

The background of the slide features a dark, moody cityscape at night. In the foreground, several palm trees are silhouetted against the light from the city. The city skyline is filled with numerous skyscrapers of various heights and architectural styles, all illuminated by their own lights. The sky is a deep, dark grey with some faint clouds.

AVR Assembly

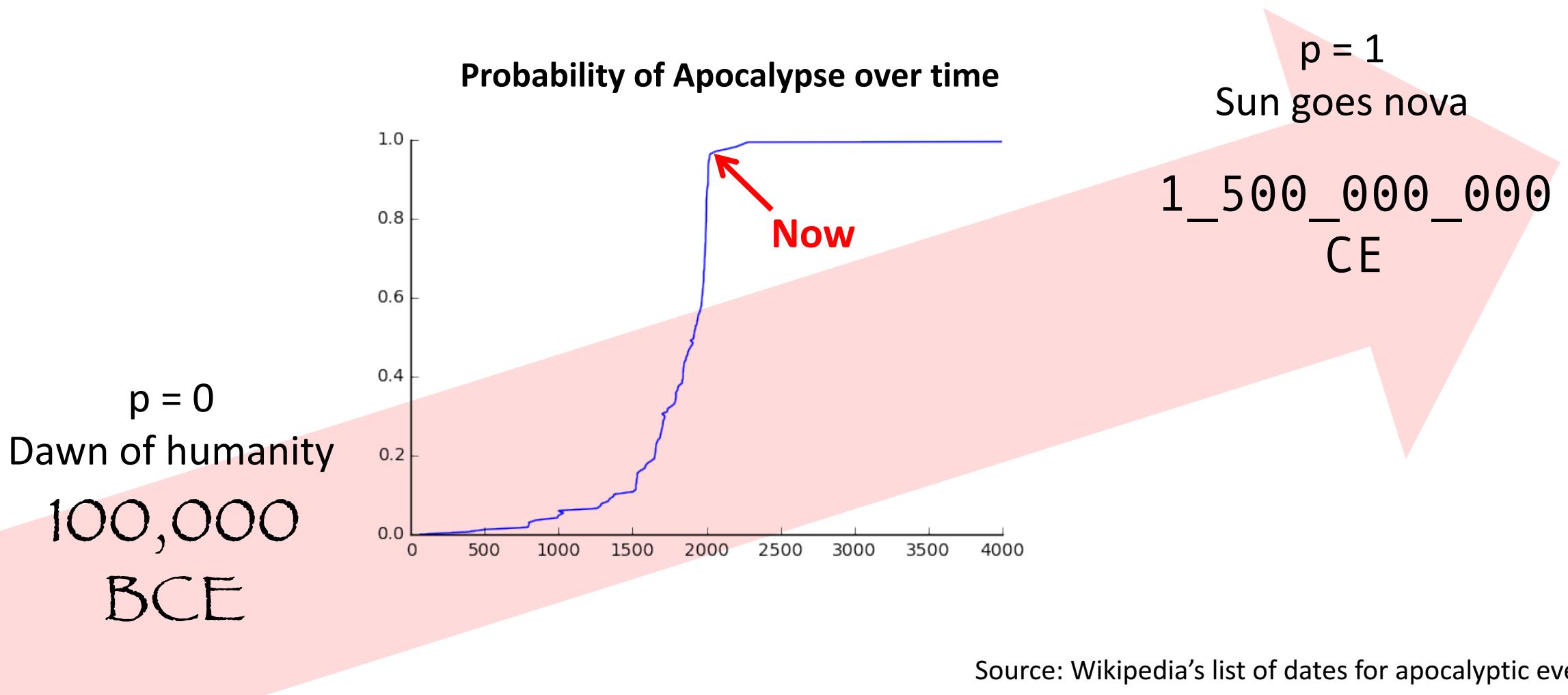
The Ultimate code

Nick Timkovich

ChiPy October 2016 Meeting: Ultimate Language Shootout @ Sully's House, Chicago, IL

Premise

Most of the world will be post-apocalyptic with scarce resources



Premise

Programming machine code for an AVR can be done with minimal resources



Premise

Merriam-Webster, keeper of the American English language, agrees



SINCE 1828

¹ultimate

adjective | ul·ti·mate | \əl-tə-mət\

Full Definition of ULTIMATE

- 1
 - a : most remote in space or time : FARTHEST ✓
 - b : last in a progression or series : FINAL <their *ultimate* destination was Paris> ✓
 - c : EVENTUAL 2 <they hoped for *ultimate* success> ✓
 - d : the best or most extreme of its kind : UTMOST <the *ultimate* sacrifice> ✓
- 2 : arrived at as the last result <the *ultimate* question> ✓
- 3
 - a : BASIC, FUNDAMENTAL <the *ultimate* nature of things — A. N. Whitehead> ✓
 - b : ORIGINAL 1 <the *ultimate* source> ✓
 - c : incapable of further analysis, division, or separation ≈

The Scenario



You are in a post-apocalyptic world and you must make a machine that can tell if a number is divisible by 7 to demonstrate your magic for the local warlord so he doesn't strap you to his bumper.



The Bootstrapping problem

- How do you program a computer in a world without computers?



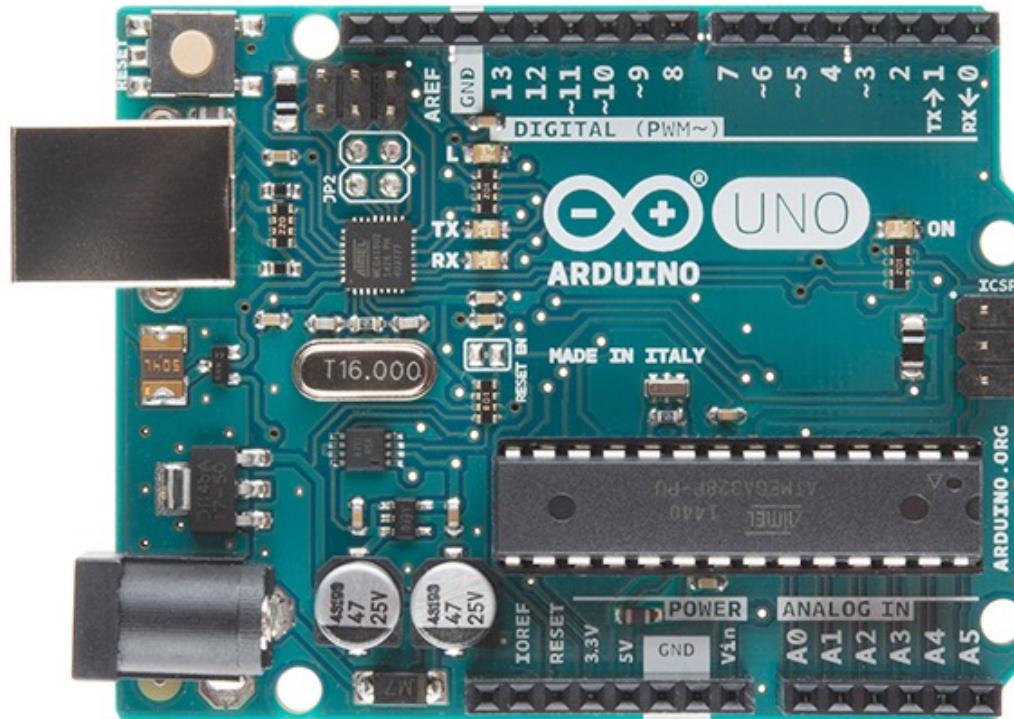
- There are billions of computers in perfect working order, they're in landfills and junkyards.

The background image shows a panoramic view of a city under a hazy, orange-tinted sky. In the center, several tall buildings are emitting thick, dark plumes of smoke from their chimneys. The foreground is filled with sprawling slums and piles of debris. A prominent billboard in the lower center reads "It's Better Than ANYTHING In The World!"

Step 1: Gather Materials

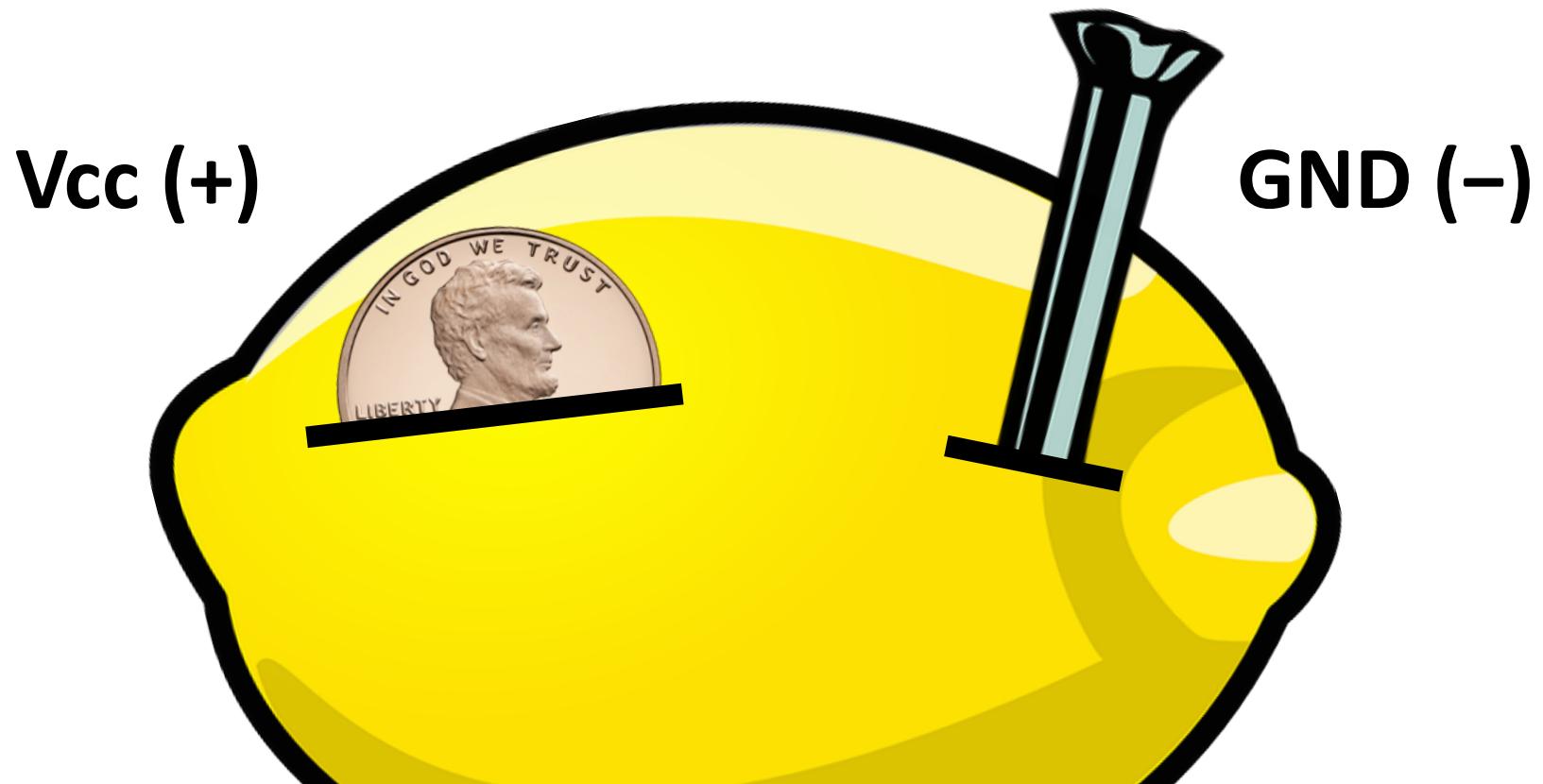
Step 1: Gather materials

- One of the billions and billions of AVRs produced
 - Arduinos mean there are a disproportionate number of DIP packaged AVRs



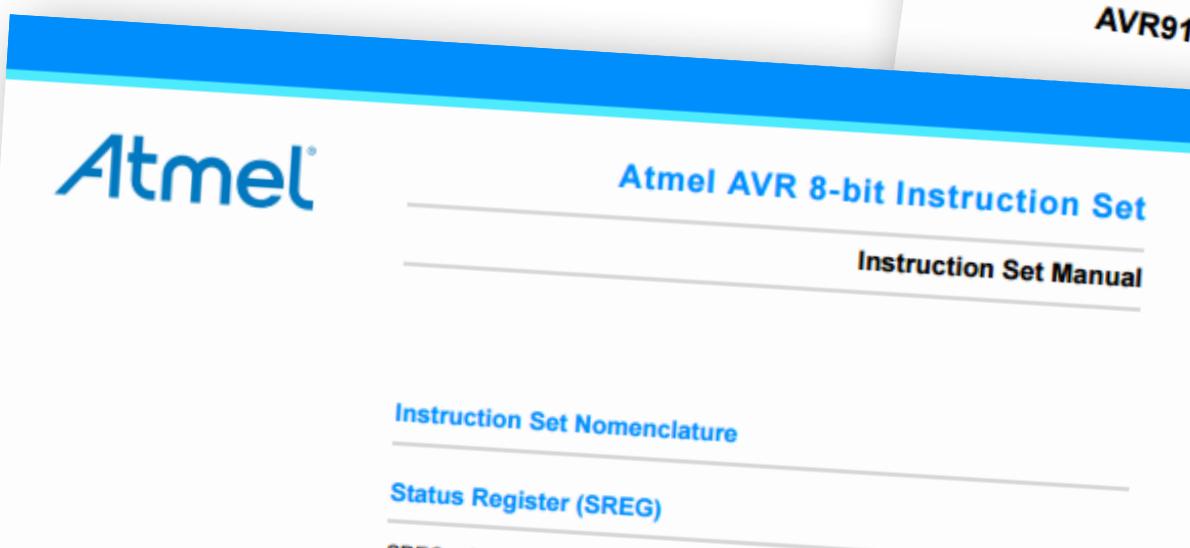
Step 1: Gather materials

- Two different metals
 - one of them should be long & thin
- Electrolytes
 - what plants crave



Step 1: Gather materials

- Reference material you've stashed
- Where to keep copies of it
 - in your go-bag
 - in your cabin in the woods
 - buried outside

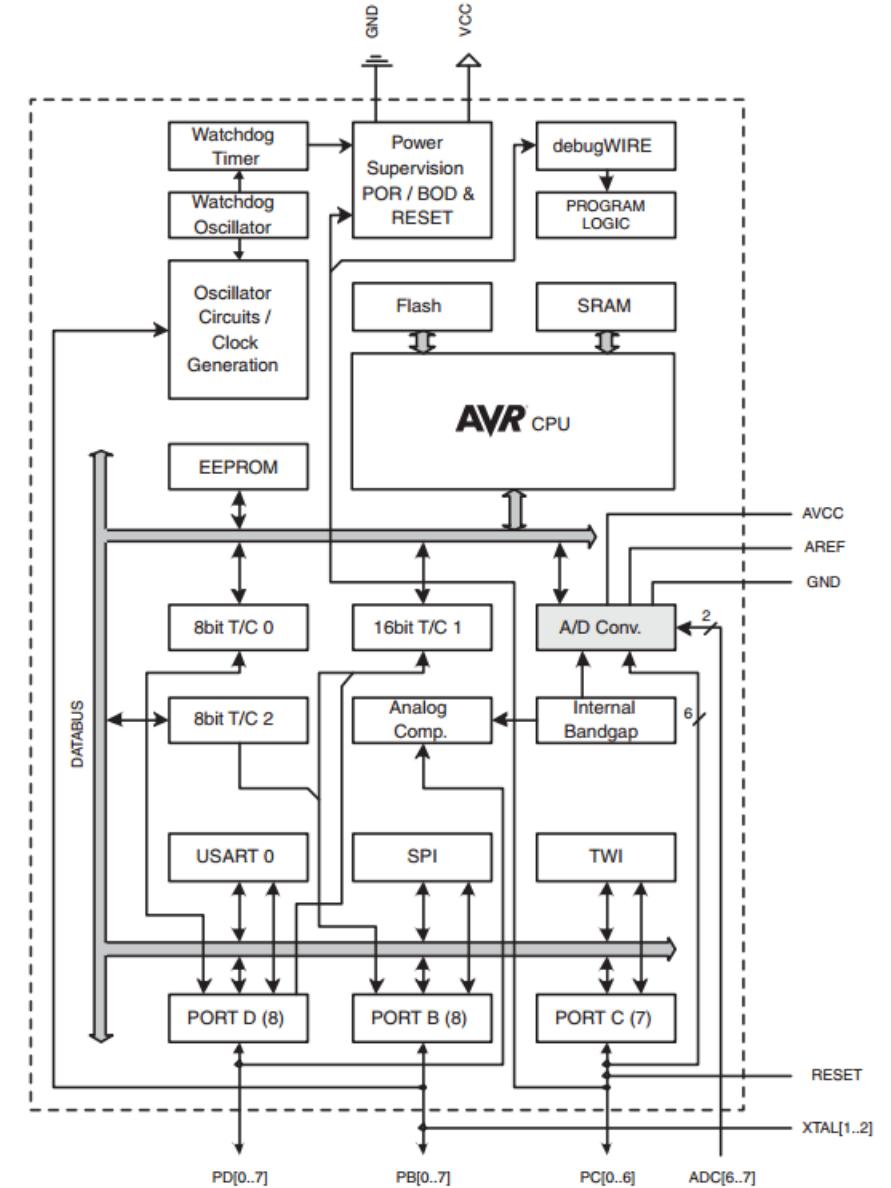


Step 2: Write the Program



Step 2: Create your program

- AVR features:
 - 16-bit instructions
 - Most run within one clock cycle
 - (32) 8-bit general-purpose registers
 - Modified Harvard Architecture



Step 2: Create your program

- Spec
 - When a switch is pressed, read a 16-bit number from input pins
 - If divisible by 7, make a noise (oscillate a pin), otherwise be silent

Step 2: Create your program

- Loop and wait for switch on Port C (0x08), Pin 0 to go low
- A “while” loop

```
SBIS $08, 0
RJMP -2
; prog...
```

101. SBIS – Skip if Bit in I/O Register is Set

101.1 Description

This instruction tests a single bit in an I/O Register and skips the next instruction if the bit is set. This instruction operates on the lower 32 I/O Registers – addresses 0-31.

Operation: If I/O(A,
Syntax: SBIS A,
16-bit O
1001
101.2 Status Register

94. RJMP – Relative Jump

94.1 Description

Relative jump to an address within PC - 2K +1 and PC + 2K (words). For AVR microcontrollers with Program memory not exceeding 4K words (8KB) this instruction can address the entire memory from every address location. See also JMP.

I T
- -
Example: waitset: sk
rj
nc
Operation: PC ← PC + k + 1
Syntax: RJMP k
Operands: -2K ≤ k < 2K
16-bit Opcode: 1100 1111 1111 1111
Program Counter: PC ← PC + k + 1
Stack: Unchanged

Step 2: Create your program

- Read ports into register space
- Port B & D (0x05 & 0x0B, respectively) are usually the widest

```
IN r20, $05  
IN r21, $0B
```

64. IN - Load an I/O Location to Register

64.1 Description
Loads data from the I/O Space (Ports, Timers, Configuration Registers, etc.) into register Rd in the Register File.

Operation:
(i) $Rd \leftarrow I/O(A)$

Syntax: (i) IN Rd,A **Operands:** $0 \leq d \leq 31, 0 \leq A \leq 63$ **Program Counter:** $PC \leftarrow PC + 1$

16-bit Opcode:

1011	0Aad	dddd	AAAA
------	------	------	------

64.2 Status Register (SREG) and Boolean Formula

I	T	H	S	V	N	Z	C
-	-	-	-	-	-	-	-

Step 2: Create your program

- Divide? Woah, woah, don't make it complicated.

```
SUBI r20, 7
SBCI r21, 0
BRMI 1000 → X
; fail
BRNE -4
TST r20
BRNE -6
; success!
```

124. SUBI – Subtract Immediate

124.1 Description

Subtracts a register and a constant and places the result in the destination register Rd. This instruction is working on Register R16 to R31 and is very well suited for operations on the X, Y, and Z-pointers.

124.2 Status Register

I	T
-	-

H: Rd3 • Set if true

S: N ⊕ V

V: Rd7 • Set if true

3. Conditional Branch Summary

Test	Boolean	Mnemonic	Complementary	Boolean	Mnemonic	Comment
Rd > Rr	Z•(N ⊕ V) = 0	BRLT ⁽¹⁾	Rd ≤ Rr	Z+(N ⊕ V) = 1	BRGE*	Signed
Rd ≥ Rr	(N ⊕ V) = 0	BRGE	Rd < Rr	(N ⊕ V) = 1	BRLT	Signed
Rd = Rr	Z = 1	BREQ	Rd ≠ Rr	Z = 0	BRNE	Signed
Rd ≤ Rr	Z+(N ⊕ V) = 1	BRGE ⁽¹⁾	Rd > Rr	Z•(N ⊕ V) = 0	BRLT*	Signed
Rd < Rr	(N ⊕ V) = 1	BRLT	Rd ≥ Rr	(N ⊕ V) = 0	BRGE	Signed
Rd > Rr	C + Z = 0	BRLO ⁽¹⁾	Rd ≤ Rr	C + Z = 1	BRSH*	Unsigned
Rd ≥ Rr	C = 0	BRSH/BRCC	Rd < Rr	C = 1	BRLO/BRCS	Unsigned
Rd = Rr	Z = 1	BREQ	Rd ≠ Rr	Z = 0	BRNE	Unsigned
Rd ≤ Rr	C + Z = 1	BRSH ⁽¹⁾	Rd > Rr	C + Z = 0	BRLO*	Unsigned
Rd < Rr	C = 1	BRLO/BRCS	Rd ≥ Rr	C = 0	BRSH/BRCC	Unsigned

Step 2: Create your program

- Set Port C, Pin 1 to Output (DDRC = 0x07)
- Oscillate pin.
- Use NO-OPs to tune frequency.

SBI \$07, 1
SBI \$08, 1
NOP
NOP
CBI \$08, 1
RJMP -5

The screenshot shows a portion of a microcontroller instruction manual. At the top, there is a blue header bar. Below it, the page content starts with the heading "99. SBI – Set Bit in I/O Register". Under this heading, there is a sub-section "99.1 Description" which states: "Sets a specified bit in an I/O Register. This instruction operates on the lower 32 I/O Registers – addresses 0-31." To the left of the text, there is some partially visible assembly code and a blue arrow pointing from the text area towards the assembly code. Below the SBI section, there is another section titled "85. NOP – No Operation". The "85.1 Description" part of this section states: "This instruction performs a single cycle No Operation." There is also a small diagram of a microcontroller chip with various pins labeled.

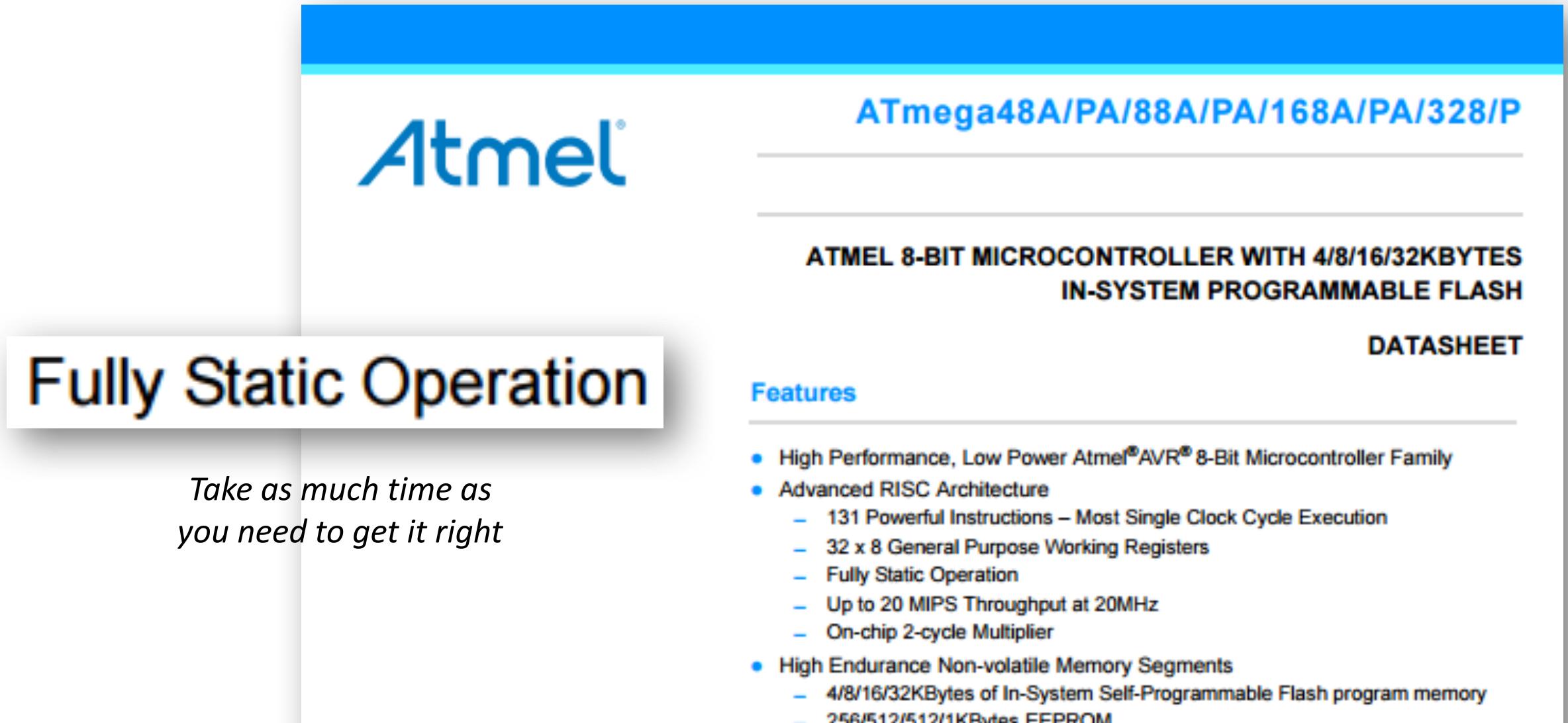
The background of the image is a dark, atmospheric scene from an anime. On the right side, a character wearing a yellow and black suit with a red cape is seen from behind, looking out over a city skyline at night. The city lights are visible through the haze, and the sky is filled with clouds. The overall mood is mysterious and intense.

Step 3:
Flash the Micro

Step 3: Flash the microcontroller

- Make a battery (3-4 cells of most junk should be OK)
- Physically toggle wires between power and ground
- Consult AVR Reference Manuals and App Notes in your Go Bag

Step 3: Flash the microcontroller



The image shows the first page of the Atmel ATmega48A/PA/88A/PA/168A/PA/328/P datasheet. The page has a blue header bar. In the top right corner, the part number "ATmega48A/PA/88A/PA/168A/PA/328/P" is printed in blue. The Atmel logo is on the left. Below the header, the title "ATMEL 8-BIT MICROCONTROLLER WITH 4/8/16/32KBYTES IN-SYSTEM PROGRAMMABLE FLASH" is centered. To the right of the title is the word "DATASHEET". On the left side, there is a large callout box with the text "Fully Static Operation" in bold. Below this, a quote says "Take as much time as you need to get it right". The main content area contains a "Features" section with a bulleted list of the microcontroller's capabilities.

ATmega48A/PA/88A/PA/168A/PA/328/P

**ATMEL 8-BIT MICROCONTROLLER WITH 4/8/16/32KBYTES
IN-SYSTEM PROGRAMMABLE FLASH**

DATASHEET

Fully Static Operation

*Take as much time as
you need to get it right*

Features

- High Performance, Low Power Atmel® AVR® 8-Bit Microcontroller Family
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4/8/16/32KBytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1KBytes EEPROM

Step 3: Flash the microcontroller

- Make a battery (3-4 cells of most junk should be OK)
- Physically toggle wires between power and ground
- Consult AVR Reference Manuals and App Notes in your Go Bag

Average state transitions: 5x size of program in bits

Takeaways

- Print off the AVR Reference Manuals & App Notes, put them in your Bug-Out Bag
 - Atmel AVR 8-bit Instruction Set
 - AVR200: Multiply and Divide Routines
 - AVR910: In-System Programming
- AVR Assembly is the Ultimate language



Sources

All Rights Reserved their respective owners
Declared as Fair use under 17 U.S.C. § 107

Freely Licensed

Atmel Corporation, Appl. Note "AVR200: Multiply and Divide Routines". Rev. 0936D-avr-09/09

Atmel Corporation, Appl. Note "AVR910: In-System Programming". Rev. 0943E-avr-08/08

Atmel Corporation, "Atmel 8-bit Microcontroller with 4/8/16/32KBytes In-System Programmable Flash"
ATmega48A/PA/88A/PA/168A/PA/328/P Datasheet. Rev 8271J, 11/2015

Atmel Staff, *Atmel AVR 8-bit Instruction Set*. Atmel Corporation. Rev. 0856K-avr-05/2016

Idiocracy. Directed by Mike Judge. Performed by Luke Wilson, Maya Rudolph. United States: Twentieth Century Fox Film Corp., 2007. <http://www.imdb.com/title/tt0387808>

Justice League: The Flashpoint Paradox. Directed by Jay Oliva. Performed by Justin Chambers, C. Thomas Howell.
United States: DC Entertainment, 2013. <http://www.imdb.com/title/tt2820466>

Life After People. Directed by David de Vries. Performed by James Lurie. United States: Flight 33 Productions, 2008.
<http://www.imdb.com/title/tt1173907>

Mad Max: Fury Road. Directed by George Miller. Performed by Tom Hardy, Charlize Theron, Nicholas Hoult.
Australia: Warner Brothers Pictures, 2015. <http://www.imdb.com/title/tt1392190>

The Matrix Revolutions. Directed by The Wachowskis. Performed by Keanu Reeves, Laurence Fishburne, Carrie-Anne Moss. United States: Warner Bros., 2003. <http://www.imdb.com/title/tt0242653>

The Office. Performed by Rainn Wilson, John Krasinski, Jenna Fischer. United States: Reveille Productions, 2005-2013. Created by Greg Daniels, Ricky Gervais, Stephen Merchant. <http://www.imdb.com/title/tt0386676>

Turbo Kid. Directed by François Simard and Anouk Whissell. Performed by Munro Chambers, Laurence Leboeuf, Michael Ironside. Canada: EMA Films, 2015. <http://www.imdb.com/title/tt3672742>

WALL-E. Directed by Andrew Stanton. Performed by Ben Burtt, Elissa Knight, Jeff Garlin. United States: Walt Disney Pictures, 2008. <http://www.imdb.com/title/tt0910970>

Pixabay. "Lemon fruit yellow food citrus 25244."

Accessed October 15, 2016.

<https://pixabay.com/en/lemon-fruit-yellow-food-citrus-25244/> CC0

Pixabay. "Screws screw nail repair rivet 24244."

Accessed October 15, 2016.

<https://pixabay.com/en/screws-screw-nail-repair-rivet-24244/> CC0

Red Cross. 37173 - Red Cross "ready to Go"

Preparedness Kit. March 7, 2006. FEMA.

In Wikimedia Commons. October 12, 2009.

Accessed October 14, 2016. **Public Domain**.

https://commons.wikimedia.org/wiki/File:FE_MA_-_37173_-_Red_Cross_"ready_to_go",_preparedness_kit.jpg

SparkFun Electronics. "Arduino Uno - R3." SparkFun Electronics Shop. Accessed October 14, 2016.

<https://www.sparkfun.com/products/11021>.

Images CC BY-NC-SA 3.0

United States Mint. "United States One Cent Obverse" February 10, 2014. United States Department of the Treasury. In Wikimedia Commons.
Accessed October 15, 2016. **Public Domain**.

https://commons.wikimedia.org/wiki/File:US_One_Cent_Obv.png