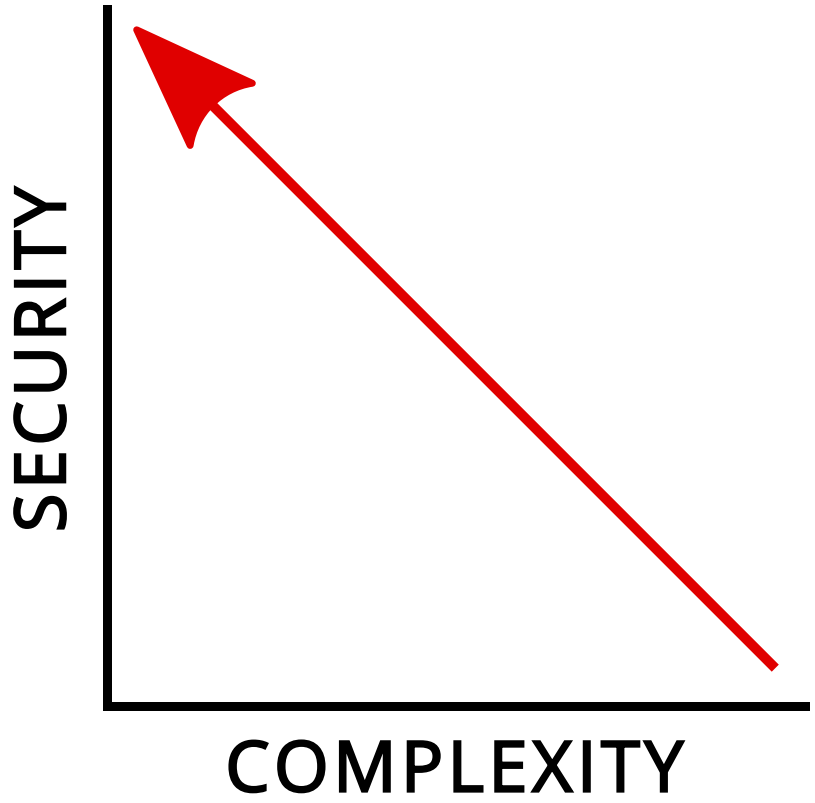
When Makers Make Secure CPUs

Marc W. Abel

Wakefield Cybersecurity

To increase security, reduce complexity



What's wrong with our hardware?

- Too complex
- Beyond control of purchasers
- Brazenly spiteful of programmers
- Oligopolist suppliers

Categories of vulnerability-inducing hardware irregularities

Category	I	II	III
Origin	purposeful	unexpected	malicious
Example	arithmetic wrap	RowHammer	hidden backdoor
Software fix?	yes	no	no
VLSI fix?	yes	yes	no
Manufacturing fix?	yes	yes	yes

Category I example: Integer wraparound

(Category I irregularities exist for a purpose.)

C programmers used to write:

```
    c = a + b;
```

Today, they would need to write:

```
if (b > 0 && a > (INT_MAX - b) || b < 0 && a < (INT_MIN - b))
    longjmp(CATCHIT, SIGNED_ADD_OVERFLOW);
else
    c = a + b;
```

Some well-known Category II irregularities

(Category II irregularities are unplanned and unexpected.)

When	Architecture	Name	Synopsis
1985	80386	multiply bug	arithmetic error
1994	Pentium	FDIV	arithmetic error
1998	Pentium	F00F	lockup
2003	Via C3	God mode	privilege escalation
2008	Intel AMT	Silent Bob	full control of everything
2015	DRAM	RowHammer	memory corruption
2017	x86	Spectre	read others' memory
2017	x86, POWER, ARM	Meltdown	read all memory
2020	Intel SGX	load value inj.	inject data values
2020	Intel CSME	[M. Ermolov]	broken authentication

Actual and rumored Category III exploits

(Category III irregularities are intentionally malicious.)

Who	Architecture	Synopsis
AMD	Platform Security Processor	hypothesized backdoor
Apple	iPhone 6 + iOS 10.2.1	sabotaged performance
Deere	8520T tractor	right to repair infringements
Huawei	5G cellular infrastructure	potential for China influence
Intel	Management Engine	hypothesized backdoor
Intel	RDRAND instruction	non-randomness suspicions
NSA	ANT Catalog	implantable surveillance products
VIA	C3 (x86 clone)	backdoors claimed by C. Domas
ZTE	5G cellular infrastructure	potential for China influence

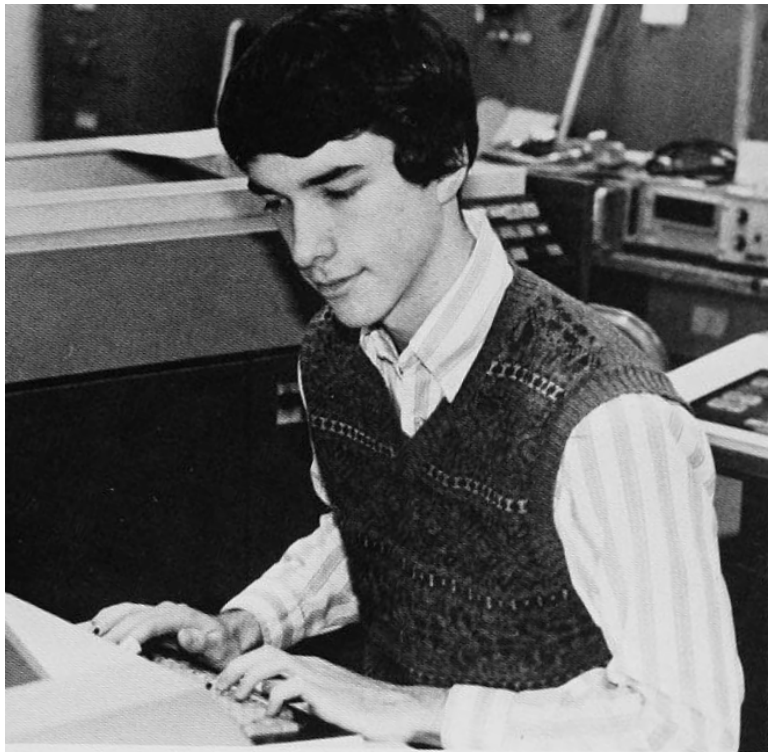
Proposed Category III countermeasures

Proponent	Synopsis
Michael Pompeo	geopolitical controls
Adam Waksman	lock down VLSI supply chain
Eric Love	add formal proofs of security to hardware IP
Mirko Holler	X-ray ptychographic inspection
Marc Abel	complex logic to be built by end user

The architecture of this talk targets all categories

Category	I	II	III
Origin	purposeful	unexpected	malicious
Example	arithmetic wrap	RowHammer	hidden backdoor
Software fix?	yes	no	no
VLSI fix?	yes	yes	no
Manufacturing fix?	yes	yes	yes

Computers were once BIG



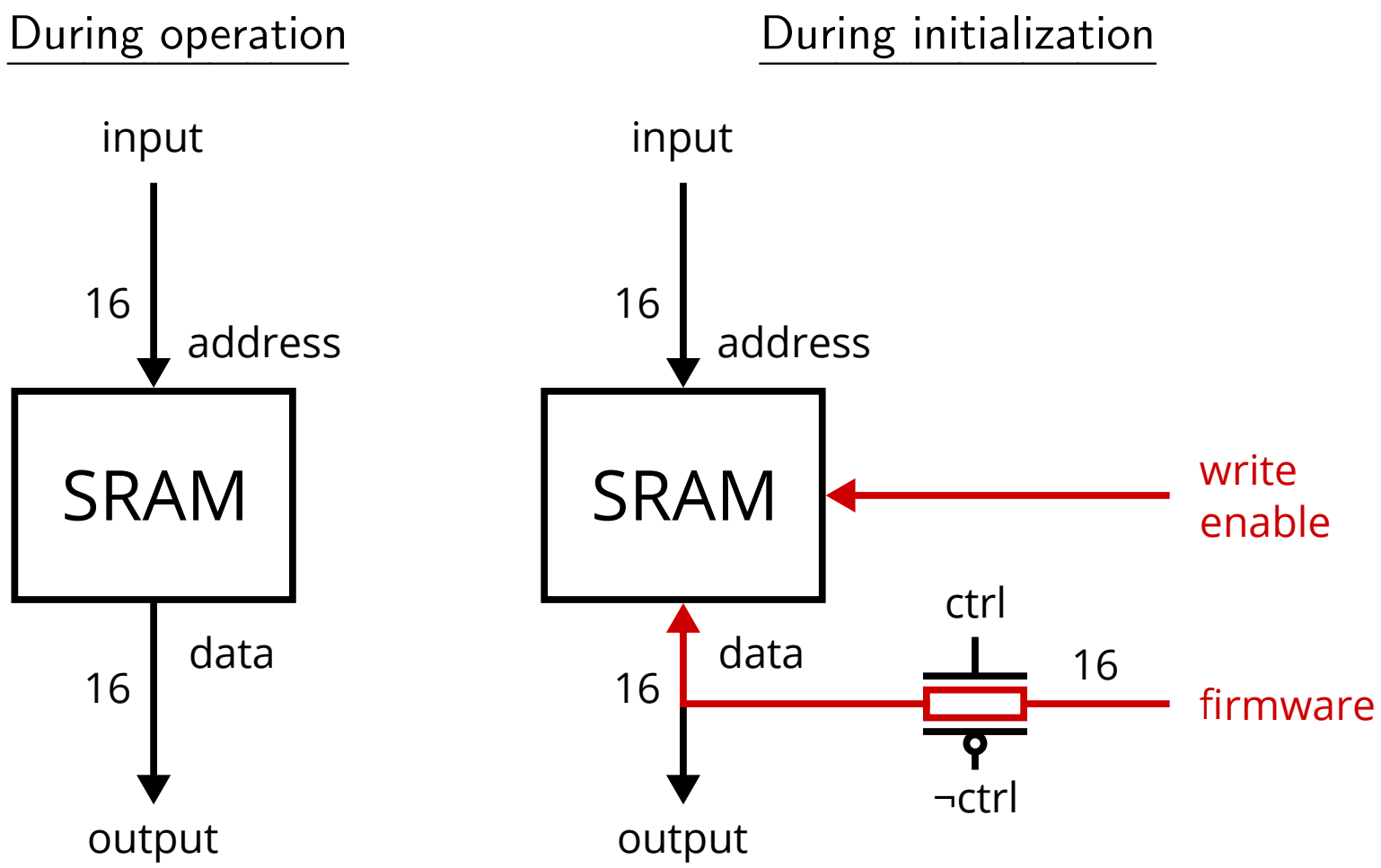
The speaker using an IBM 1130.

Norwester, 62, p. 73 (1986).
Used with permission.

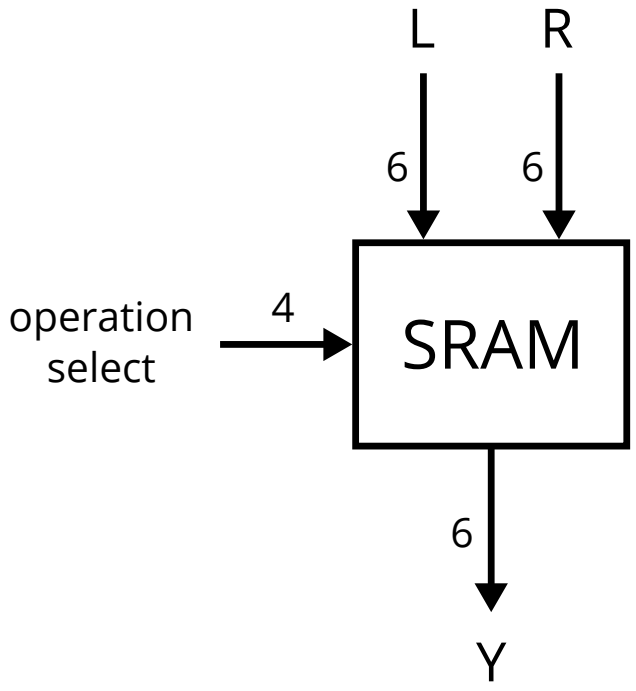
Alternative logic families

Technology	Challenges
mechanical relays	contacts only last a few million instructions
vacuum tubes	power; weight; availability
transistors	not designed for this use; board capacitance
solid-state relays	slow; more cost-effective logic exists
7400 “glue logic” derivatives	mostly discontinued; mostly dismal speed
current-mode logic	very high cost (despite promising speed)
mask ROM	not on market; high economic order quantity
EPROM, EEPROM	self-erasure over years; slow
NOR flash	self-erasure over years; slow
NAND flash	unsuitable interface; self-erasure; slow
dynamic RAM (DRAM)	unsuitable interface; complexity; exploits
PLDs, FPGAs	too few suppliers; susceptible to backdoors

SRAM logic gate



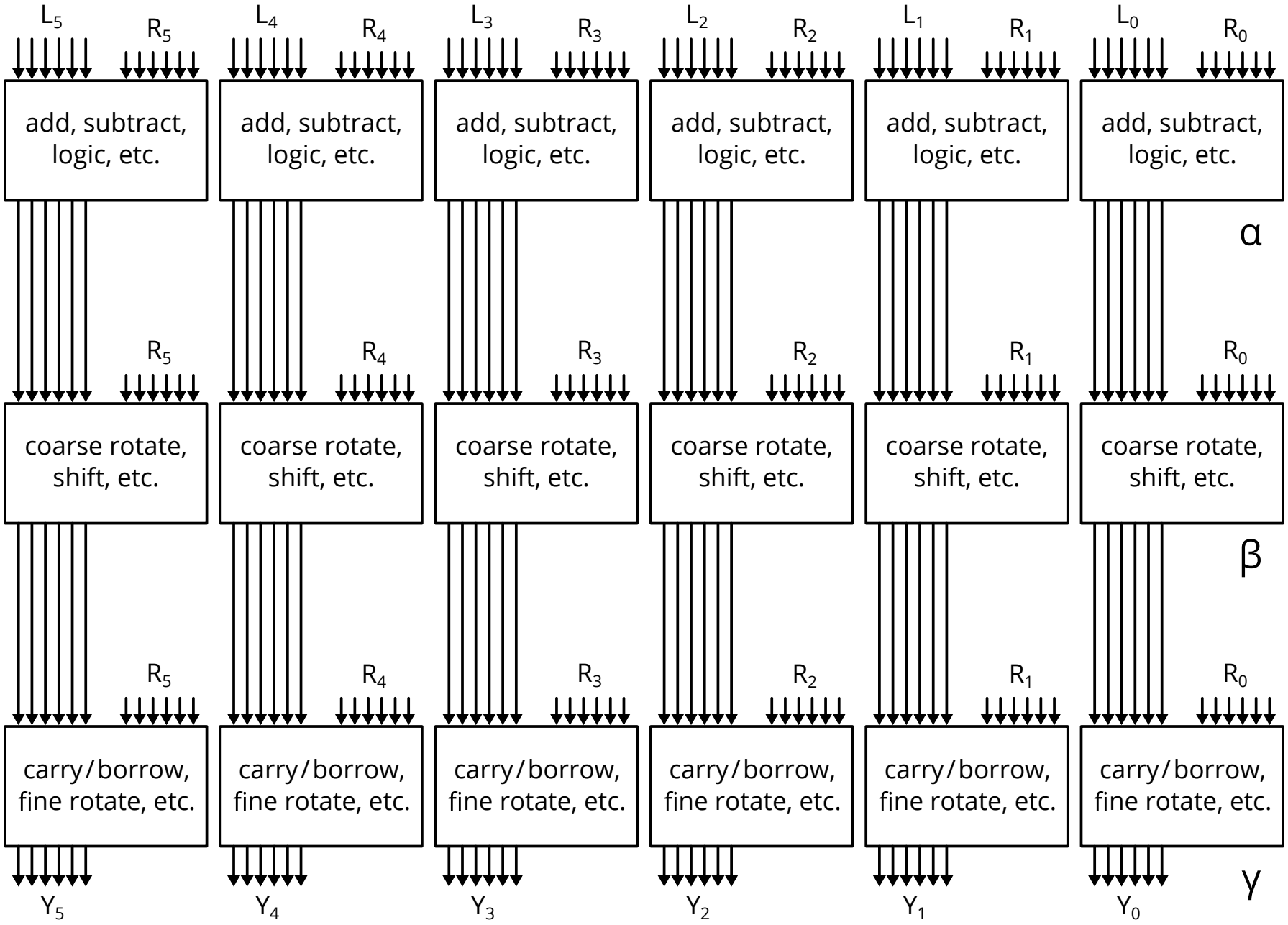
SRAM logic gate sample application



Select	Use
0	L + R
1	L − R
2	R − L
3	NOT L
4	L AND R
5	L NAND R
6	L OR R
7	L NOR R
8	L XOR R
9	L AND NOT R
10	L × R, low 6 bits
11	L × R, high 6 bits
12	shifts/rotations of L (select by R)
13	permutations of L (select by R)
14	L encrypted with key R
15	L decrypted with key R

36-bit ALU with 18 SRAMs

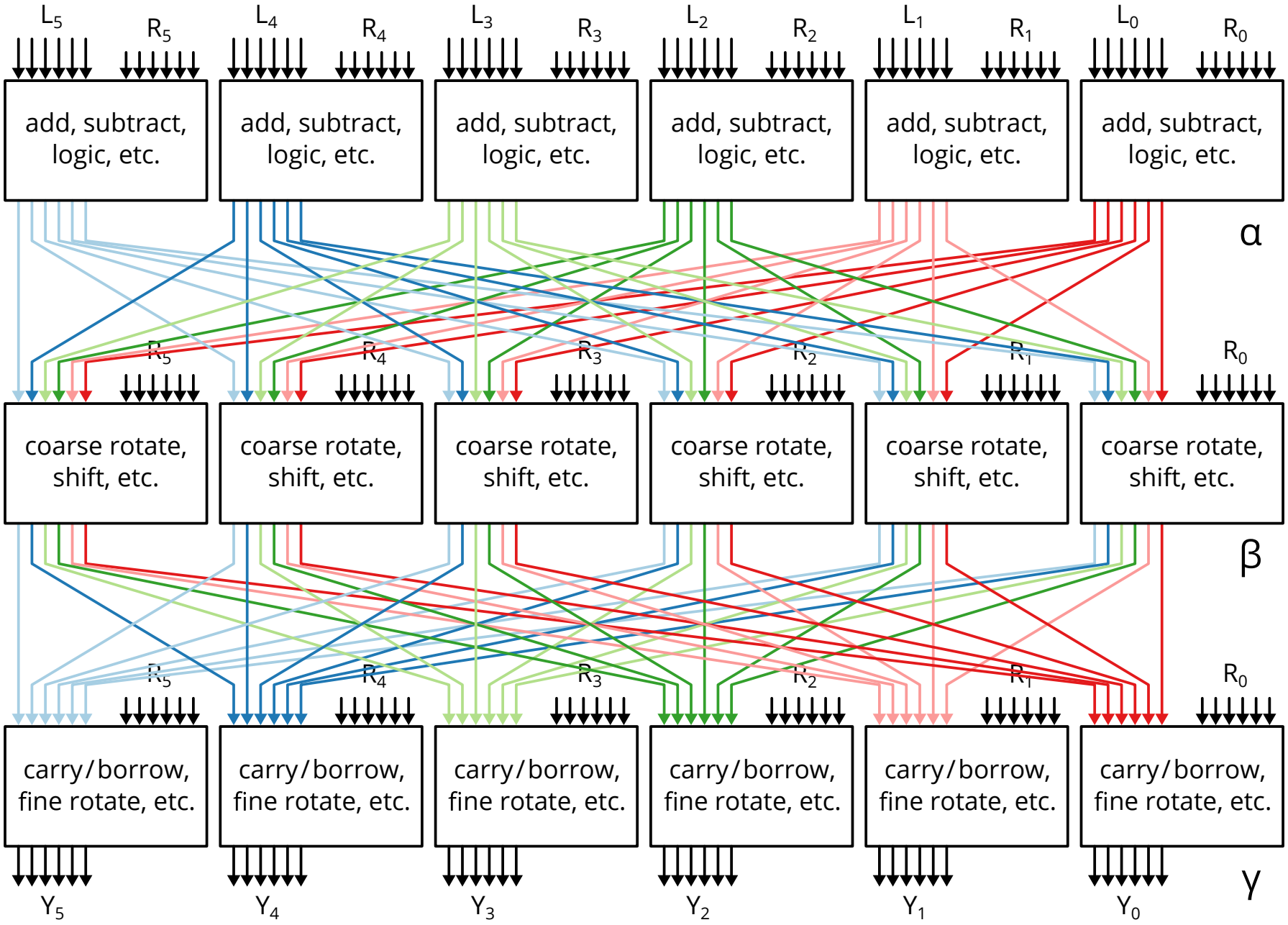
L and R: 36-bit Left and Right operands, 6 bits per slice



Y: 36-bit result, 6 bits per slice

36-bit ALU with 18 SRAMs

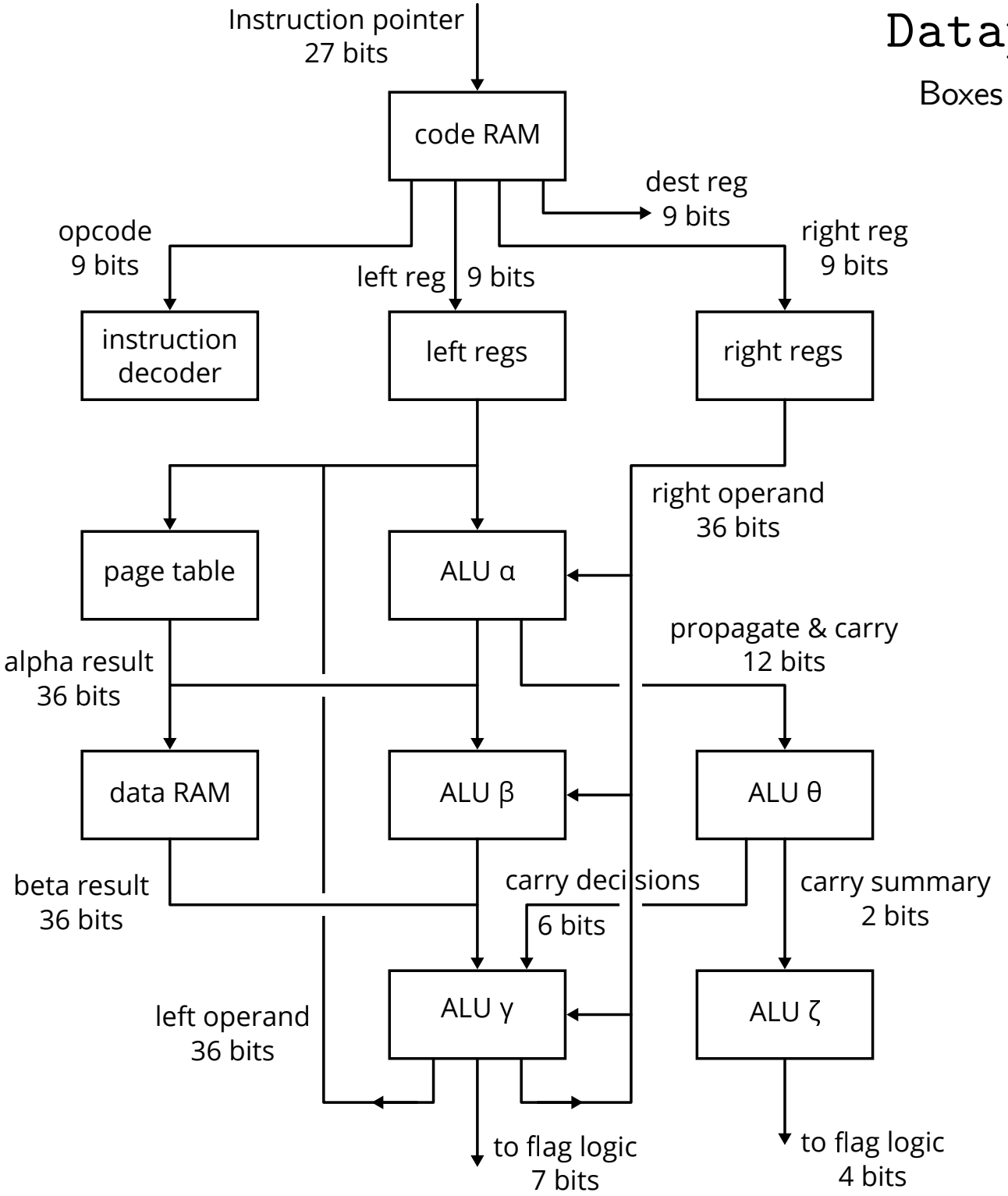
L and R: 36-bit Left and Right operands, 6 bits per slice

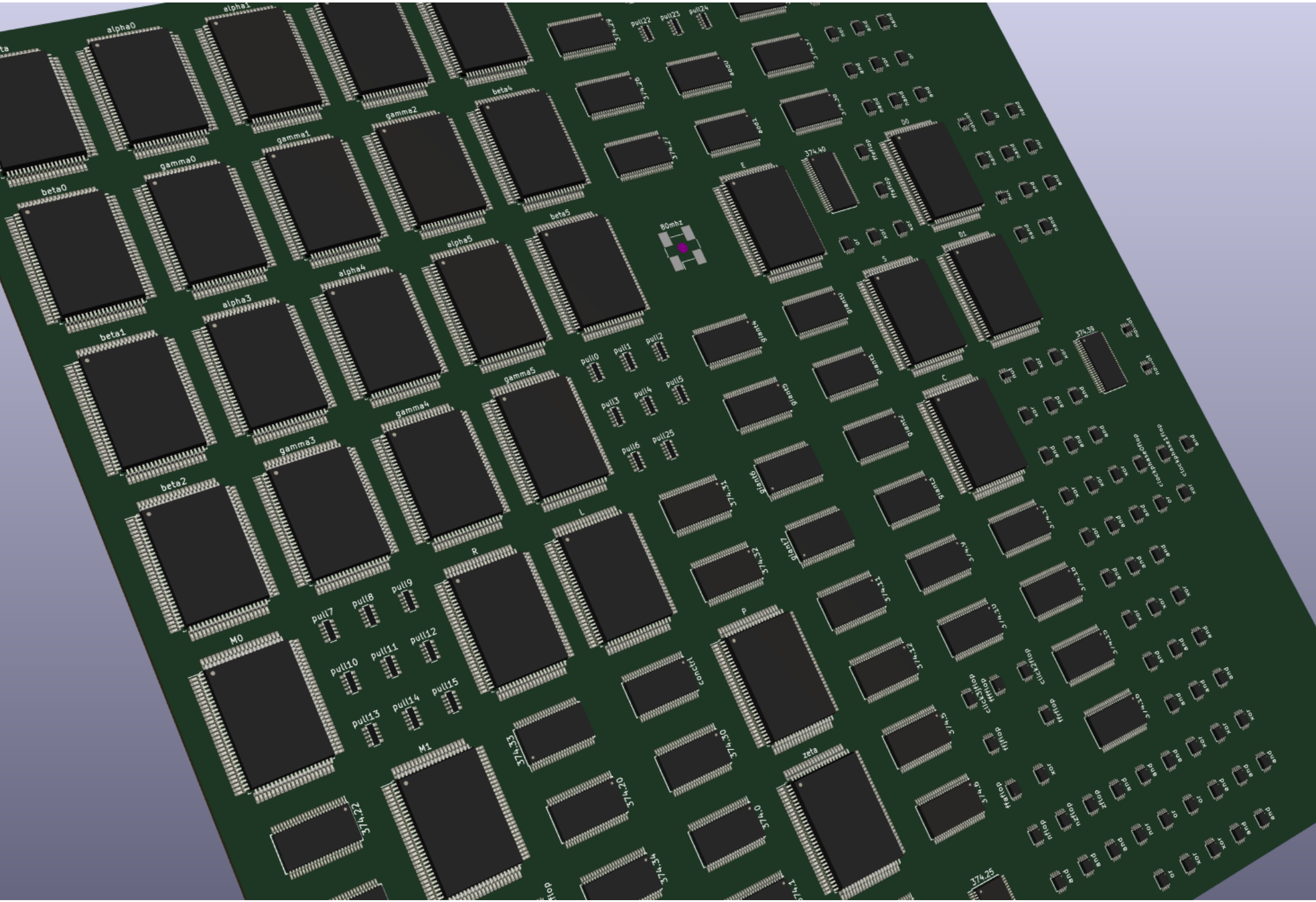


Y: 36-bit result, 6 bits per slice

Datapath summary

Boxes represent static RAM.




alpha0
alpha1
alpha2
alpha3
alpha4
alpha5
beta0
beta1
beta2
beta4
beta5
gamma0
gamma1
gamma2
gamma3
gamma4
gamma5
80mhz
pull0
pull1
pull2
pull3
pull4
pull5
pull6
pull7
pull8
pull9
pull10
pull11
pull12
pull13
pull14
pull15
pull22
pull23
pull24
pull25
M0
M1
R
L
P
S
C
D0
D1
zeta
374.40
374.39
374.25

Not a toy computer!

Operation	CPU cycles
read or write data memory	1
bitwise Boolean logic (16 operations)	1
add, subtract	1
magnitude compare, maximum, minimum	1
shift/rotate by 0 to 63 bit positions	1
reverse bits of 36-bit word	1
leading or trailing bit manipulation (40 operations)	1
36-bit linear feedback shift register	1
hash function for associative arrays, per word hashed	1
round function for 36-bit cipher	1
absolute value	2
pseudorandom number, per word output	2
population count (Hamming weight)	2
count leading or trailing zeros or ones	2
unsigned multiply 36 bits $\times$ 6 bits	3
any permutation of 36 bits	≤ 5
36-bit multiply with 72-bit result (rivals Intel 80486)	35

Not a broken computer!

- Sticky out-of-range flag for all arithmetic
- Mixed sign variants for add, subtract, multiply, shift, absolute value
- No stack overflow possible
- No program access to stack except CALL and RETURN
- No branch to addresses not present in the instruction word
- No privilege escalation via the CPU
- No DRAM or DRAM-associated vulnerabilities
- No complex logic from IC manufacturers within CPU
- Every I/O device confined to its own bus and buffer
- No CPU persistent state except for one firmware IC
- No secret functionality
- No vendor lock-in
- No encrypted or closed-source firmware
- No license fees to build, use, or modify
- No purpose of use limitations
- No right to repair infringements

What are the tradeoffs?

- compatible with NOTHING on the planet ... by design
- new OS, toolchains, software, and docs needed for everything
- no talent already familiar with the hardware
- unlikely to ever run Linux, support GCC or CLANG, etc.
- speeds above 20 MIPS not on the immediate horizon
- memory size constrained by SRAM market
- higher cost per unit, around $1,000
- native peripherals limited to SPI and I^2C buses for now
- larger and more energy-consuming than prevalent architectures
- not yet peer-reviewed or validated by field experience

On the other hand, you truly own it.

Potential applications

Fast enough for

- hardened desktop applications
- electronic mail
- light- to moderate-use servers
- control objects that move
- process controls
- peripheral and device controllers
- telephony
- modest Ethernet switches

Too slow for

- most Web surfing
- machine learning
- image and video processing
- self-driving vehicles
- fast raster or vector graphics
- fast symmetric cryptography
- fast asymmetric cryptography
- micro air vehicles

Questions and answers

Thank you for spending this time with us!

Word size	36 bits
MIPS	20
Memory protection	yes
Multitasking	preemptive
Registers per program	512
Programs ready to run	256
I/O buses	SPI and I^2C
Maximum code RAM	4M $\times$ 36 bits
Maximum data RAM	8M $\times$ 36 bits
Manufacturer	you

WAKEFIELD CYBERSECURITY

https://wakesecure.com