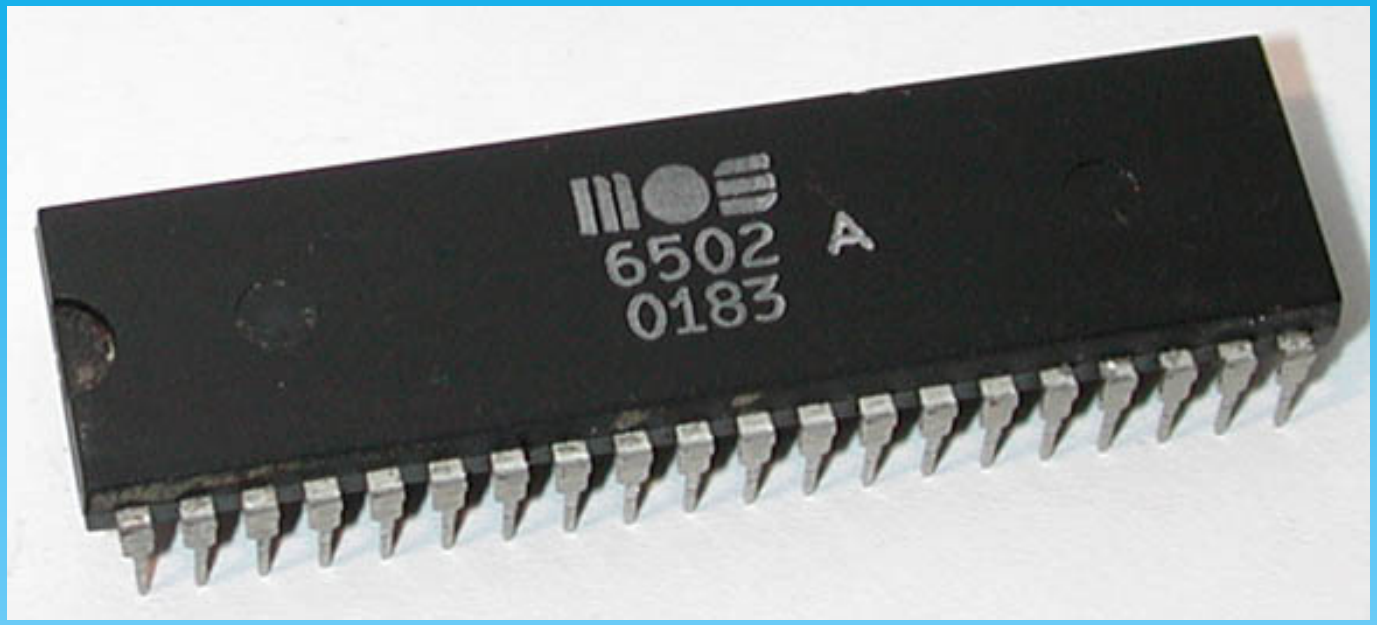
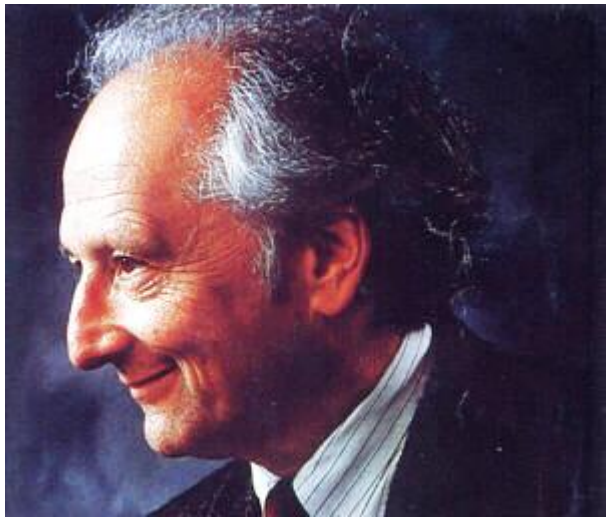


Jason Dagit
Twitter: @thedagit
Blog: dagit.github.io

Motorola in 1970's

* 1971 Microprocessor project starts
* Chuck Peddle joined in 1973 as an engineer
* In 1974, Chuck grew frustrated with management for ignoring customers (asking for $25 processor)
* $300 in 1974 is $1300 in 2010



6500 Project at MOS

* Chuck Peddle, Bill Mensch, and 6 other engineers left Motorola
* MOS was eager to break into processor market
* Based on Motorola 6800 experience
* Goals:
  * Needed to outperform 6800
  * Cheaper than 6800
  * Every interested engineer and hobbyist can get access

Lowering the Cost

* Size = money
  * 3510 transistors (modern CPUs use billions!)
* Defect rate at the time of 70%
* Morally the first RISC processor

Defect Rate

* In 1970's processors were designed by hand
* Images had to be reduced to fit on the wafer
* MOS developed a process for clarifying reduction at each step
  * 70% failure rate during fabrication
    ➔ 70% success rate
* Bill Mensch: Legendary layout engineer

RISC Processor

* Simplified addressing modes
  * Dropped 16bit index register
  * Three-state control of bus removed
* Only the essential instructions: 56 instructions
  * Not completely true: included non-essential BCD arithmetic
* Very few registers: PC, SP, A, X, Y, Status

Instruction Set

		IMMEDIATE			ABSOLUTE			ZERO PAGE			ACCUM.			IMPLIED			(IND, X)			(IND), Y			Z. PAGE, X			ABS, X			ABS, Y			RELATIVE			INDIRECT			Z. PAGE, Y			CONDITION CODES					
MNEMONIC	OPERATION	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	N	Z	C	I	D	V
LDX	M → X (1)	A2	2	2	AE	4	3	A6	3	2																			BE	4	3											B6	4	2	✓	✓	–	–	–	–
LDY	M → Y (1)	A0	2	2	AC	4	3	A4	3	2													B4	4	2	BC	4	3														✓	✓	–	–	–	–
LSR	0 → 7 0 → C				4E	6	3	46	5	2	4A	2	1										56	6	2	5E	7	3														0	✓	✓	–	–	–
NOP	NO OPERATION													EA	2	1																									–	–	–	–	–	–
ORA	A ∨ M → A	09	2	2	0D	4	3	05	3	2							01	6	2	11	5	2	15	4	2	1D	4	3	19	4	3											✓	✓	–	–	–	–
PHA	A → Ms S-1 → S													48	3	1																									–	–	–	–	–	–
PHP	P → Ms S-1 → S													08	3	1																									–	–	–	–	–	–
PLA	S + 1 → S Ms → A													68	4	1																									✓	✓	–	–	–	–
PLP	S + 1 → S Ms → P													28	4	1																									(RESTORED)					
ROL	7 0 C				2E	6	3	26	5	2	2A	2	1										36	6	2	3E	7	3														✓	✓	✓	–	–	–
ROR	C 7 0				6E	6	3	66	5	2	6A	2	1										76	6	2	7E	7	3														✓	✓	✓	–	–	–
RTI	(See Fig. 1) RTRN INT													40	6	1																									RESTORED)					
RTS	(See Fig. 2) RTRN SUB													60	6	1																									–	–	–	–	–	–
SBC	A-M-C̄ → A (1)	E9	2	2	ED	4	3	E5	3	2							E1	6	2	F1	5	2	F5	4	2	FD	4	3	F9	4	3											✓	✓	(3)	–	–	✓
SEC	1 → C													38	2	1																									–	–	1	–	–	–
SED	1 → D													F8	2	1																									–	–	–	–	1	–
SEI	1 → I													78	2	1																									–	–	–	1	–	–
STA	A → M				8D	4	3	85	3	2							81	6	2	91	6	2	95	4	2	9D	5	3	99	5	3											–	–	–	–	–	–
STX	X → M				8E	4	3	86	3	2																															96	4	2	–	–	–	–	–	–
STY	Y → M				8C	4	3	84	3	2													94	4	2																	–	–	–	–	–	–
TAX	A → X													AA	2	1																									✓	✓	–	–	–	–
TAY	A → Y													A8	2	1																									✓	✓	–	–	–	–
TSX	S → X													BA	2	1																									✓	✓	–	–	–	–
TXA	X → A													8A	2	1																									✓	✓	–	–	–	–
TXS	X → S													9A	2	1																									–	–	–	–	–	–
TYA	Y → A													98	2	1																									✓	✓	–	–	–	–

(1) ADD 1 TO "N" IF PAGE BOUNDARY IS CROSSED

(2) ADD 1 TO "N" IF BRANCH OCCURS TO SAME PAGE
ADD 2 TO "N" IF BRANCH OCCURS TO DIFFERENT PAGE

(3) CARRY NOT = BELOW

(4) IF IN DECIMAL MODE Z FLAG IS INVALID
ACCUMULATOR MUST BE CHECKED FOR ZERO RESULT

X INDEX X
Y INDEX Y
A ACCUMULATOR
M MEMORY PER EFFECTIVE ADDRESS
Ms MEMORY PER STACK POINTER

+ ADD
– SUBTRACT
∧ AND
∨ OR
∀ EXCLUSIVE OR
✓ MODIFIED

– NOT MODIFIED
M_7 MEMORY BIT 7
M_6 MEMORY BIT 6
N NO CYCLES
NO BYTES

Improvements over 6800

* Pipelining
* Zero-page addressing
  * Allowed indirect indexing to give 128 pseudo registers
  * Faster than normal memory access
* Programmers trained on the 6800 found the 6502 intuitive
* Almost the same clock speed, but nearly 4x the computational power

Marketplace

* 6502 went on sale in Sept. 1975 for about $20
  * That would be $80 today
  * Instruction manual for additional $10
* Made possible the Personal Computer revolution
* Instant success

Motorola's Reaction

* By late 1975, Motorola reorganized its semiconductor division
* Motorola sued MOS, who settled in 1976 for $200,000 (over $700,000 today)
* Price of 6800 lowered to $69

PC Revolution

Video Games

Marketplace (cont.)

* 6502 and predecessors used in:
  * Apple I/II
  * Atari 2600/Atari home computers
  * Commodore:
    * PET/VIC-20/C-64/C-128
    * Disk drive controller for 1541
  * BBC Micro
  * Nintendo's NES/SNES
* And many other systems…

MOS & 6502 Today

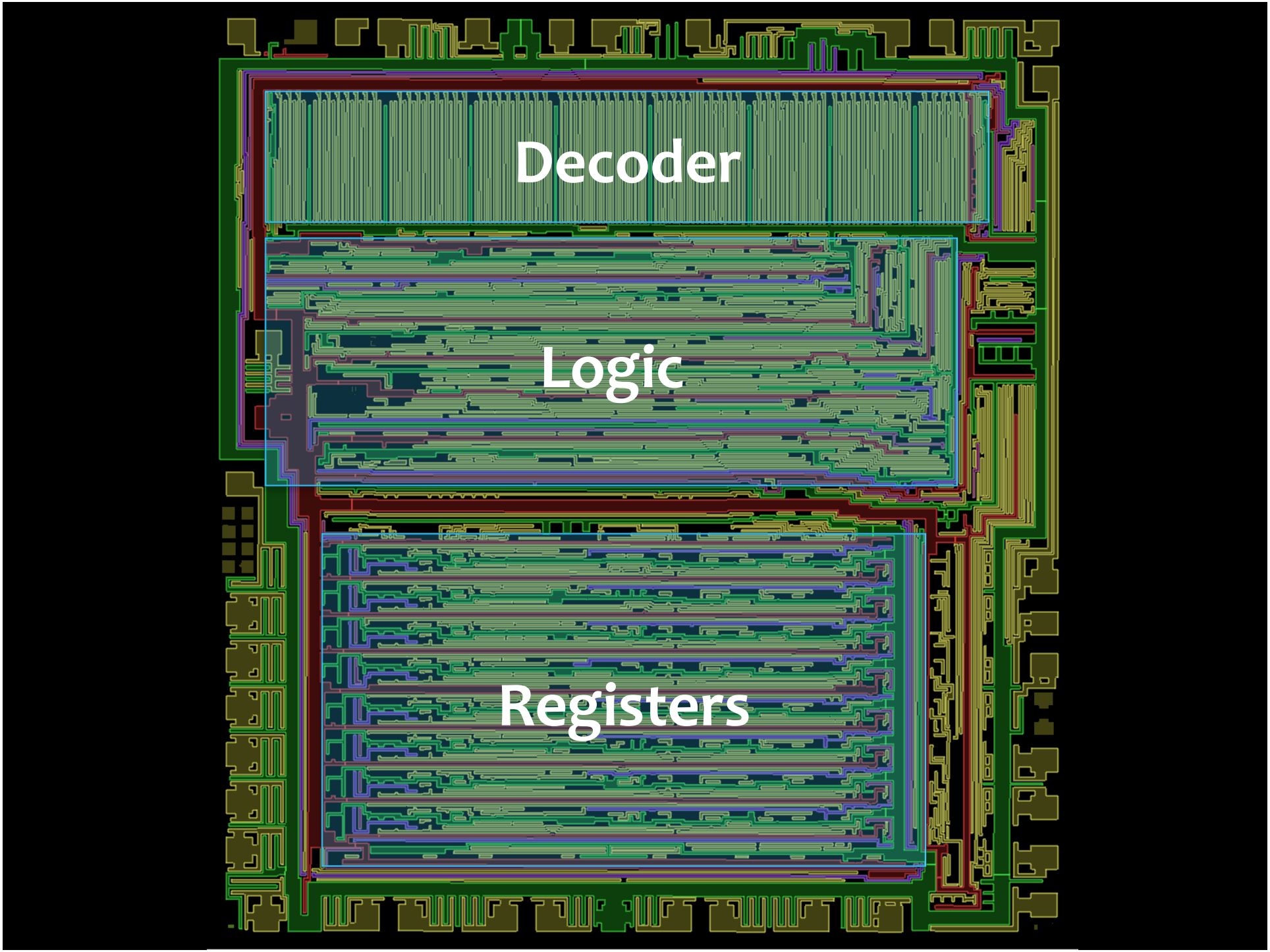
* MOS acquired by Commodore
  * Commodore Semiconductor Group
* Commodore split up and bankrupt
* CSG becomes GMT Microelectronics
  * … Until the EPA has to step in
* Western Design Center
  * WDC 65C02

Lessons

* Chuck Peddle:
  * Listen to your customers
  * Listen to your sales people/engineers
  * Everyone should have access
* Simplicity matters:
  * Reduced cost
  * Better optimized
* Takes more than a great product to run a business

Visual6502.org

* Transistor level reverse engineered netlist
* Multiple 6502 emulators (physical simulations of wires)
* Goal: Study, document, and archive the 6502 design for future generations to enjoy


Decoder
Logic
Registers

Questions?

Jason Dagit
Twitter: @thedagit
Blog: dagit.github.io

Sources

* https://en.wikipedia.org/
* http://www.commodore.ca/history
* http://www.westegg.com/inflation/
* http://oldcomputers.net/appleii.html
* http://www.cpushack.com/CPU/cpu1.html
* http://visual6502.org/
* http://research.swtch.com/6502
* http://events.ccc.de/congress/2010/Fahrplan/events/4159.en.html