1 \oplus= A_2$; Negate B_1

$X \oplus= A_1$

for $i = 0$ **to** $n - 1$: $B_i \oplus= A_i$

Aggregate.Org UNBRIDLED COMPUTING

SC19 Denver, CO | hpc is now.

University of Kentucky

2^n Uses for a Live/Dead Cat

Parallel Bit Pattern Computing

SEE WHAT'S *Wildly Possible*

`pint sqrt(29929)` in 310 gates:

```
int main(int argc, char **argv) {
  pbit_init();
  pint a = pint_mk(16, 29929);   // 16-pbit value 29929
  pint b = pint_h(8, 0xff);      // all 8-bit values
  pint c = pint_mul(b, b);       // square them
  pint d = pint_eq(c, a);        // where square is 29929
  pint e = pint_mul(d, b);       // make non-sqrts all 0
  pint_measure(e);               // prints 0, 173
}
```

Aggregate.Org UNBRIDLED COMPUTING

SC19 Denver, CO hpc is now.

University of Kentucky

Some Older Aggregate.Org UNBRIDLED COMPUTING Work @SC

2017: Gate-level compiler optimization to minimize power
2013: Time Domain Continuous Imaging (TDCI)
2012: KY Network Implementation Topology Tool (KNITT)
2008: MIMD On GPU (MOG)
2003: Cluster/Beowulf Design Rules tool (CDR/BDR), KASY0
2000: Flat Neighborhood Networks (FNNs), Bell Award for KLAT2
1996: SIMD Within A Register (SWAR), Video Walls (VWLib)
1994: Aggregate Function Networks (AFNs), 1st Linux PC Clusters
1992: PCCTS/Antlr compiler construction tools
1989: Barrier synchronization for SIMD on MIMD

Aggregate.Org UNBRIDLED COMPUTING

SC19 Denver, CO | hpc is now.

University of Kentucky