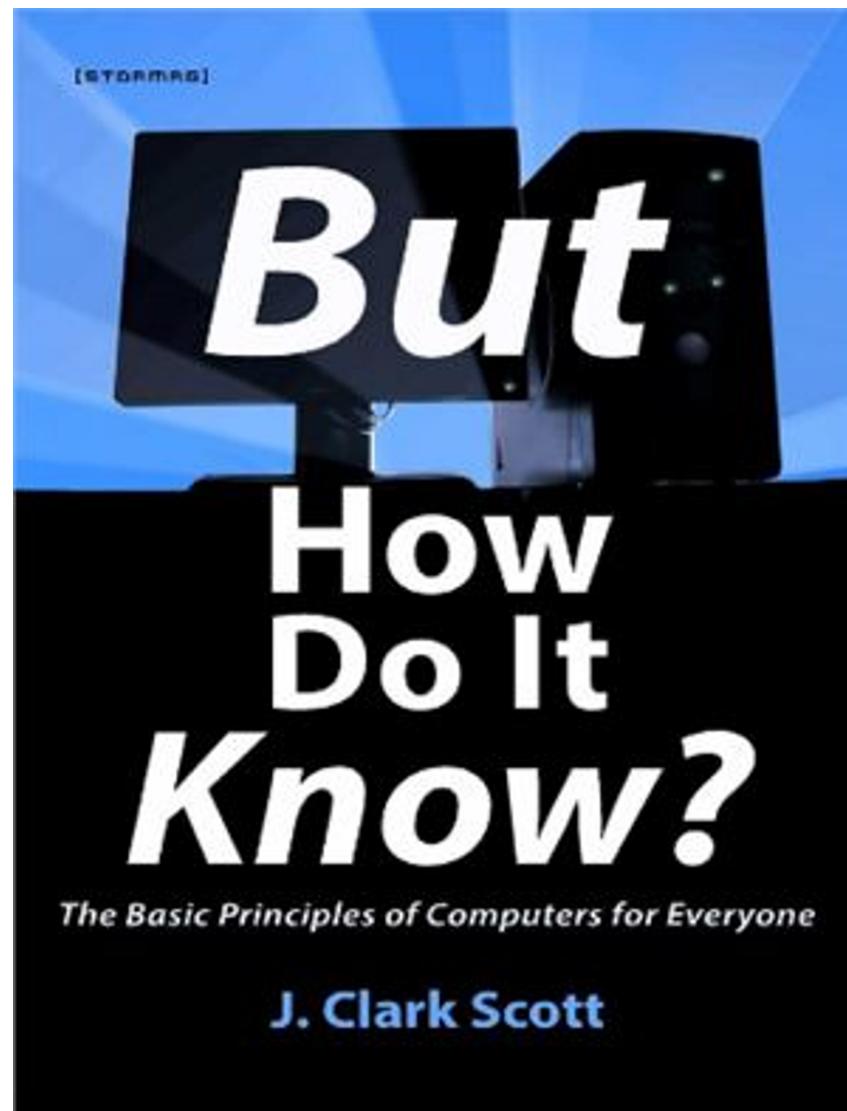


ScottE CPU

Designed by Artur Faltenberg

Original Idea



Overview

- [*https://bitbucket.org/faltenberg/scotte.git*](https://bitbucket.org/faltenberg/scotte.git)
- 8bit RISC CPU
- 18 instructions
- 4 general-purpose registers
- 1KB address space
- performance: 2 clock ticks per instruction
- example: fibonacci(10) in 25s at 16Hz clock speed
- transistors (CMOS): approx. 6000

Registers and Flags

R0

R1

R2

R3

SP

PC

M	H	IX	C	V	S	Z
---	---	----	---	---	---	---

Z: zero flag

S: sign flag

V: overflow flag

C: carry flag

IX: index flags

H: halt flag

M: manual mode

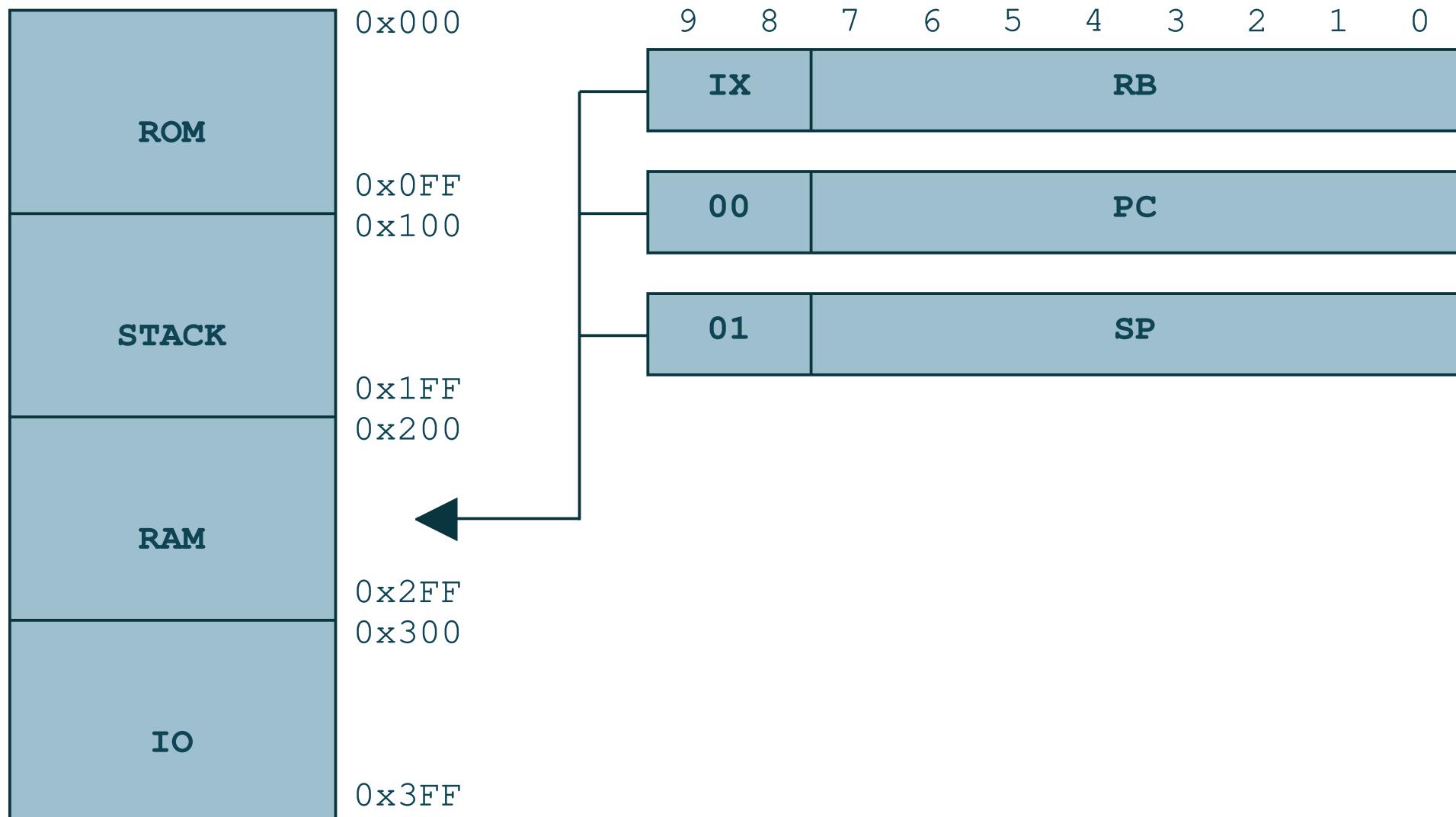
Instruction Set

Instruction	Encoding	Effect
NOP	0000 0x xx	do nothing
SIX ROM STACK RAM IO	0000 10 nn	$ix = nn$
HALT	0000 11 xx	ignore clock
DATA RA, IMM8	0001 ra xx imm[7..0]	$ra = imm$
LDR RA, [RB]	0010 ra rb	$ra = mem[(ix, rb)]$
STR RA, [RB]	0011 ra rb	$mem[(ix, rb)] = ra$
JALR RA	0100 ra rb	$pc = rb; ra = pc+1$
PUSH RA	0110 ra xx	$stack[--sp] = ra$
POP RA	0111 ra xx	$ra = stack[sp++]$
ADD RA, RB	1000 ra rb	$ra = ra + rb$ (cvsz)
SUB RA, RB	1001 ra rb	$ra = ra - rb$ (cvsz)
SHL RA, RB	1010 ra rb	$ra = rb \ll 1$ (c0sz)
SHR RA, RB	1011 ra rb	$ra = rb \gg 1$ (c0sz)
AND RA, RB	1100 ra rb	$ra = ra \& rb$ (00sz)
ORR RA, RB	1101 ra rb	$ra = ra rb$ (00sz)
XOR RA, RB	1110 ra rb	$ra = ra ^ rb$ (00sz)
CMP RA, RB	1111 ra rb	$ra - rb$ (cvsz)

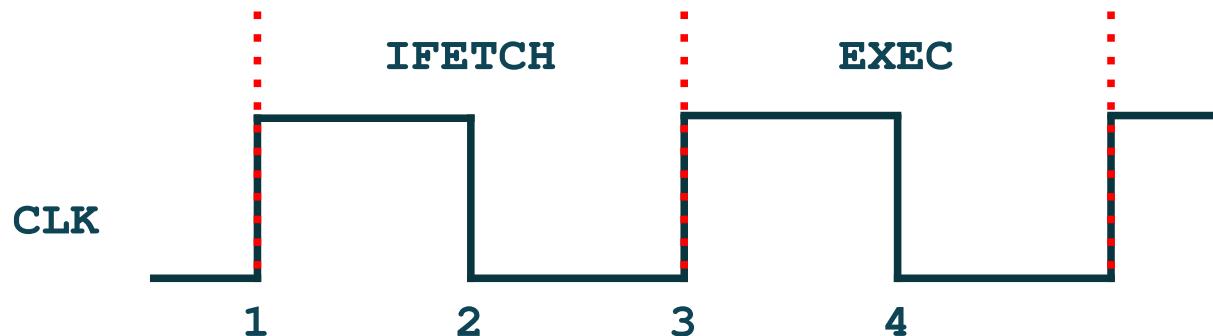
Instruction Set

Instruction	Encoding	Effect
B<COND> IMM8	0101 cond imm[7..0]	pc += cond ? imm : 1
BR IMM8	0101 0000 imm[7..0]	branch always
BZS IMM8	0101 0001 imm[7..0]	branch if fr[z] == 1
BSS IMM8	0101 0010 imm[7..0]	branch if fr[s] == 1
BVS IMM8	0101 0011 imm[7..0]	branch if fr[v] == 1
BCS IMM8	0101 0100 imm[7..0]	branch if fr[c] == 1
BEQ IMM8	0101 0001 imm[7..0]	branch on equal
BLTU IMM8	0101 0100 imm[7..0]	branch on uint <
BGEU IMM8	0101 0101 imm[7..0]	branch on uint >=
BLT IMM8	0101 0110 imm[7..0]	branch on int <
BGE IMM8	0101 0111 imm[7..0]	branch on int >=

Address Space



Clocking



1. update PC
2. update IR, store incremented PC temporarily
3. update PC again if there is an immediate value
4. store result at the destination

Microarchitecture

