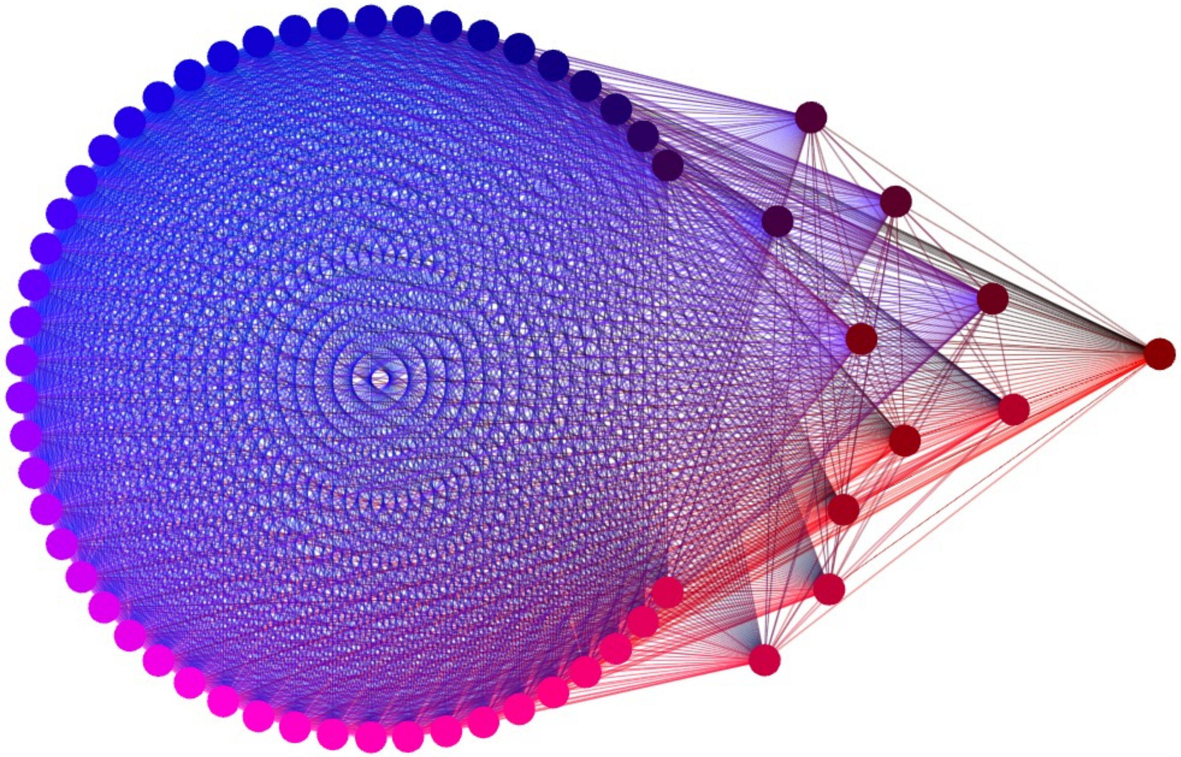
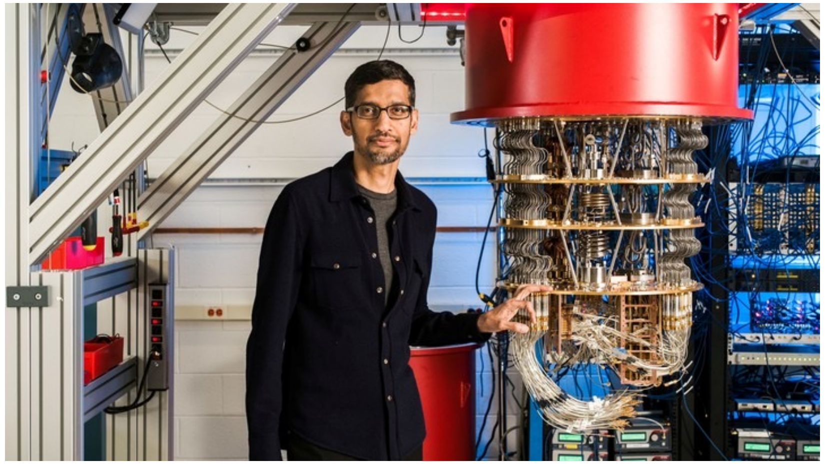
Quantum-inspired Ising Chips for Solving Hard Optimization Problems

Chris Kim

University of Minnesota
chriskim@umn.edu, chriskim.umn.edu



Quantum Computing's Hardware Problem



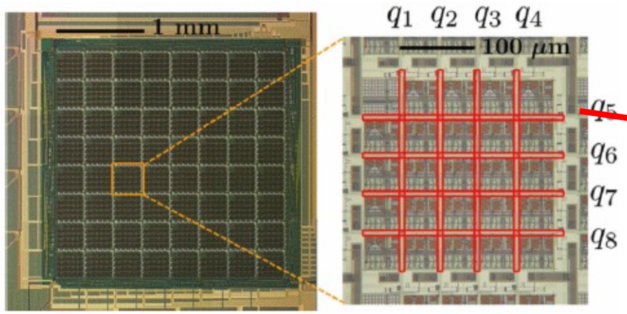
Private investments:

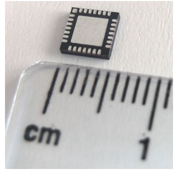
Psiquantum > $100's M (fabrication: GF)
Dwave > $100's M (fabrication: Skywater)
IonQ > $100's M (fabrication: NIST)
Rigetti > $100's M (fabrication: in-house)
IBM > $100's M (fabrication: in-house)
Toshiba > $100 M (est.)
Hitachi > $100 M (est.)
Google, Microsoft, Intel, NASA, Lockheed, …

Key challenges:

Kilowatts of cooling power
Radiation, vibration, noise shielding
Laser beams, microwave circuits, wiring
Premature device technology
Large form factor
Limited scalability

Quantum vs. Quantum-Inspired

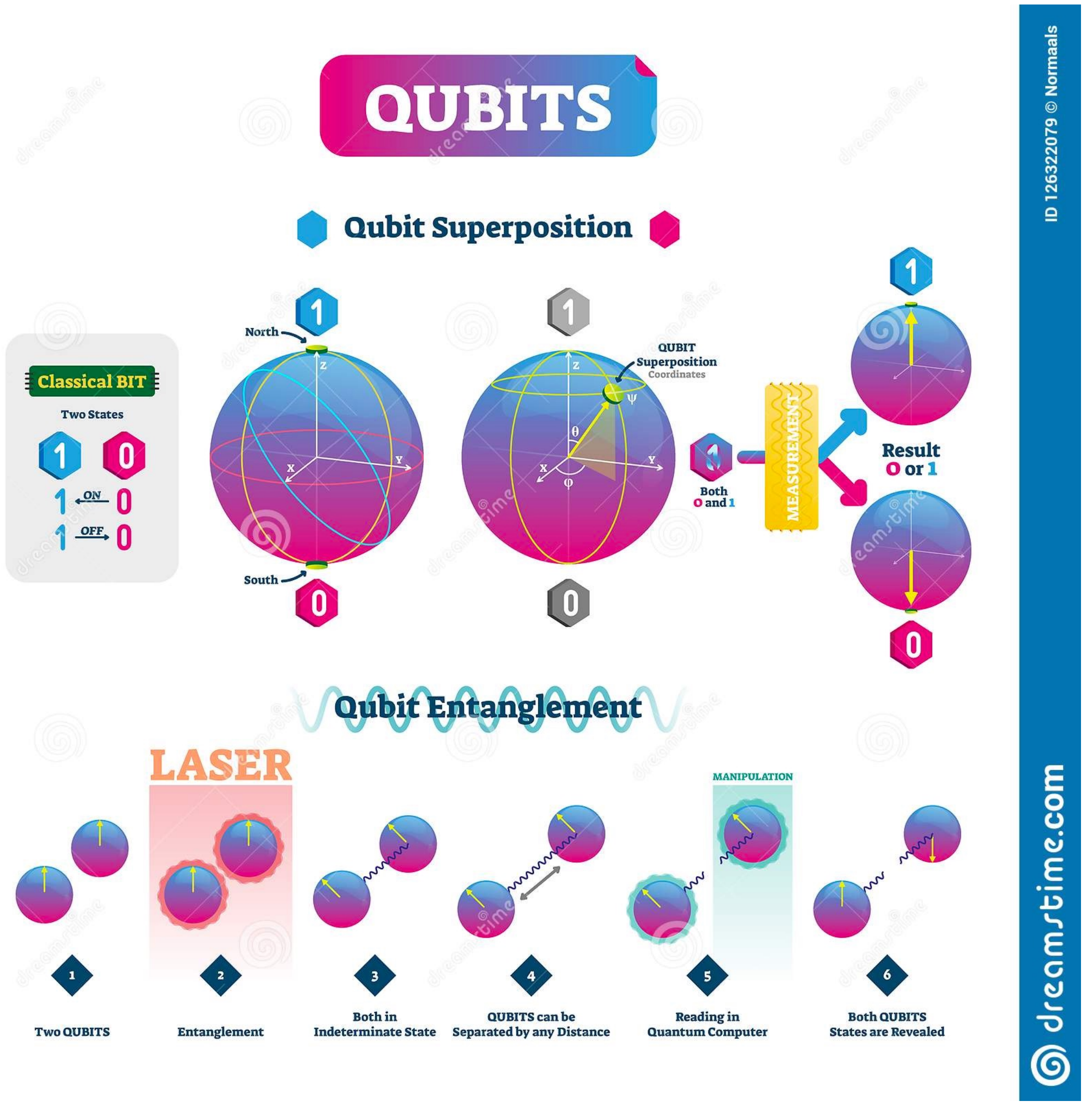


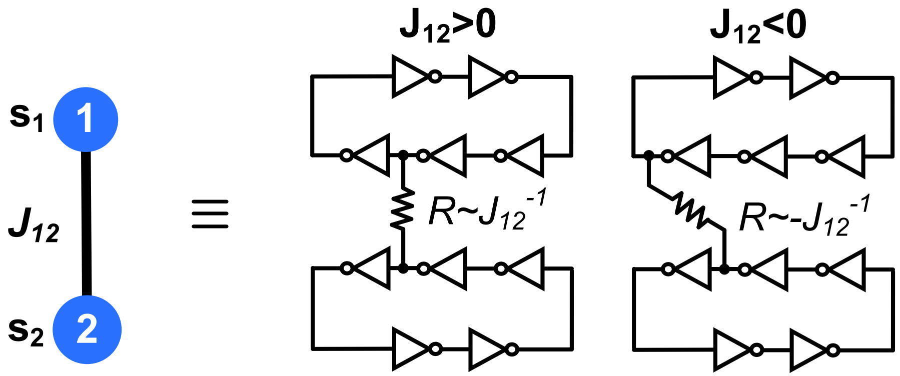


QFN package

	Dwave Quantum Annealer	Latest COBI chip
Qubits	5,000+ physical qubits = 47 (all-to-all)	60 physical qubits = 59 (all-to-all)
Connectivity	15	58
Couplers	35,000+	60x59/2 = 1,770
Weight resolution	Unknown	~5 bits
Power consumption	25 kW	0.010W @ 1.2V
Technology	Superconducting	TSMC 65nm CMOS
Price	$15M	$60 per die (MPW run)

Quantum vs. Quantum-Inspired

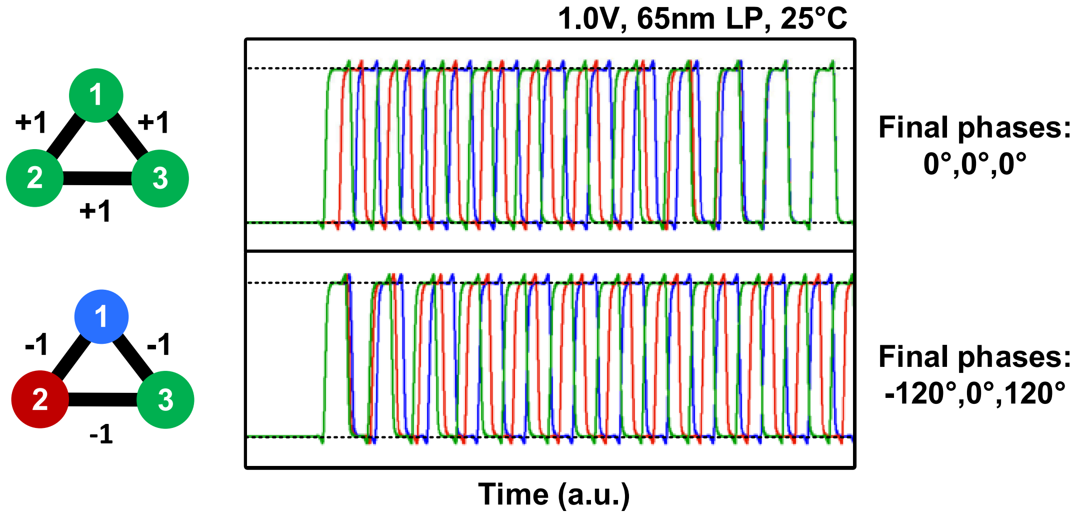




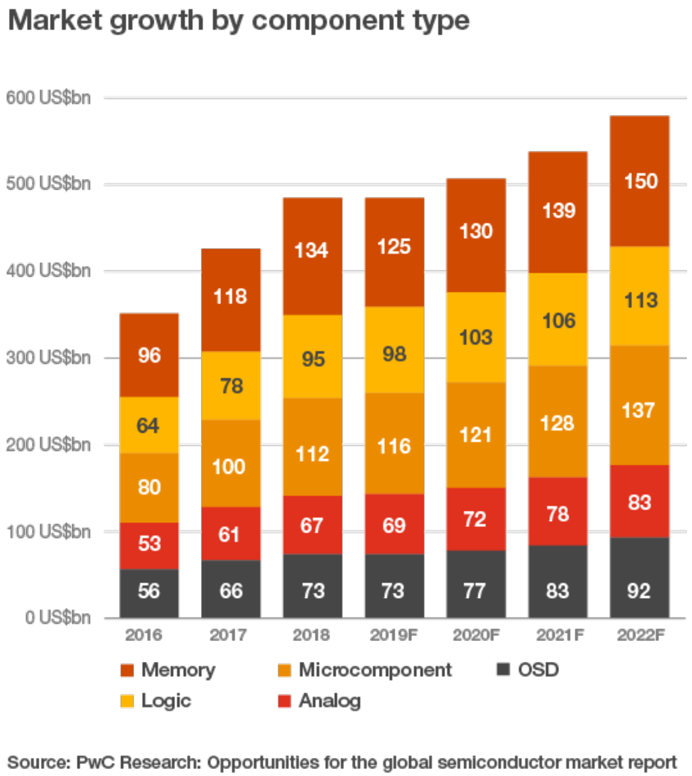
$$H(s) = -J_{ij} s_i s_j$$

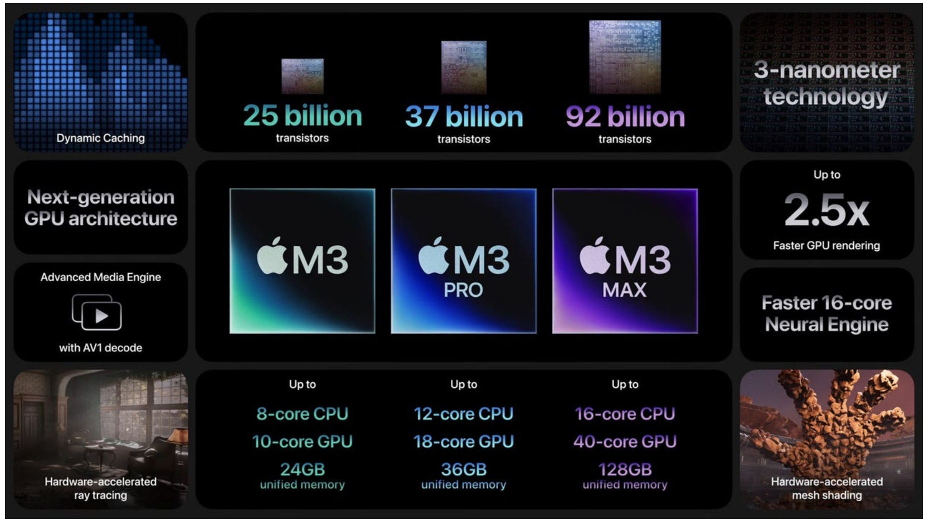
if $J_{ij} > 0$, then $\{s_i, s_j\}$={+1,+1} or {-1,-1} : Same phase

if $J_{ij} < 0$, then $\{s_i, s_j\}$={+1,-1} or {-1,+1} : Opposite phase

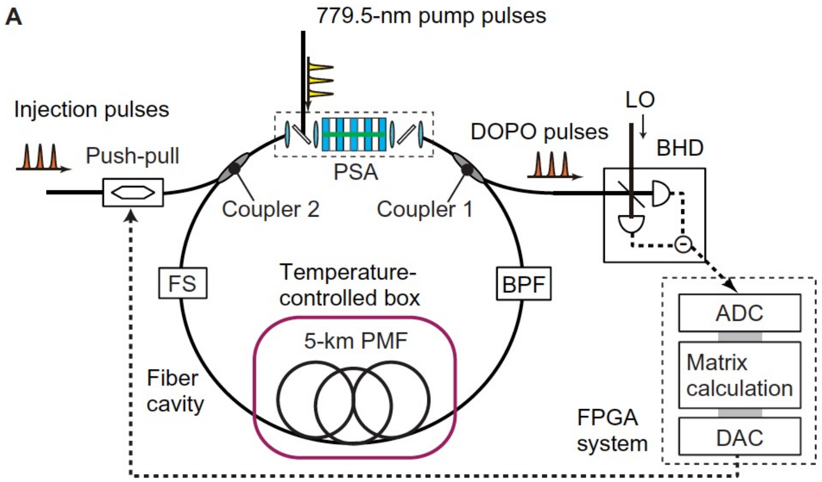


Why CMOS based Ising Machines?

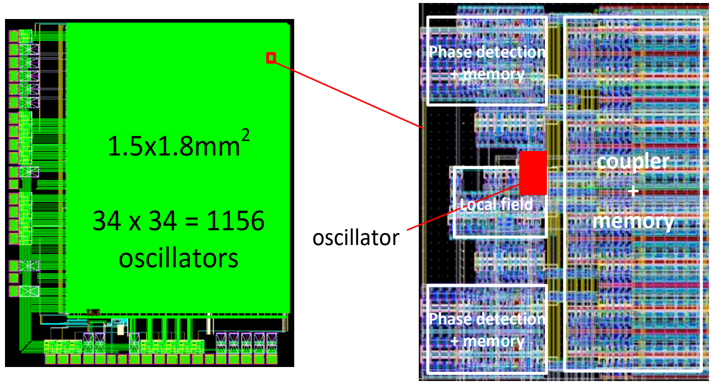




Extreme Ultra-violet (EUV) equipment, ~$150M

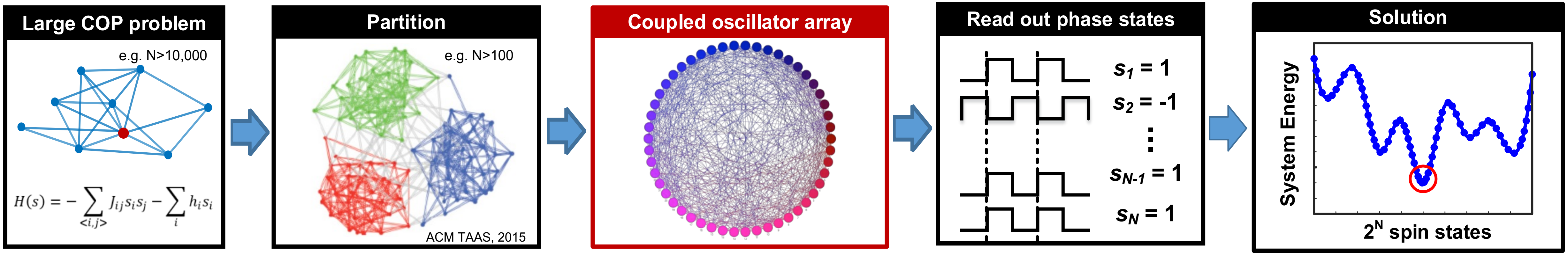


100,000 spin coherent Ising machine with **56 FPGA chips** [Science Advances '21]



Oscillator area < 3% in COBI chips. Rest is memory, coupler, logic, control, and IO.

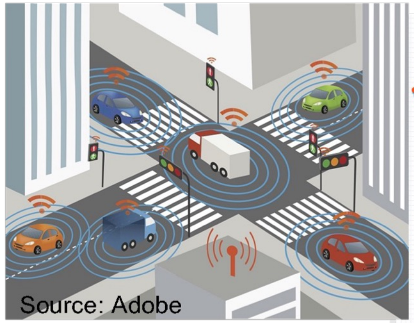
Introduction: Ising Compute Model



$H(s) = -\sum_{<i,j>} J_{ij} s_i s_j - \sum_i h_i s_i$: Ising Hamitonian

s_i, s_j : Spin state {+1 or -1} J_{ij} : Coupling strength h_i : local field strength

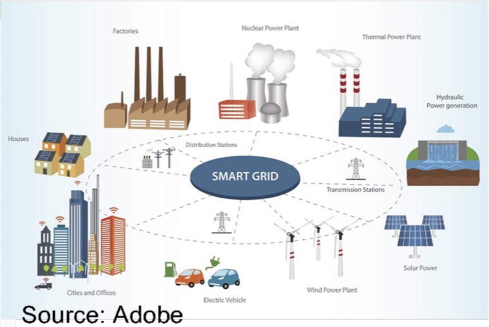
Autonomous vehicles



Source: Adobe

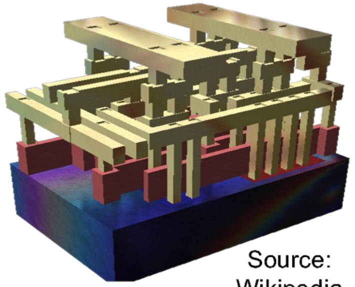
Communication networks

Smart grid



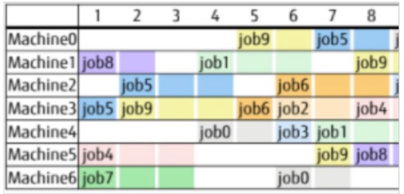
Source: Adobe

VLSI routing



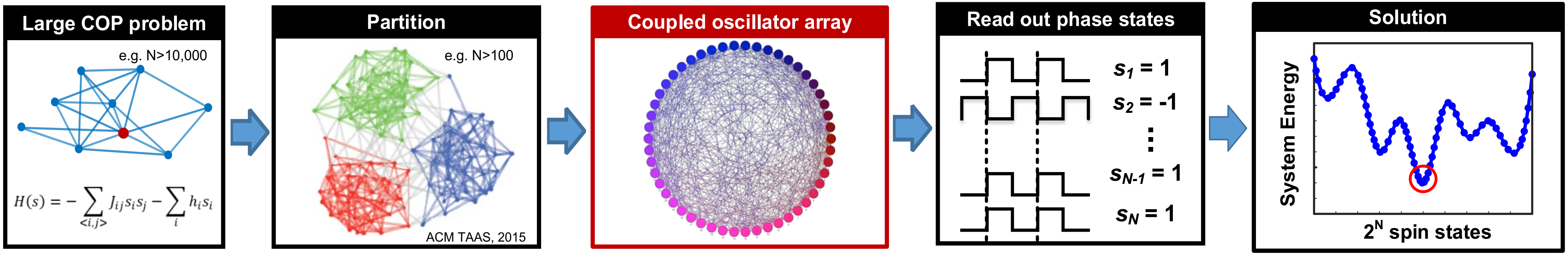
Source: Wikipedia

Scheduling + resource management



Source: Fujitsu

Introduction: Ising Compute Model

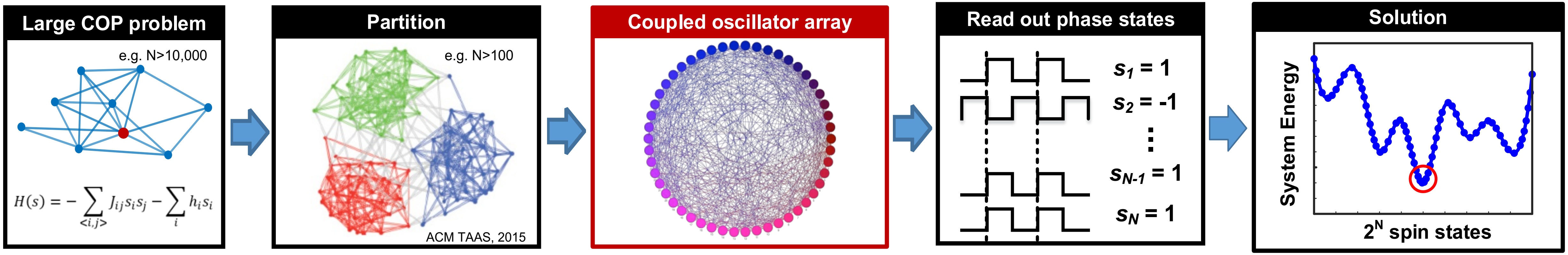


Random states (time=0)

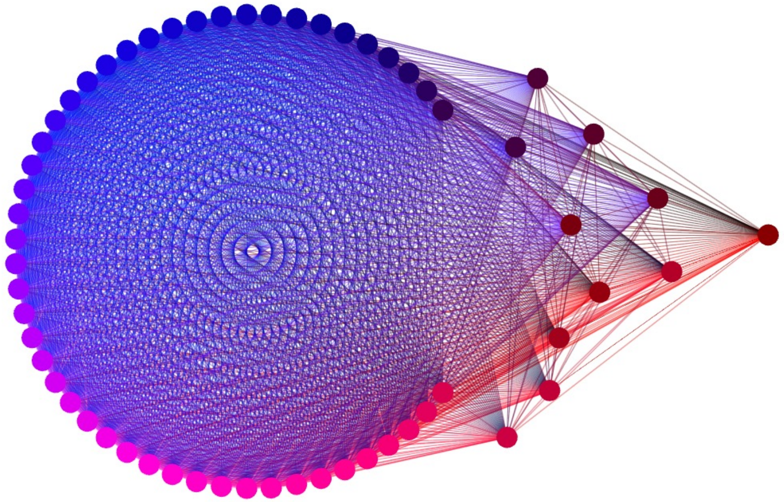
Same states (time = 1 min)

Youtube: Coupled Metronomes

Introduction: Ising Compute Model



8 metronomes with all positive coupling



COBI chip: ~60 metronomes with ~5 bit programmable coupling

Example Toy Problem #1: Factorizing 15

$$\mathrm{p} = (x_1\ 1)_2\ , \mathrm{q} = (x_2\ x_3\ 1)_2$$
$$H = (15 - pq)^2$$

x_i: state {0 or +1}
(QUBO model)

$$H = 128x_1x_2x_3 - 56x_1x_2 - 48x_1x_3 + 16x_2x_3 - 52x_1 - 52x_2 \text{-} 96x_3 + 196$$

$$H_{mod} = 200x_1x_2 - 48x_1x_3 - 512x_1x_4 + 16x_2x_3 - 512x_2x_4 + 128x_3x_4 - 52x_1 - 52x_2 - 96x_3 + 768x_4 + 196$$

S. Jiang, et al., "Quantum Annealing for Prime Factorization", Scientific Reports 2018

Example Toy Problem #2: Graph Coloring

For graph $G(V, E)$ of the map problem—no two vertices, V, connected by an edge, E, should select the same color from set C—construct a cost function with binary variables, $x_{v,c} = 1$ when $v \in V$ selects color $c \in C$, by implementing two constraints:

$$(\sum_c x_{v,c} - 1)^2,$$

Hard constraint

which has minimum energy (zero) when vertices select one color only, and

$$\sum_c \sum_{v_a, v_b \in E} x_{v_a,c} x_{v_b,c},$$

Soft constraint

which adds a penalty if the vertices of an edge select the same color.

These constraints give a QUBO,

$$E(x_v, x_{v_a,v_b}) = \sum_v (\sum_c x_{v,c} - 1)^2 + \sum_c \sum_{v_a,v_b \in E} x_{v_a,c} x_{v_b,c}.$$

D-Wave Problem-Solving Handbook

e.g. $x_{Minn,Red} = 0, x_{Minn,Blue} = 0,$
$x_{Minn,Sand} = 1, x_{Minn,Green} = 0$
$x_{Wisc,Red} = 0, x_{Wisc,Blue} = 1,$
$x_{Wisc,Sand} = 0, x_{Wisc,Green} = 0$

DARPA Quantum-Inspired Classical Computing (QuICC) Program Problem Classes

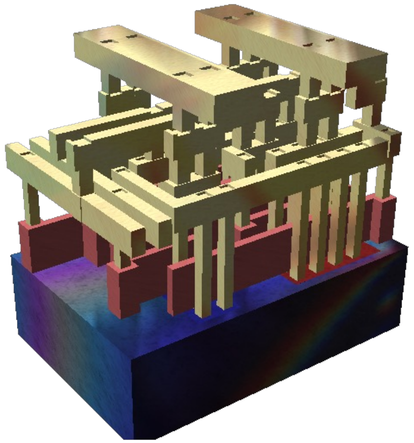
- Cryptanalysis benchmarks, e.g.,reduced-round MD5 or SHA-1
  - Boolean Satisfiability (SAT) problems
- 5G/6G Telecommunication applications in decoding, channel estimation, or precoding (MIMO)
  - Maximum-likelihood Estimation (MLE)
- Electronic reliability fault diagnostics test vector generation
  - Maximum-Fault Minimum Cardinality (MFMC) sampling
- Logistics, e.g., Vehicle Routing Problems (VRP)
  - Mixed-Integer Linear Programming (MILP)

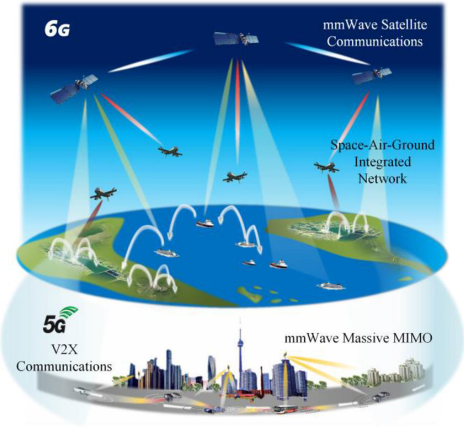
Commercial Market Opportunities

Bloomberg

	BZIC	GPRO	ATER	DEER	HBB	LPL	AAPL	MSFT	
BZIC		-0.66	0.01	-0.52	-0.7	-0.87	-0.78	-0.74	BZIC
GPRO	-0.66		0.52	0.43	0.29	0.59	0.86	0.75	GPRO
ATER	0.01	0.52		-0.25	-0.31	-0.25	0.2	-0.04	ATER
DEER	-0.52	0.43	-0.25		0.35	0.55	0.47	0.58	DEER
HBB	-0.7	0.29	-0.31	0.35		0.75	0.58	0.65	HBB
LPL	-0.87	0.59	-0.25	0.55	0.75		0.81	0.87	LPL
AAPL	-0.78	0.86	0.2	0.47	0.58	0.81		0.94	AAPL
MSFT	-0.74	0.75	-0.04	0.58	0.65	0.87	0.94		MSFT
	BZIC	GPRO	ATER	DEER	HBB	LPL	AAPL	MSFT	

macroaxis





ResearchGate

Manufacturing

- Automotive manufacturing only spends $500 billion
- A 1-5% increasing in productivity could see huge savings
- American manufacturing expenditure is $2.3 trillion
- (National Association of Manufacturers, Mckinsey)

Finance

- Portfolio and risk management
- Loan optimization could free up capital and possibly lower interest rates
- 1% efficiency improvement of a $6.9 Trillion industry could lead to a massive market
- (Mckinsey)

Silicon Design

- Verification, routing, and placement require constant optimization
- Serviceable and obtainable market
- Capture 1% of the $150 billion market
- (Anandtech, Mckinsey)

5G/6G + Edge Compute

- Massive multiple-input multiple-output (MIMO) antenna systems
- Basestation needs to detect optimal configuration in milliseconds → cloud access not possible
- (Acumen, Consegic)

Early Quantum/Quantum-inspired Cloud Computing Services

- Microsoft Azure Quantum Inspired Optimizer (QIO), $90/hr
- CPU based

SKU Quota per month	SOLVERS	PERFORMANCE	PRICE (GLOBAL) Price per compute hour, billed per second
Learn & Develop 1/20 hours of compute (default/max)	Simulated annealing Simulated annealing (Parameter-free) Parallel tempering Parallel tempering (Parameter-free) Quantum Monte Carlo Tabu search	Up to 5 concurrent jobs	0-1 hour - Free 1-20 hours - $10/hour
	FPGA simulated annealing*	Up to 2 concurrent jobs	0-1 hour - Free
Performance at scale 1,000 hours of compute (default) 50,000 hour of compute (max)	Simulated annealing Simulated annealing (Parameter-free) Parallel tempering Parallel tempering (Parameter-free) Quantum Monte Carlo Tabu search	Up to 100 concurrent jobs	0-1 hour - Free 1-100 hours - $90/hour 100-500 hours - $80/hour 500-1,000 hours - $70/hour > 1,000 hours - $50/hour
	FPGA simulated annealing*	Up to 10 concurrent jobs	0-1 hour - Free 1-100 hours - $900/hour 100-500 hours - $800/hour 500-1,000 hours - $700/hour > 1,000 hours - $500/hour

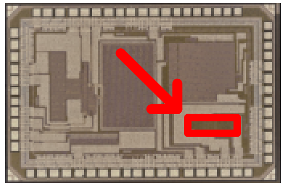
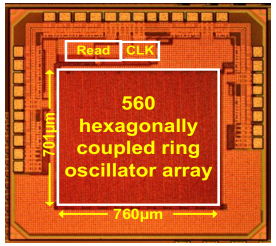
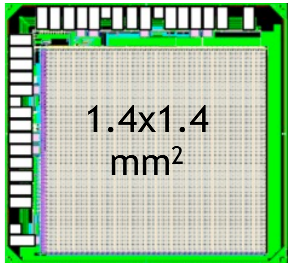
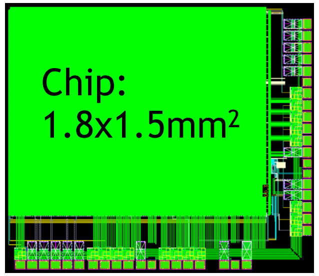
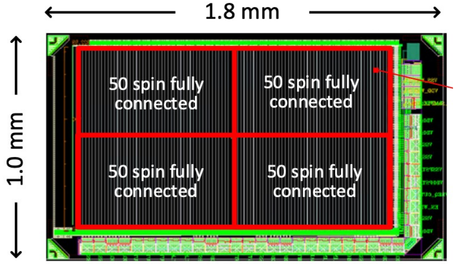
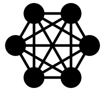
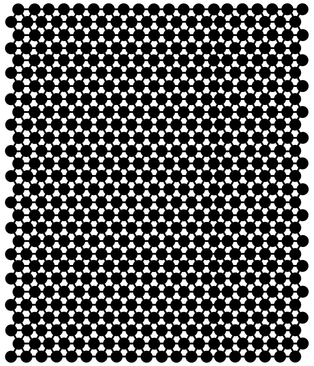
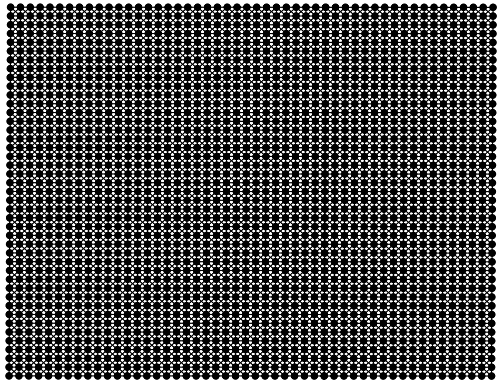
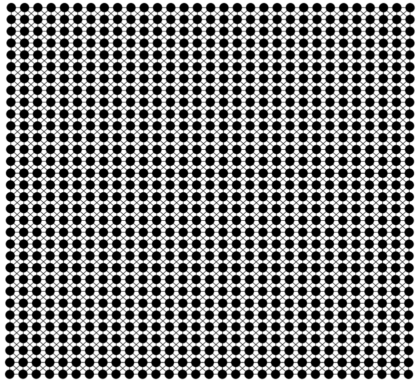
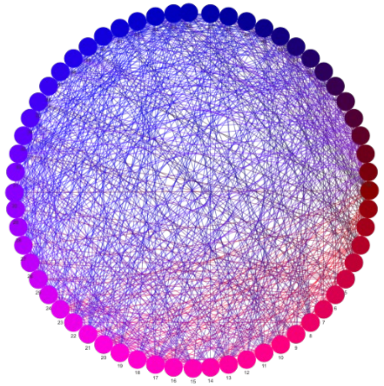
https://azure.microsoft.com/en-us/pricing/details/azure-quantum/

- AWS Braket quantum computer gateway ($0.3 per problem + per shot price)

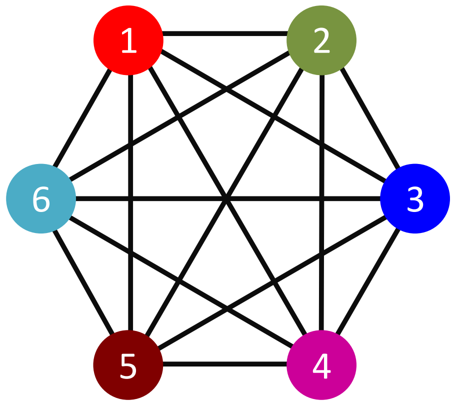
Hardware Provider	QPU family	Per-task price	Per-shot price
D-Wave	2000Q	$0.30000	$0.00019
D-Wave	Advantage	$0.30000	$0.00019
IonQ	IonQ device	$0.30000	$0.01000
OQC	Lucy	$0.30000	$0.00035
Rigetti	Aspen-11	$0.30000	$0.00035
Rigetti	M-1	$0.30000	$0.00035
Xanadu	Borealis	$0.30000	$0.0002

https://aws.amazon.com/braket/pricing/

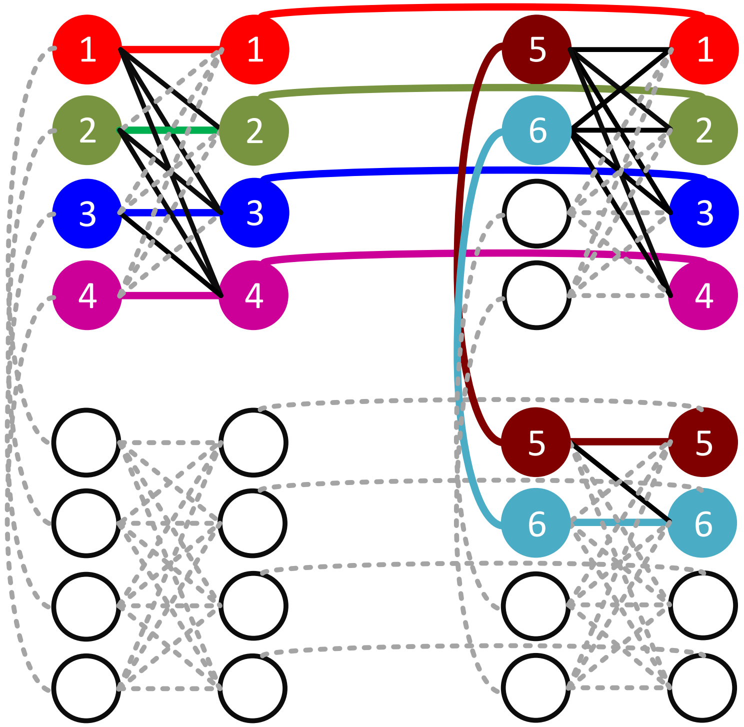
Early COBI Chips

Array size	6 spins (2018)	560 spins (2019-2020)	1,968 spins (2020-2021)	1,024 spins (2021-2022)	48 spins (2022-)
Die photo or layout					
Architecture					
Publications	N/A	VLSI 21, IEEE JSSC 21	Nature Electronics '22	Unpublished	Nature Electronics '23
Connectivity	All-to-all (5)	Hexagonal (6)	King's graph (8)	King's graph (8)	All-to-all, 47
Weight resolution	N/A	3 levels	5 levels	9 levels	29 levels
Power	>100μW per weight	41μW per oscillator (measured)	0.042 W (measured)	0.03W (measured)	0.11W (measured)
Applications	Max-cut	Max-cut	Max-cut	Max-cut	Any QUBO problem

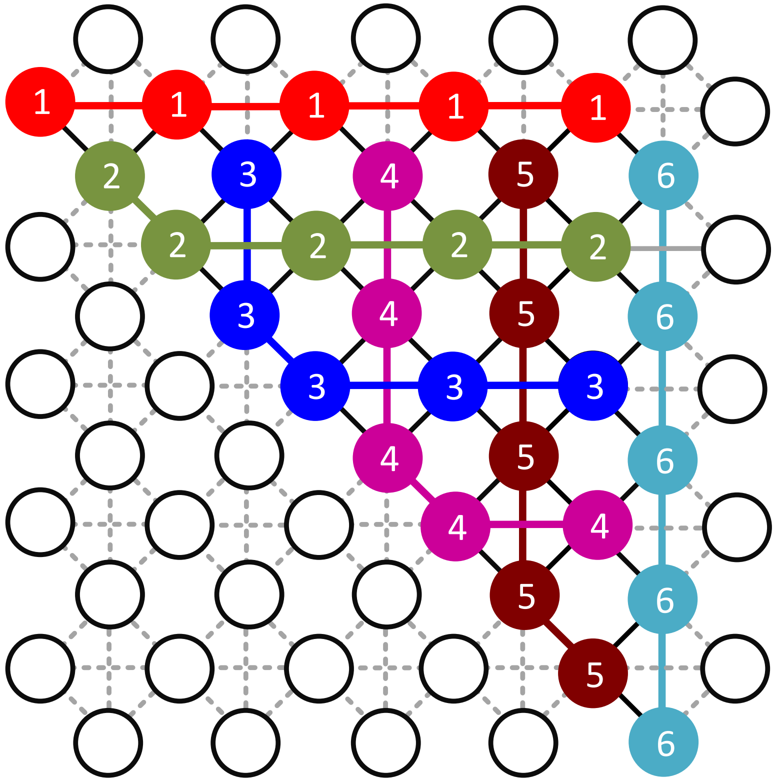
Achieving All-to-All Connectivity



All-to-all graph



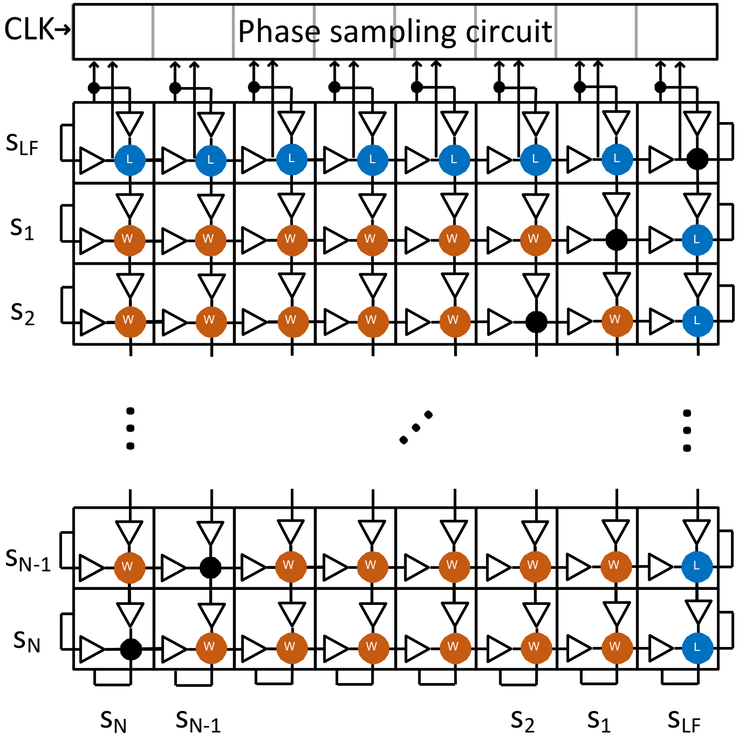
Chimera graph



King's graph

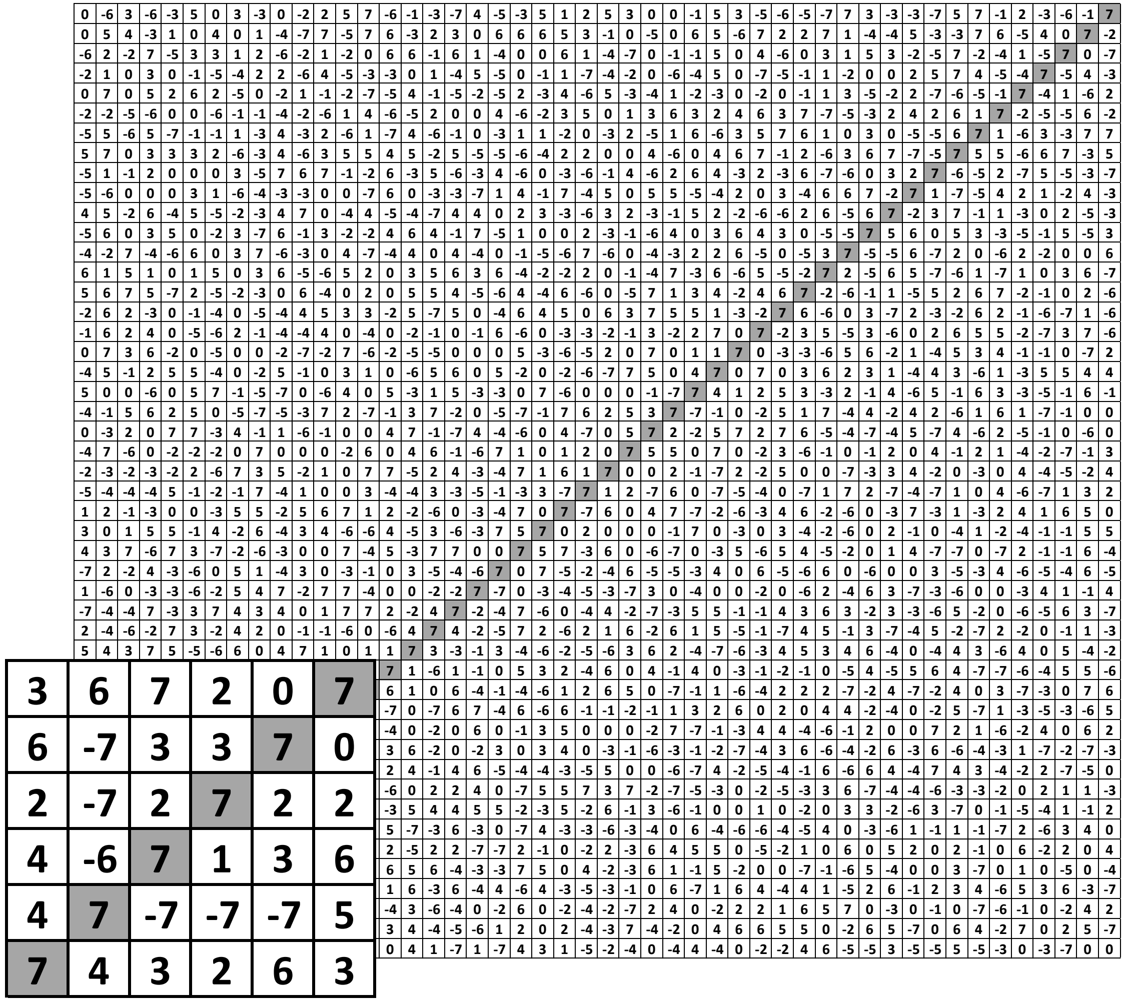
Core Tech #1: All-to-All Architecture

H. Lo, C. Kim, Nature Electronics ‘23, patent pending

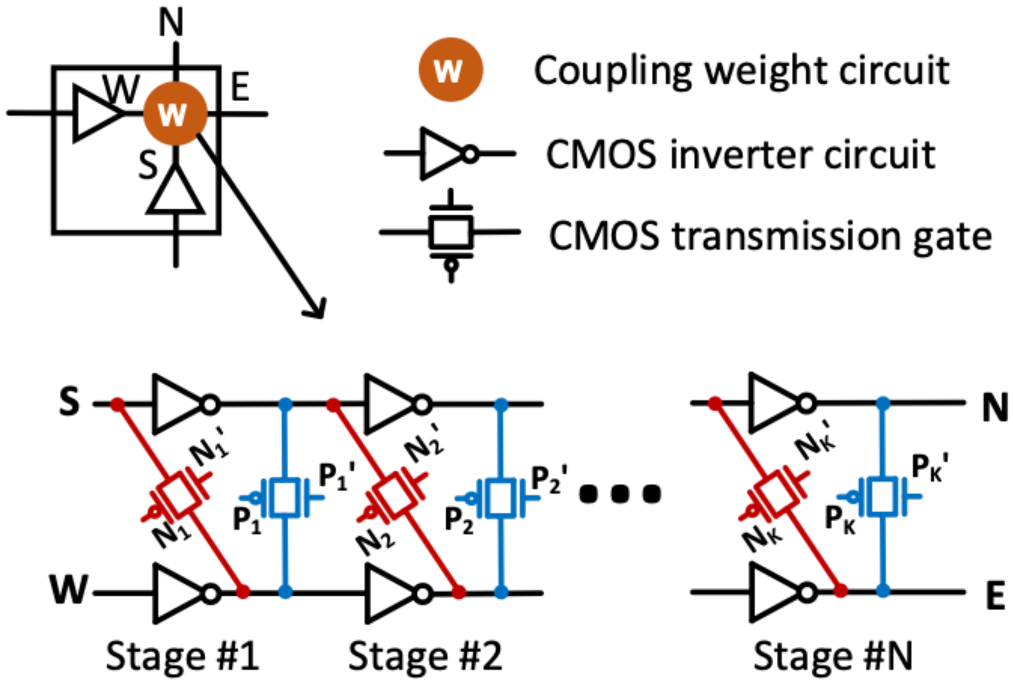


Topology	Degree
King’s graph (Hitachi)	8
Chimera graph (Dwave)	6
Pegasus graph (Dwave)	15
Zephyr (Dwave)	20
All-to-all (Nature Electronics ‘23)	**47**

48x48 adjacency matrix (J_{ij}, h_i = {-14, -13, …, 13, 14})



Core Tech #2: Multi-bit Coupler Circuit



H. Lo, C. Kim, Nature Electronics '23, patent pending

- Red transmission gates: negative coupling
- Blue transmission gates: positive coupling
- Cascaded 1 bit weights ensure uniform coupling

Energy Savings and Speedup Over Classical Computers

COBI chip results					
Random graph density	**Frequency (MHz)**	Current (mA)	**Power (mW)**	Energy per sample (uJ, estimated)	Energy to solution (uJ, estimated)
0	**5.190**	6.870	**8.244**	0.794	7.943
0.1	**5.226**	7.010	**8.412**	0.805	8.047
0.2	**5.243**	7.150	**8.580**	0.818	8.183
0.3	**5.251**	7.300	**8.760**	0.834	8.341
0.4	**5.267**	7.430	**8.916**	0.846	8.463
0.5	**5.276**	7.590	**9.108**	0.863	8.632
0.6	**5.276**	7.660	**9.192**	0.871	8.712
0.7	**5.308**	7.900	**9.480**	0.893	8.929
0.8	**5.313**	8.000	**9.600**	0.904	9.035
0.9	**5.329**	8.160	**9.792**	0.919	9.188
1	**5.358**	8.350	**10.020**	0.935	9.351

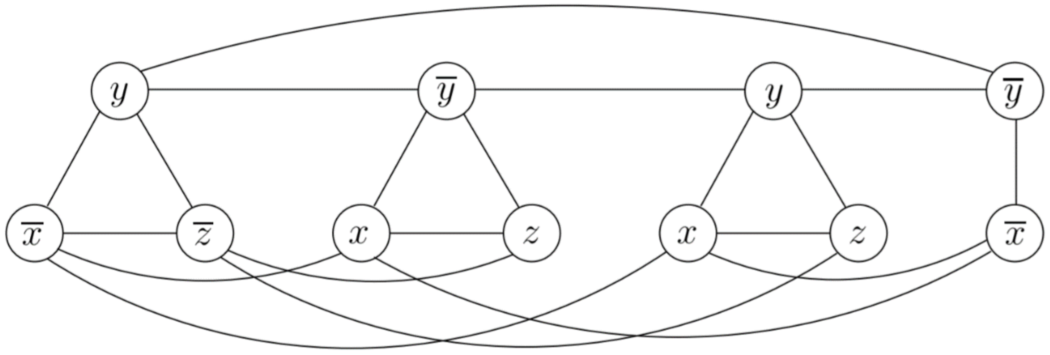
Classical computing	
CPU spec	Intel Xeon Gold 6240R, 14nm FinFET, 2.4GHz, 256GB, 0.75V
Classical solver software	Dwave qbsolv (tabu search)
CPU time to solution	2.25 msec
CPU power (Watts over baseline, measured)	3 (TDP = 165W)
Energy to solution (uJ)	6750
COBI energy savings	**722 X**
COBI speedup	**2.35 X**

- 722X energy savings, 2.35X speedup in spite of COBI chip being implemented in 65nm

Applications (3SAT Case Study), Open Questions, Future Directions

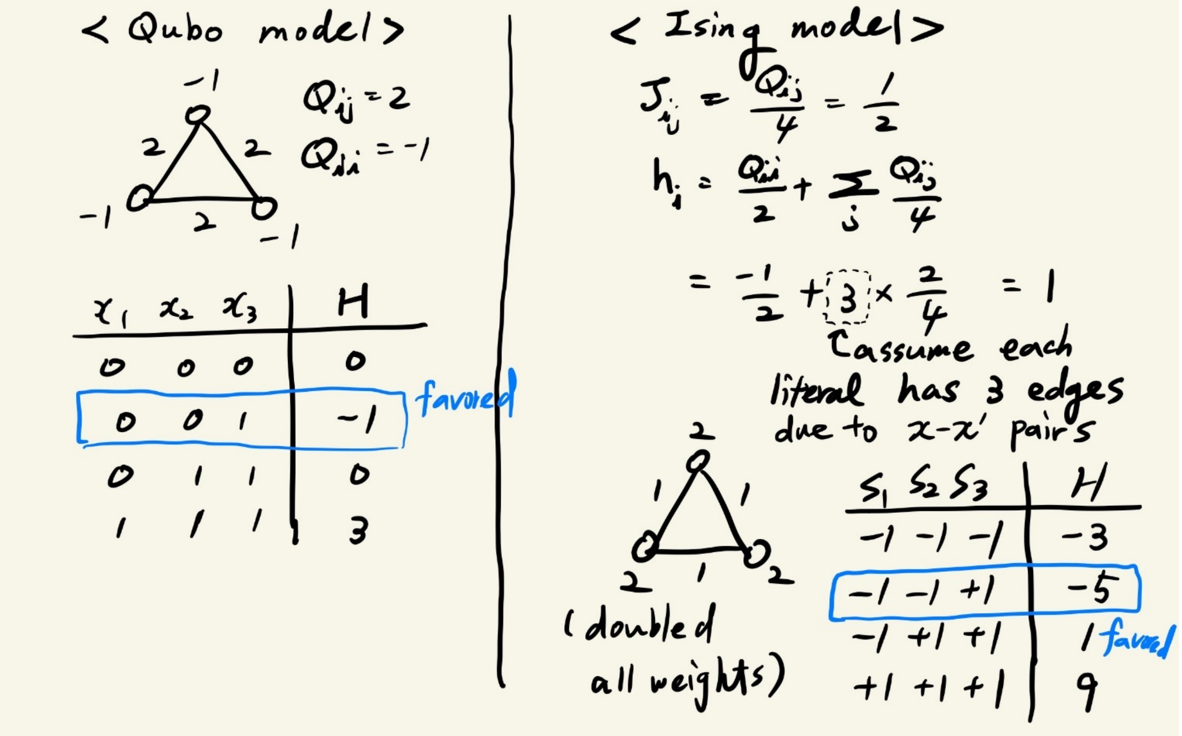
Solving 3SAT (Satisfiability) Using QUBO

Figure 8.8 The graph corresponding to $(\overline{x} \vee y \vee \overline{z})\ (x \vee \overline{y} \vee z)\ (x \vee y \vee z)\ (\overline{x} \vee \overline{y})$.



[1] S. Dasgupta, C. Papadimitriou, U. Vazirani (2008), Algorithms, McGraw Hill.

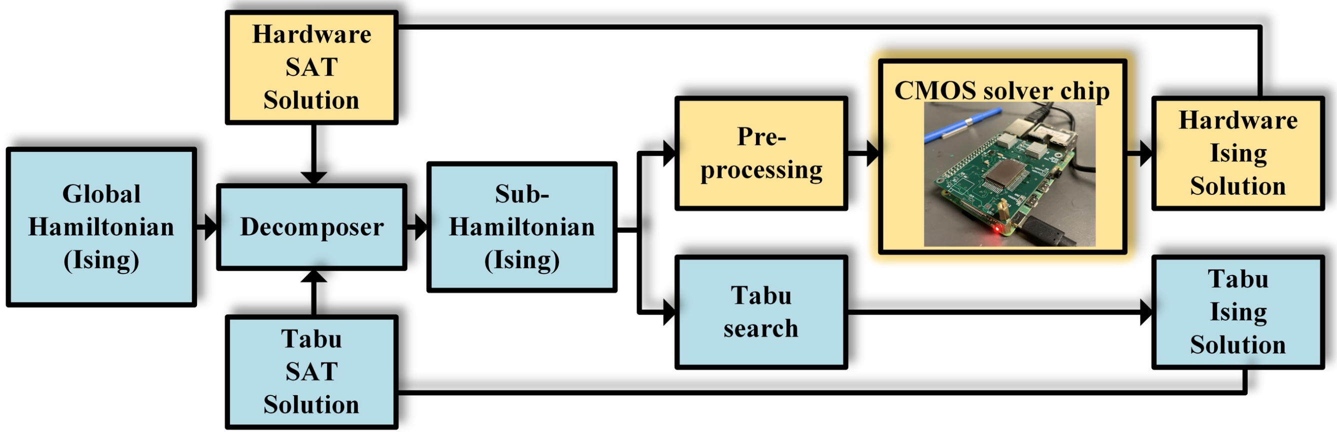
- A 3SAT expression can be mapped to a graph
  - Create a node for each literal
  - Connect nodes in the same clause
  - Connect nodes between opposite literals
- Then solve the Maximum Independent Set (MIS) on the graph while ensuring that the set size is equal to the number of clauses [1]



Hardware-Software Co-Design Framework

- 5 formulations
  - 3SAT benchmark with *n* variables + *m* clauses

Abbreviation	Formulation	# of Spins	Reference
MIS	Maximal independent set	*3m*	[Choi 10]
ILP	Integer linear programming	*n+2m*	[Ours]
C(n+m)	Chancellor with n+m spins	*n+m*	[Chancellor 16]
N(n+m)	Nüßlein with n+m spins	*n+m*	[Nüßlein 23]
N(2n+m)	Nüßlein with 2n+m spins	*2n+m*	[Nüßlein 23]



H. Cilasun, C. Kim, U. Karpuzcu, S. Sapatnekar, "SAT on an All-to-All-Connected CMOS Ising Solver Chip" arXiv preprint arXiv:2309.11017 (2023).

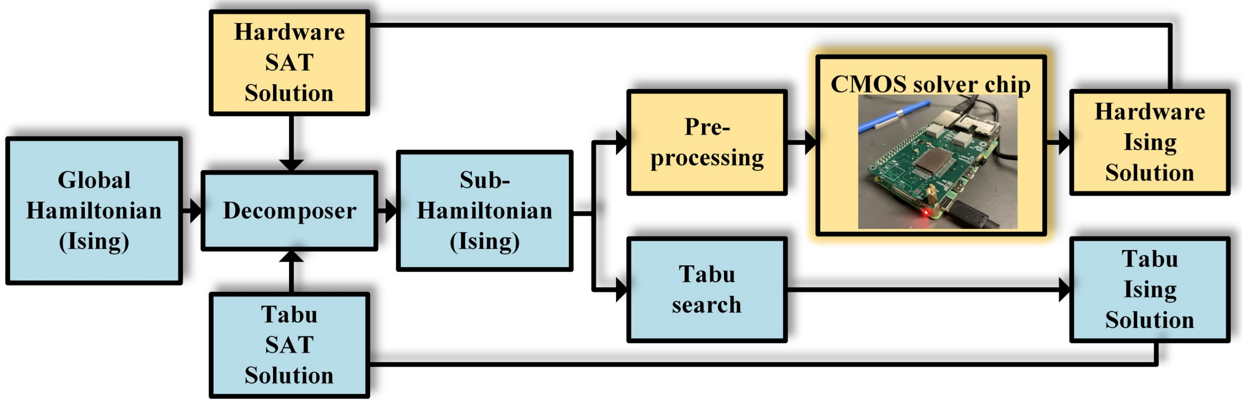
- 5 decomposers
  - Random: Randomly and independently select the spins in all iterations.
  - Energy impact: Always choose the spins that flipping can reduce the energy.
  - Pseudo random: Randomly but use reshuffle and shift to reduce the probability to select the same spins in iterations.
  - BFS: Randomly select a source spin and then use breadth first search (BFS) to select other spins.
  - SAT: Randomly choose clauses and then select all spins related to the clauses.

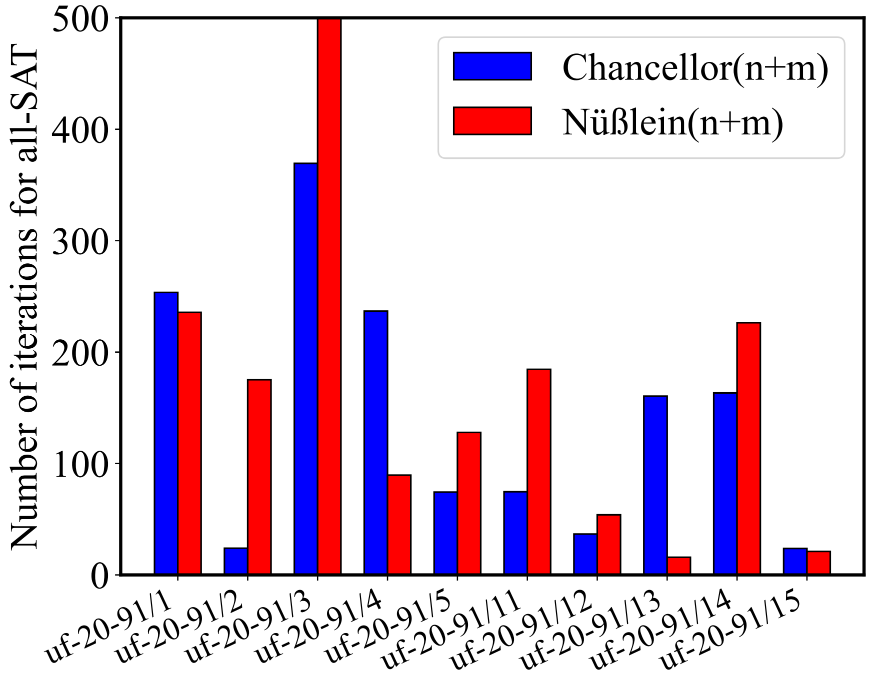
[Choi 10]: "Adiabatic quantum algorithms for the NP-complete maximum-weight independent set, exact cover and 3SAT problems." *arXiv preprint arXiv:1004.2226* (2010).
[Chancellor 16]: "A direct mapping of Max k-SAT and high order parity checks to a chimera graph." *Scientific reports* 6.1 (2016): 37107.
[Nüßlein 23]: "Solving (Max) 3-SAT via Quadratic Unconstrained Binary Optimization." *arXiv preprint arXiv:2302.03536* (2023).

COBI Chip Results

- 3SAT benchmarks: 20 variable, 91 spins
- Formulations and decomposer with highest performance
  - [Chancellor with n+m spins]
  - [Nüßlein with n+m spins]
  - [BFS decomposer]
- COBI chip used to solve the sub-QUBO
  - 100 samples for each sub-Hamiltonian
  - Choose the solution with best energy
- Hyperparameters
  - Scaling factor: 8, Local Field ROs: 4
  - Number of samples per sub-QUBO: 100

H. Cilasun, C. Kim, U. Karpuzcu, S. Sapatnekar, "SAT on an All-to-All-Connected CMOS Ising Solver Chip" arXiv preprint arXiv:2309.11017 (2023).





Key Challenge: Digital Computation Overhead

Dwave Qbsolv software (open source)
https://github.com/dwavesystems/qbsolv

```
# Get initial estimate of minimum value and backbone
solution ← random 0/1 vector
(best_energy, best_solution) ← TabuSearch(QUBO, solution)
index ← OrderByImpact(QUBO, best_solution)
passCount ← 0
solution ← best_solution
while passCount < numRepeats do
    change ← false
    for i = 0; i < fraction * Size(QUBO); i += subQUBOSize do
        # select subQUBO with other variables clamped
        sub_index ← i : i+subQUBOSize−1
        subQUBO ← Clamp(QUBO, solution, index[sub_index])
        (sub_energy, sub_solution) ← DWaveSearch(subQUBO)
        # project onto full solution
        if (solution[sub_index] ≠ sub_solution) then
            solution[sub_index] ← sub_solution
            change ← true
        end if
    end for
    if not change then
        Randomize(solution[0 : i − 1])
    end if
    (energy, solution) ← TabuSearch(QUBO, solution)
    if energy < best_energy then
        best_energy ← energy
        best_solution ← solution
        passCount ← 0
    else
        passCount + +
    end if
    index ← OrderByImpact(QUBO, solution)
end while
Output: best_energy, best_solution
```

COBI chip (arrow pointing to the DWaveSearch line)

Hamiltonian calculation (arrow pointing to the TabuSearch line)

Compute resource: Intel Xeon 22C, 2.1GHz, 30.25M Cache, 256G

Problem Size	Hamiltonian	Iterations	Total time (secs)	QPU time (secs)	Hamilt. time (secs)	QPU / Total
100	-194.79	129	0.59	0.59	0	100%
1000	-5573.54	570	2.96	2.86	0.09	96.6%
2500	-20390.63	1210	7.44	6.68	0.74	89.8%
5000	-60034.81	2376	22.02	16.18	5.79	73.5%
10000	-169716.36	5203	86.44	49.08	37.22	56.8%

- Need hardware accelerators for direct computation of local and global Hamiltonian energies

COBI Progress Summary

- Optimization is poised to become a critical workload in the future
- Physics-based Ising solvers can leverage nature to find a competitive ground state of an Ising Hamiltonian energy function
  - Conservative estimate: 1,000x lower energy-to-solution and 10x shorter time-to-solution
- Latest COBI chip: 50-60 spins, all-to-all connectivity, ~5 bit weights, >99% solution accuracy, 0.010W
- Future plans
  - Demonstrate scalability
  - Design ASIC chip with many Ising cores, controller, AXI, PCIe, etc.
  - Software-hardware co-design
  - Quantum-inspired algorithm and ecosystem development