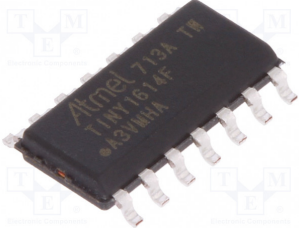

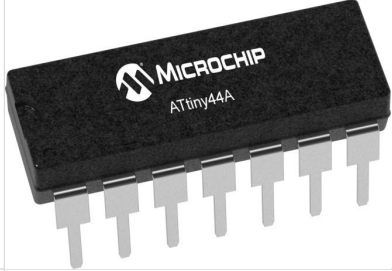


Person	hyejin	Harm	nathan & tessell
Board	attiny1614	ESP8266	ATtiny 44A
Picture			
Link to datasheet	<a href="#">ATtiny1614 datasheet</a>	<a href="#">Datasheet ESP8266</a>	<a href="#">Datasheet Attiny44a</a>
<b>Specs</b>			
Working Voltage	min: 1.8, typical: 3, max: 5.5	Min: 2.5 Typical: 3.3 Max: 3.6	1.8 - 5.5 V
Input/output voltage	min: 0.2, max: working voltage - 0.2	3.3 volt	Vil: when VCC = 1.8V - 2.4V then Vil is 0.2Vcc max. When VCC is 2.4 - 5.5V Vil is 0.3V max. Vih: VCC is 1.8 V - 2.4V then Vih is 0.7 Vcc - Vcc+ 0.5 When VCC = 2.4 V -5.5V then Vih is 0.7Vcc - Vcc+0.5. Vol: 0.6V maximum. Voh: 4.3V minimum.
Power consumption	active: 10.2mA, Idle: 4.3mA (20MHz, 5V)	Typical: 170 mA	Active mode: 210 µA at 1.8V and 1 MHz; Idle Mode: 33 µA at 1.8V and 1 MHz; Power-down Mode: 0.1 µA at 1.8V and 25°C.
Chip architecture	Harvard architecture	Tensilica L106 32-bit RISC processor	RISC Architecture
(clock)speed	16/20 MHz Oscillator	From 24 MHz to 52 MHz	4MHZ - 20 MHz range. 8MHZ is the factory setting. Settings related to voltage: 0 – 4 MHz @ 1.8 – 5.5V. 0 – 10 MHz @ 2.7 – 5.5V. 0 – 20 MHz @ 4.5 – 5.5V. See [hackster.io](https://www.hackster.io/porrey/attiny-16mhz-610d41) on how to set the clock.
Memory	Flash(16KB), EEPROM(256bytes), SRAM(2KB)	SRAM and ROM, external SPI flash to store prams	4KB Flash Program Memory; 256 Bytes of internal EEPROM; 256 Bytes of internal SRAM
Pins (type and amount)	14 pins		14 Pins Type: VCC, GND & 12 I/O pins.
Special features	16KB of flash memory	<ul style="list-style-type: none"> <li>• 2.4 GHz receiver - WiFi</li> <li>• 2.4 GHz transmitter - WiFi</li> </ul> TCP/IP and full 802.11 b/g/n WLAN MAC protocol	
<b>## Workflow</b>			
Connect to your computer			
How do you program the device:			

- programming environment			<p>The tiny is an AVR microcontroller. This is a family of In-System Programming microcontrollers that can be programmed while already soldered on the board. To program it you need an ISP programmer. We use the TinyISP we made in week 04. But you can also [use an Arduino](<a href="https://www.arduino.cc/en/tutorial/arduinoISP">https://www.arduino.cc/en/tutorial/arduinoISP</a>) for instance. Connect the TinyISP to the Hello World Board and to your computer. Run [AVRdude](<a href="https://www.nongnu.org/avrdude/">https://www.nongnu.org/avrdude/</a>) which is a utility to download/upload/manipulate the ROM and EEPROM contents of AVR microcontrollers using the in-system programming technique.</p>
- programming language			C
- how do you get code on the device			<p>1. Make: write program in C. Create a MAKEfile. Run AVRdude flash command 2. Arduino: Run the Arduino software and select the Attiny44 and tinyISP programmer in the `tools` dropdown menu. Upload code to the board with the upload button.</p>