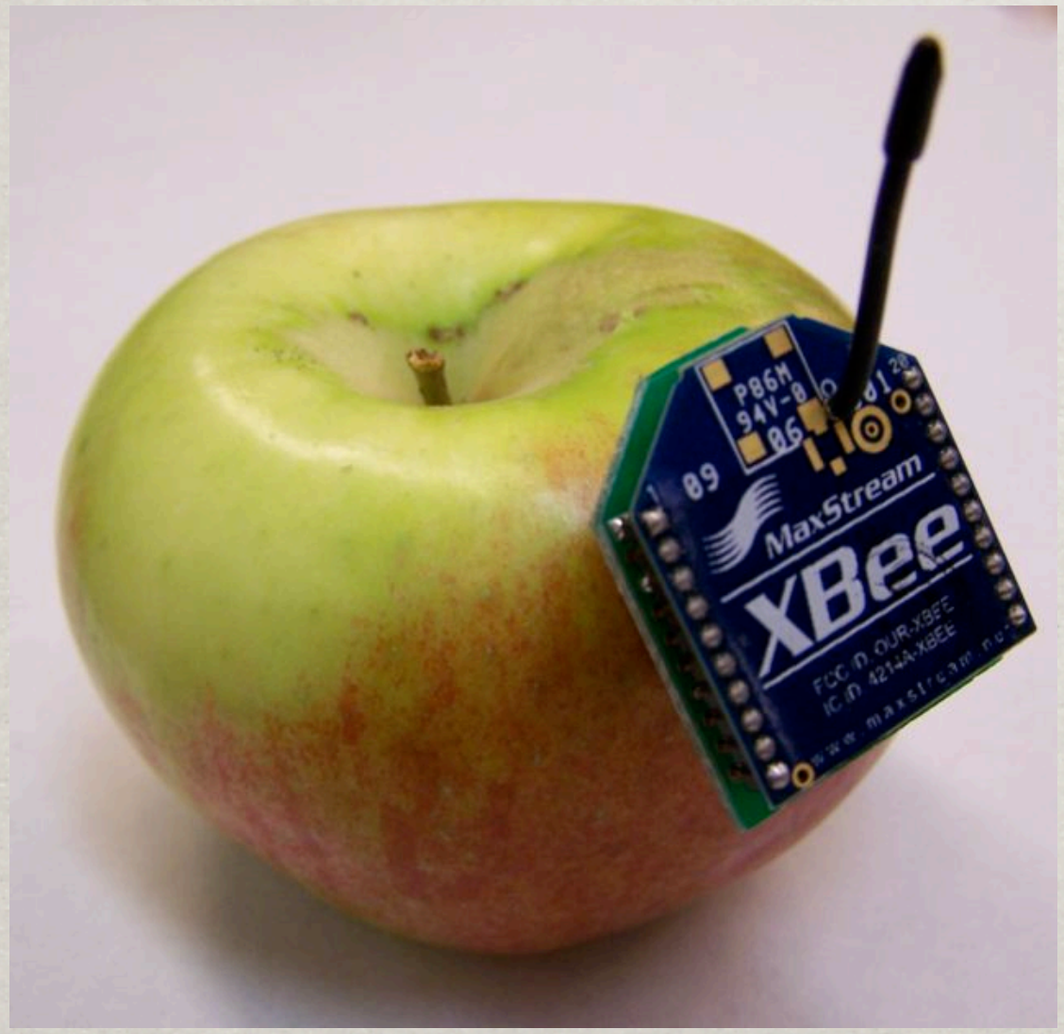
DriveBy:
ZigBee Wireless

Thursday, October 12th, 9:15 pm
Room 447



Discover the joy of moving data wirelessly using ZigBee radios. You'll learn how to install and configure XBee brand radios to link up your brilliant PComp projects. We'll compare ZigBee with other systems like Bluetooth, XPort, basic RF and more.

Applications for wearables, toys, network objects, artworks, performances and anything-that-spins will be discussed. If there's time, we'll take a peek at some of the cool advanced features.

The perfect prelude to TNO.

XBee Radios

ZigBee & More

Presented By:
Rob Faludi

"Don't tell me about the technology. Tell me about the application"

— Tom

What's the Application?

Moving Data by Radio

toys	wearables	performance
portables	emergent systems	anything spinning
network objects	sensors	audio/video
feedback	network effects	context awareness

Your Projects

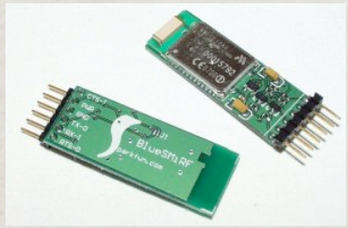
What Do We Want?

wireless	easy communication	reliability
low power	addressing	broadcast
small	standardized	cheap
bandwidth	fast	routing

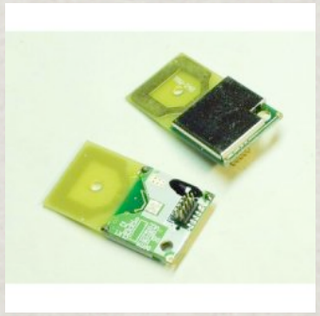
Tell Me About the Technology!

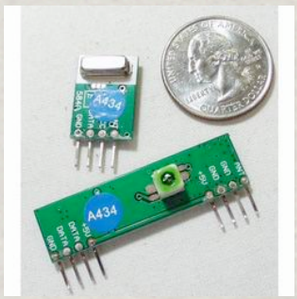
Existing Methods

- Bluetooth



- "RF"

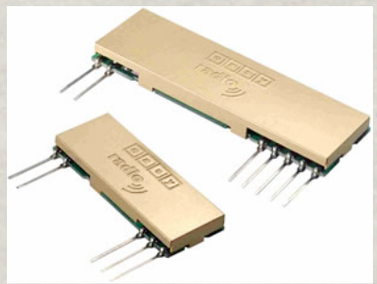




- XPort TCP/IP

- WiPort TCP/IP

- EZ Radio



- Cell Phone Data GPRS

ZigBee & 802.15.4

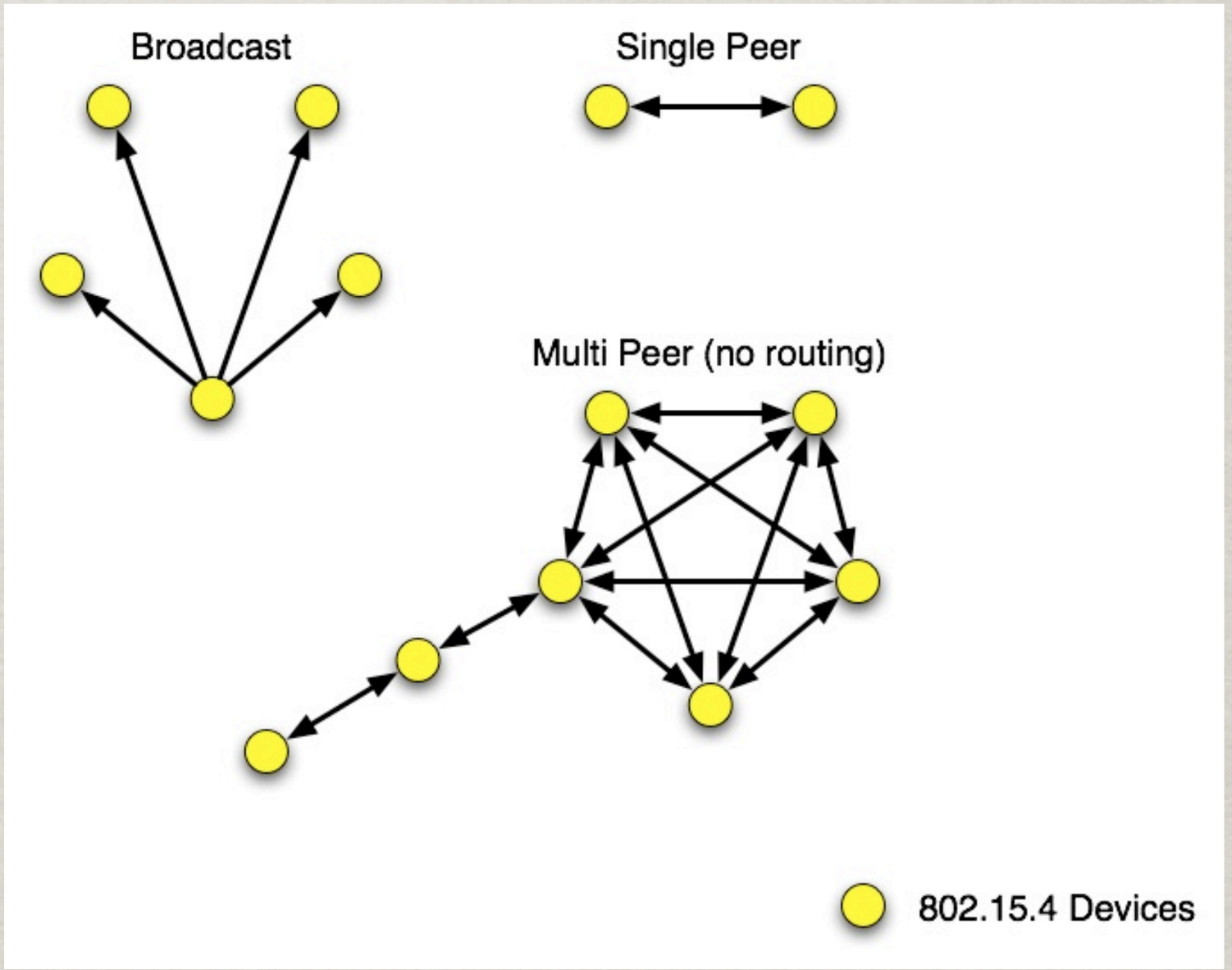
- ZigBee is built on top of the IEEE 802.15.4 protocol
- XBee radios can be configured with or without ZigBee
- Both ways are useful

802.15.4

- low power
- addressing
- cheap
- wireless
- small
- standardized

802.15.4 Topologies

- single peer
- multi-peer
- broadcast

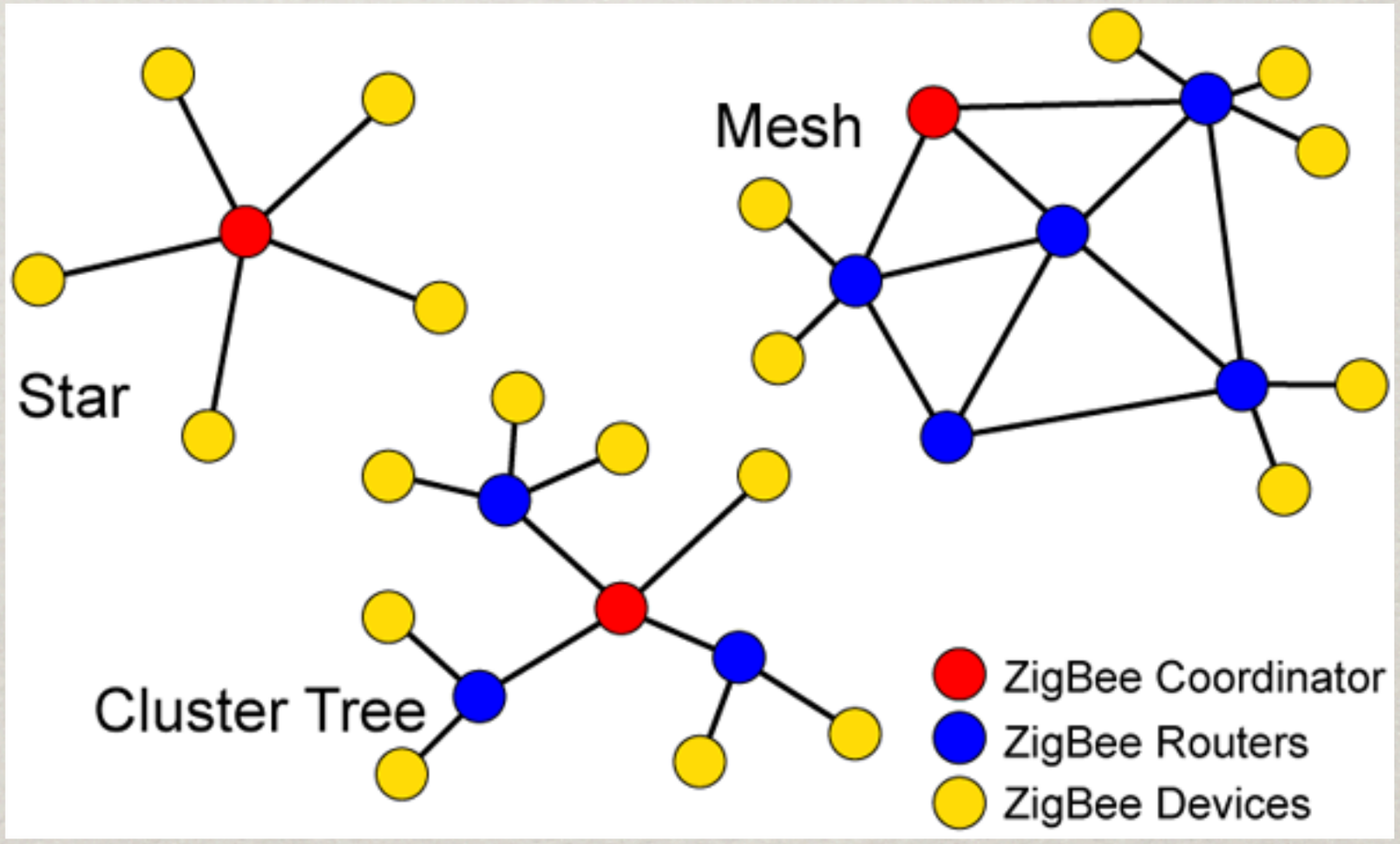


ZigBee

- routing
- self-healing mesh
- ad-hoc network creation

ZigBee Topologies

- peer
- star
- mesh
- routing



How Do I Make One?

Materials

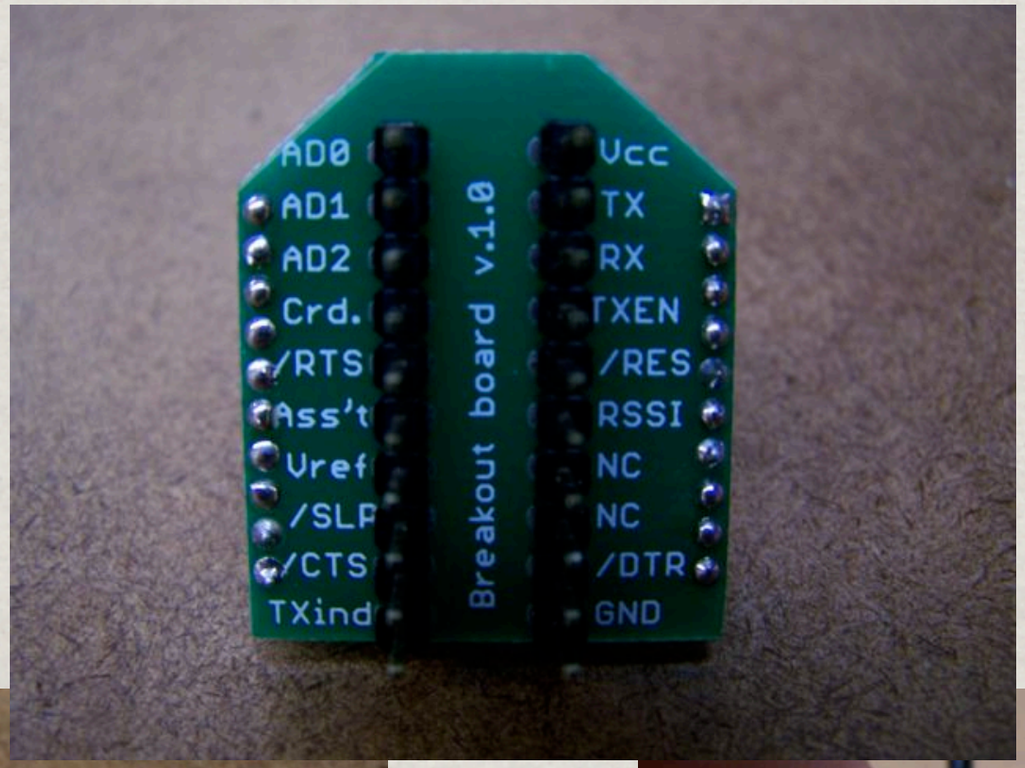
- XBee OEM Module (30-100 m range) $19
  XBee Pro (100m - 1.6 km range) $32

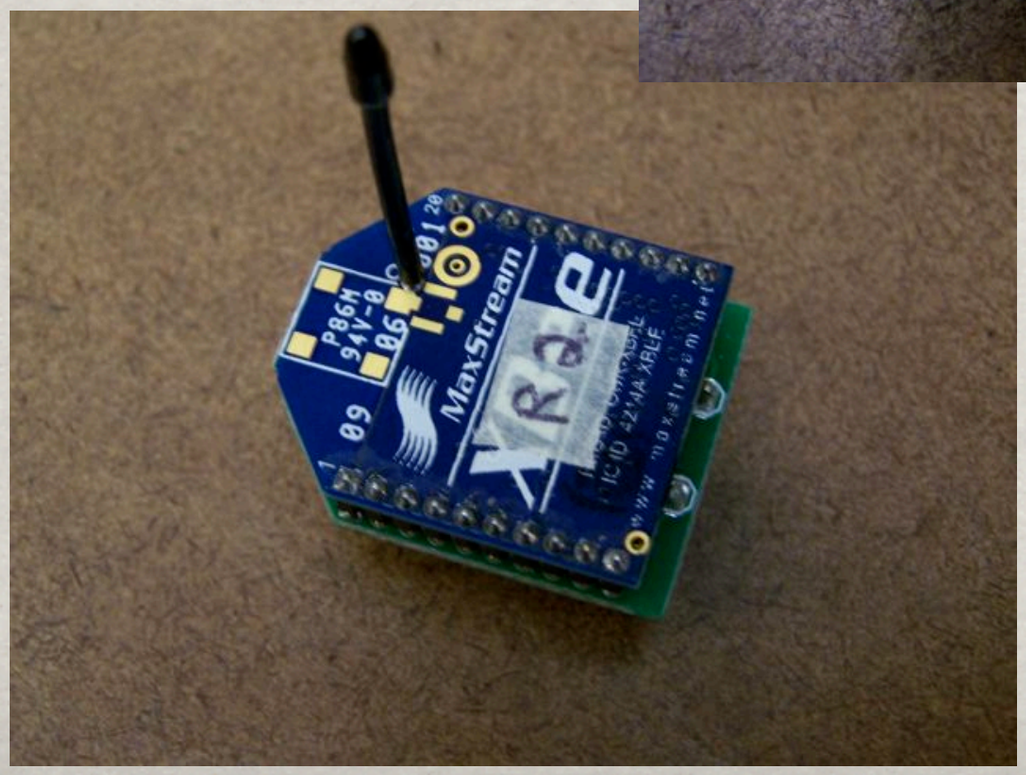
  - MaxStream: http://www.maxstream.net

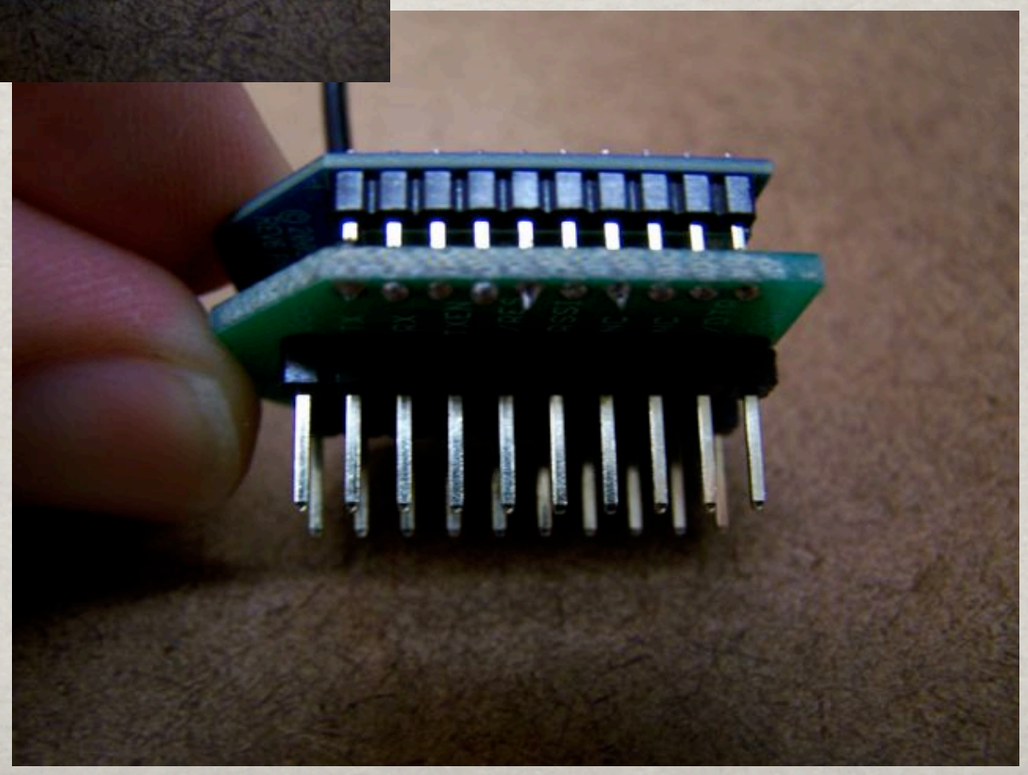
- Breakout Board, 2mm to 10 mil pin spacing. $5 special order

  - BatchPCB: http://www.batchpcb.net
    Files: http://rob.faludi.com/itp/xbee_breakout_10_gerb.zip

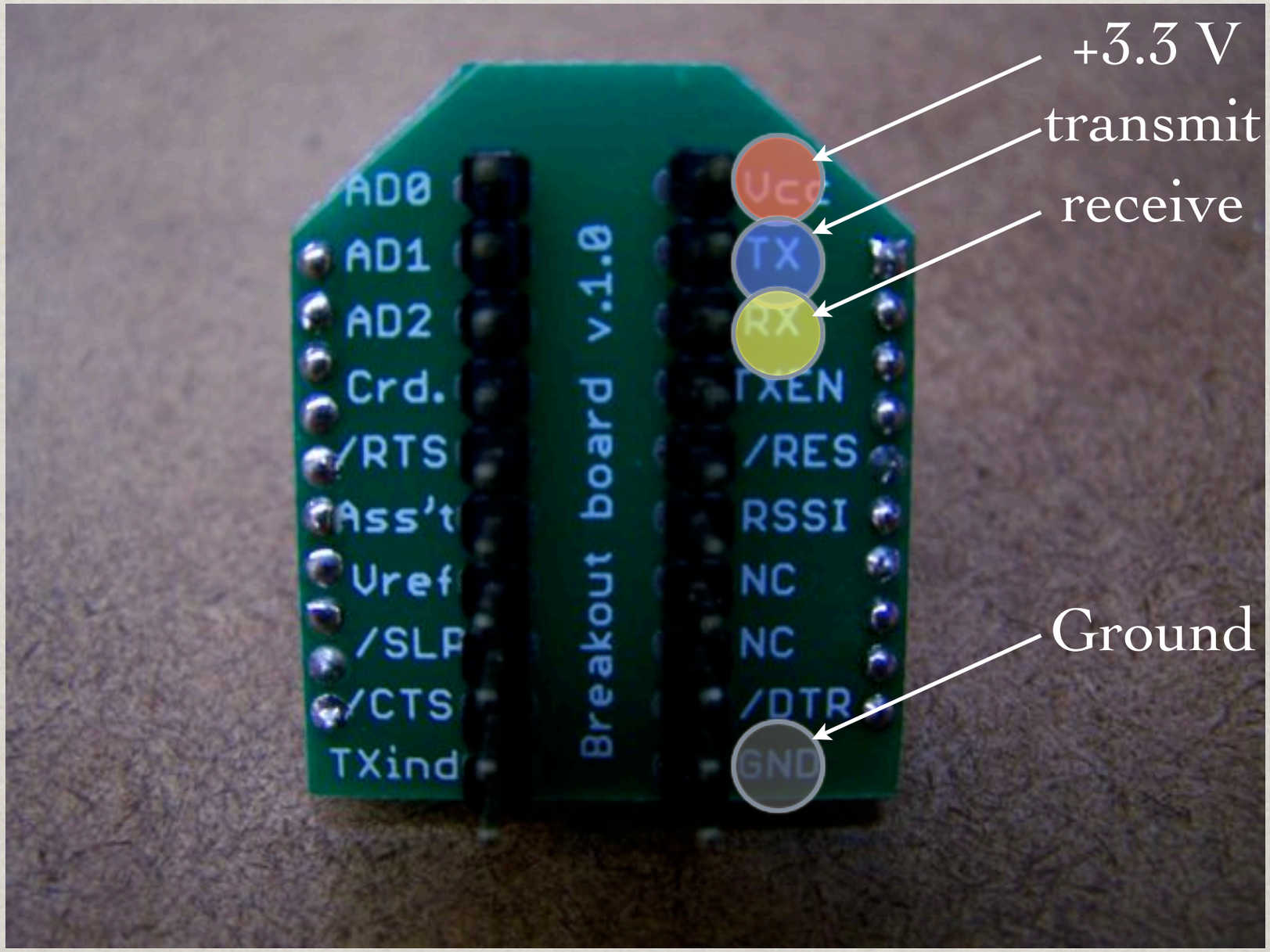
XBee with Breakout Board







Wiring


+3.3 V
transmit
receive
Ground
AD0
AD1
AD2
Crd.
/RTS
Ass't
Vref
/SLP
/CTS
TXind
v.1.0
Breakout board
Vcc
TX
RX
TXEN
/RES
RSSI
NC
NC
/DTR
GND

Remember!

- Use only +3.3 Volts. The regulator usually has a different pin arrangement: G-O-I
- Always use decoupling capacitors. The radios often don't work without them.
- XBee TX goes to Arduino RX and vice versa.
- PIC and Arduino both can run on 3.3 Volts

Instructions

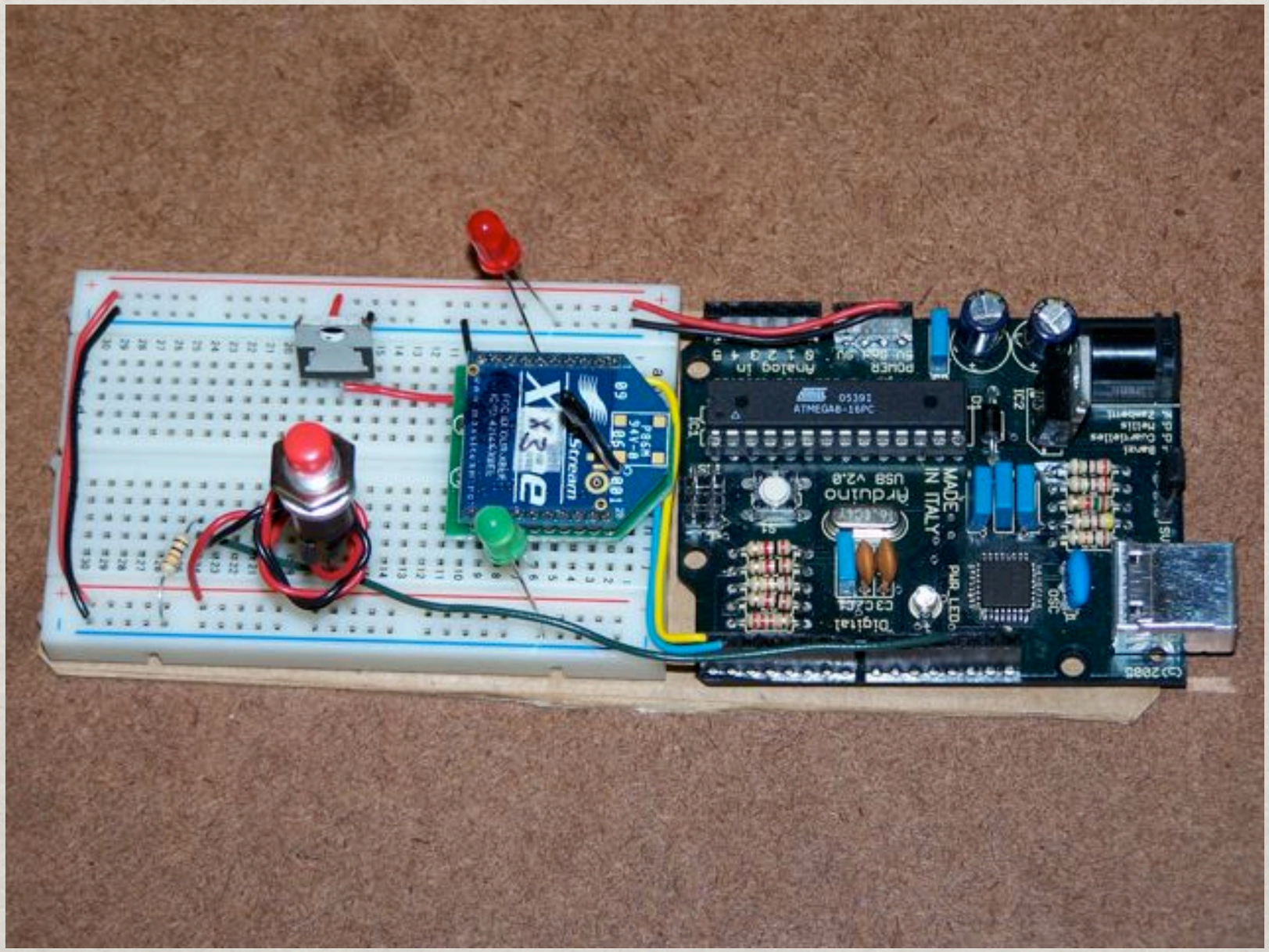
- XBee Practical Example: Paired communication between two microcontrollers. Includes building, wiring and code for PIC and Arduino

  - http://itp.faludi.com/meshnetworking/XBee/XBee_example.html

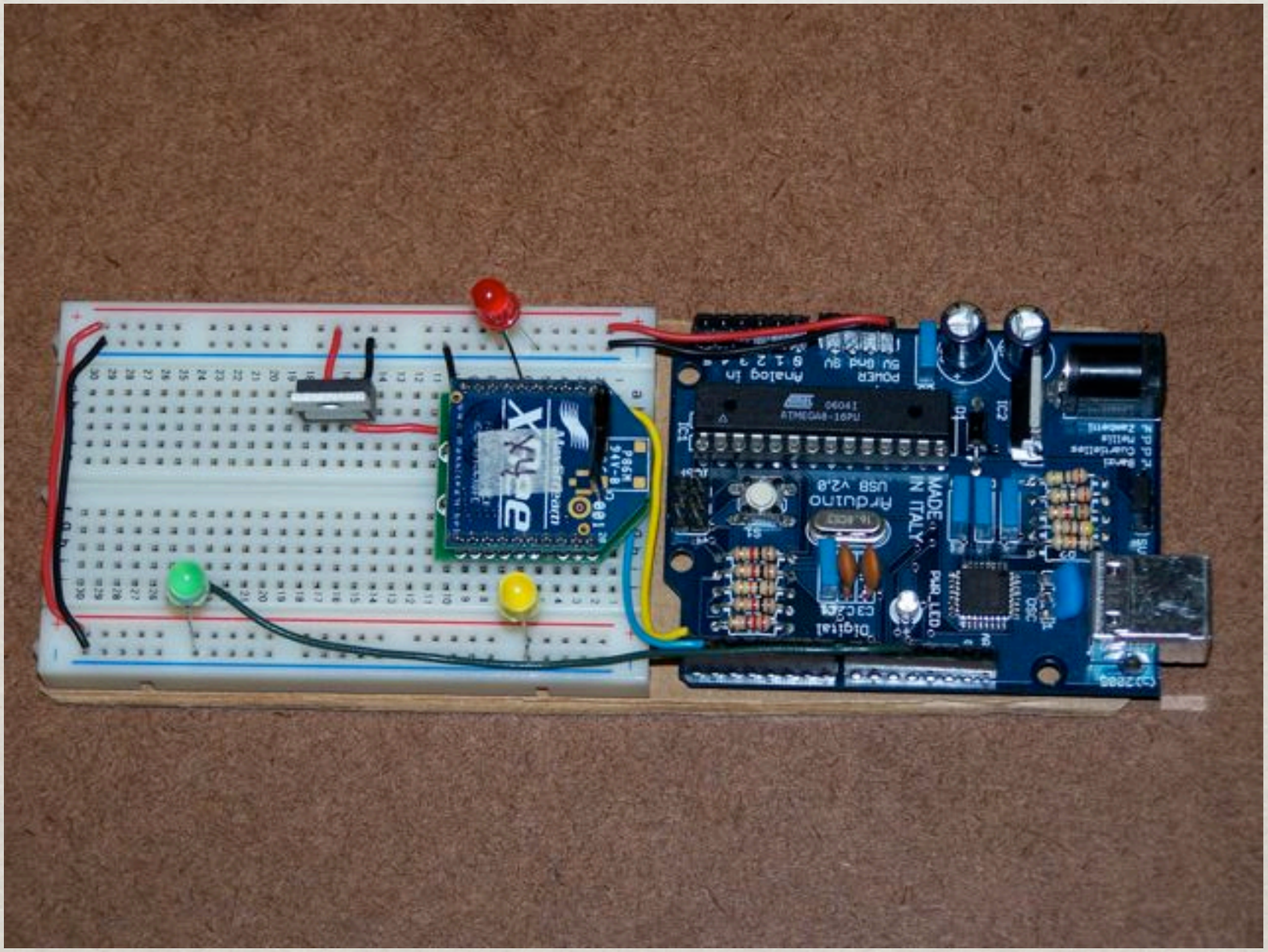
Demo:

Blink that Light

XBee Send Example



XBee Receive Example

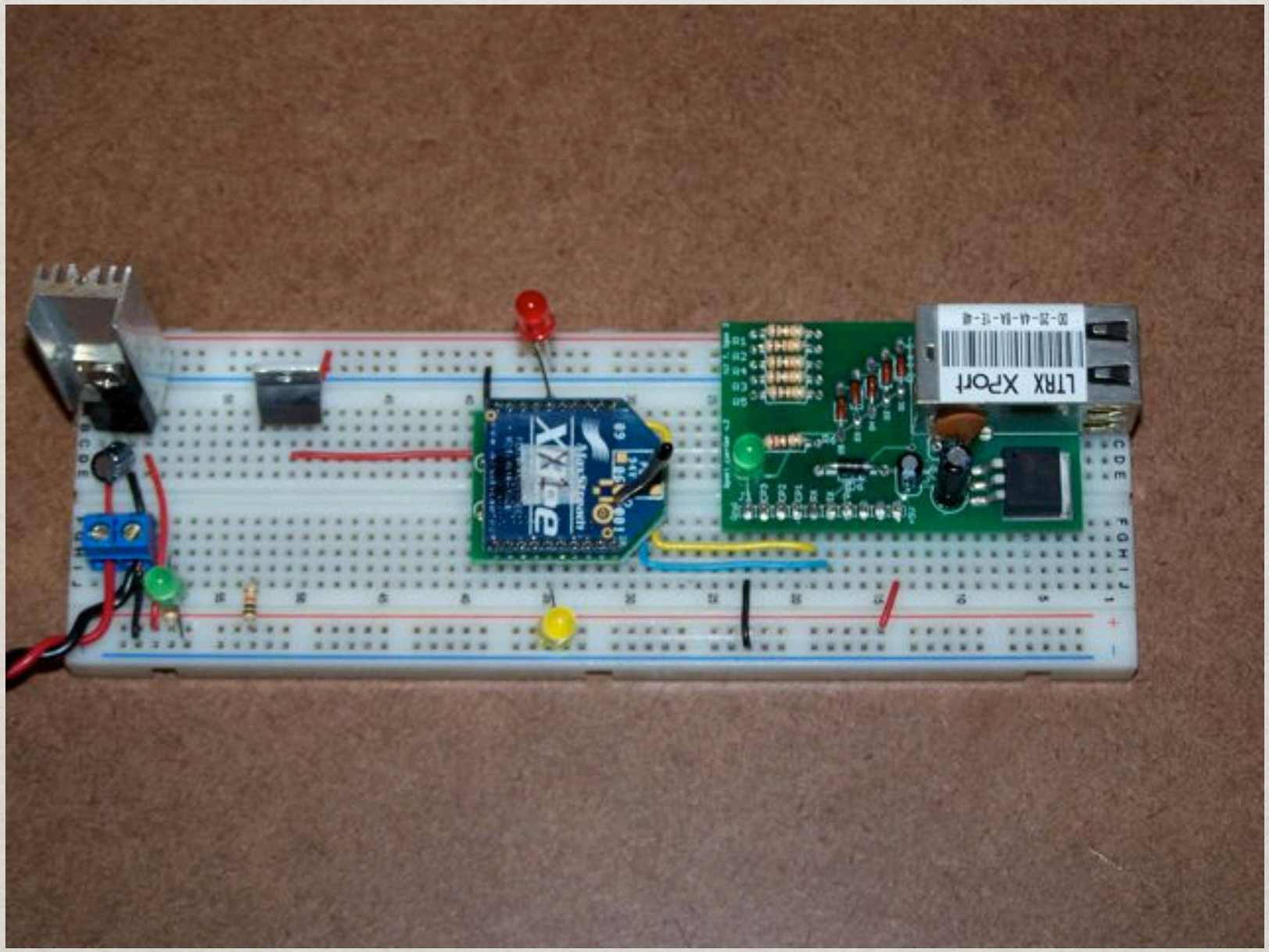


More Applications

Linking via XPort

- Get XBee on the Internet
- TCP/IP -> serial -> 802.15.4/ZigBee
- Also link Bluetooth, or RS-232, cell phone GPS...anything that supports TTL serial interface

XBee to XPort



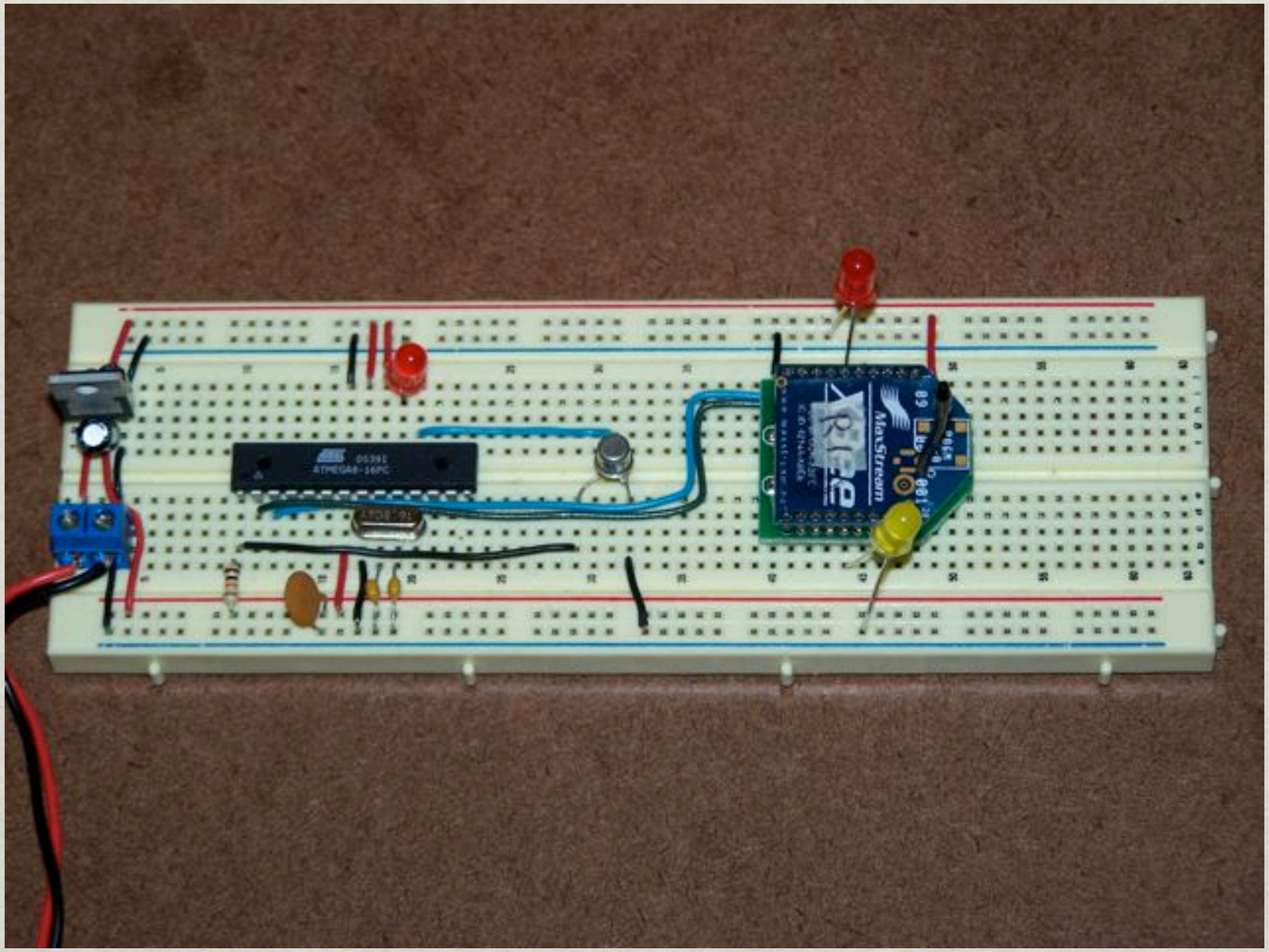
Broadcast Information

- Time
- Publishing status
- Methods & Variables for Objects
- Solving the Toaster Problem

Programming Arduino by Radio

- Put an Arduino on the ceiling, underwater or stuffed in a turkey
- Still be able to change and improve its behaviors
- Simple circuit: http://itp.faludi.com/meshnetworking/XBee/XBee_program_Arduino_wireless.html

Programming Arduino Wirelessly



TNO

(THE END)

Data Mode vs. Command Mode

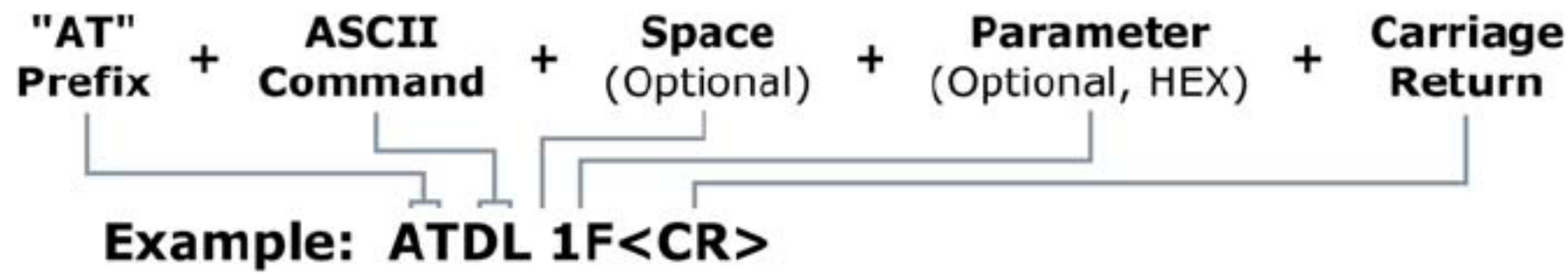
- Idle Mode, transmit and receive data
- Command Mode, talk to the XBee itself
  - +++ "*Yo, XBee*"
  - AT "*Attention!*" (Hayes command set)

Some AT Commands

- AT -> OK
- ATMY -> my address
- ATDH, ATDL -> destination address hi/lo
- ATID -> personal area network ID
- ATCN -> end command mode

AT Command Format

Figure 2-08. Syntax for sending AT Commands



Method 1 (One line per command)

Send AT Command	System Response
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D <Enter>	OK <CR> (Modify Destination Address Low)
ATWR <Enter>	OK <CR> (Write to non-volatile memory)
ATCN <Enter>	OK <CR> (Exit Command Mode)

Method 2 (Multiple commands on one line)

Send AT Command	System Response
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D,WR,CN <Enter>	OK, OK, OK <CR> (Command execution is triggered upon each instance of the comma)

Hexadecimals

- Just like decimals, but count from 0 to 15 in each position
- Since there's no existing single numeral representing 10 - 15, use A - F instead
- A = 10, B=11, C=12 ... F=15
- A1 = 161, common notation: 0xA1
- What does 3B equal?

API Mode

- Powerful, steeper learning curve
- Data wrapped together with commands, addressing and status information

Addressing In-Depth

- SL, SH: fixed serial number address
- MY: configured local 16 bit address
- DH, DL: destination address low and high
- Broadcast FF
- Broadcast PAN FF

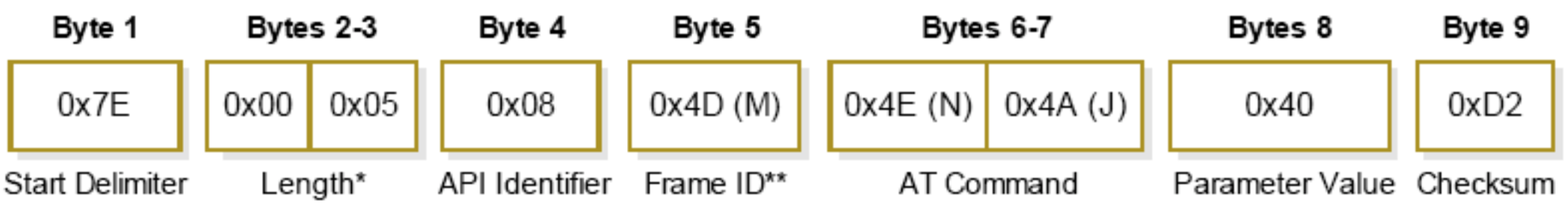
API Mode Format

Figure 4-01. UART Data Frame Structure:

MSB = Most Significant Byte, LSB = Least Significant Byte

Any data received prior to the start delimiter is silently discarded. If the frame is not received correctly or if the checksum fails, the module will reply with a module status frame indicating the nature of the failure.

Figure 4-07. Example: API frames when modifying the NJ parameter value of the module.



** Length [Bytes] = API Identifier + Frame ID + AT Command + Parameter Value*

*** "M" value was arbitrarily selected.*

*ATNJ = node join

API Mode TX/RX Frames

Figure 4-09. TX Packet Frames

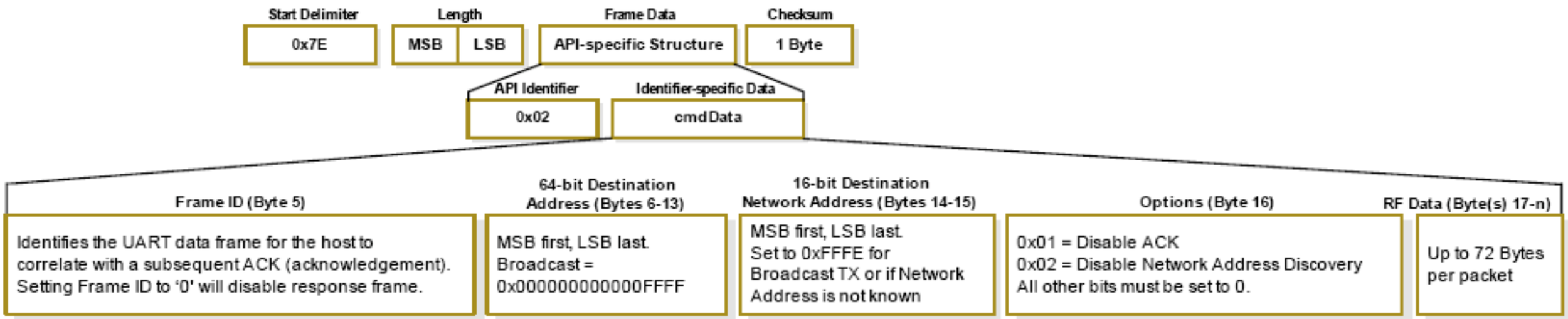
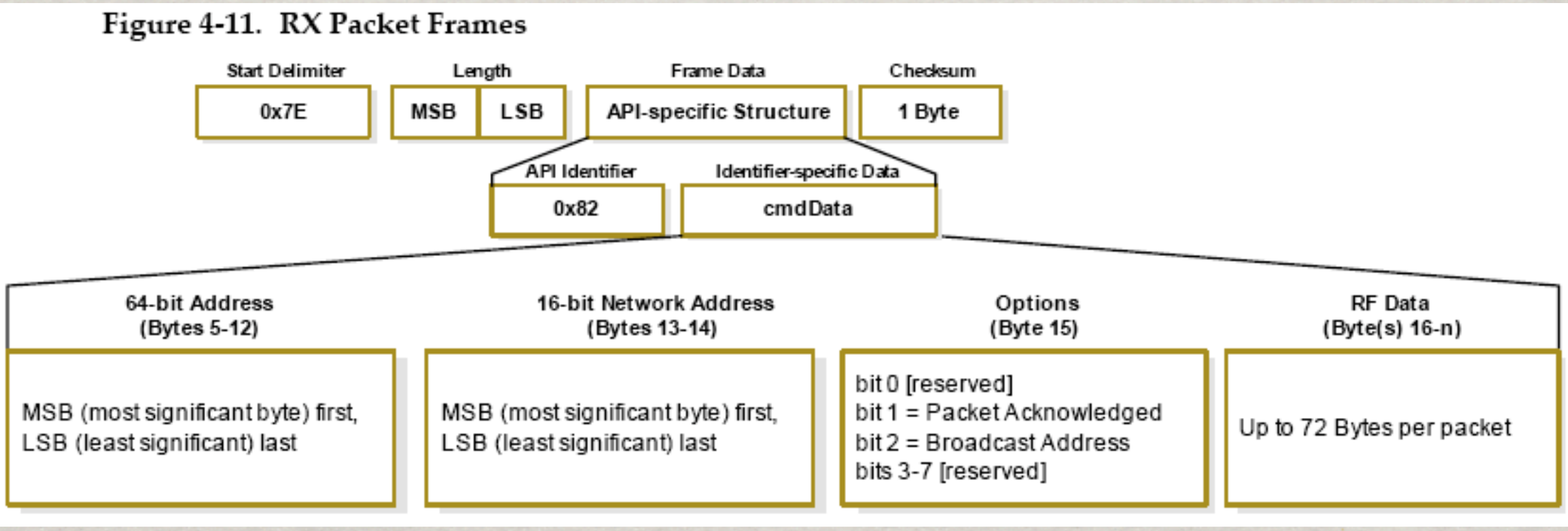


Figure 4-11. RX Packet Frames



Firmware Upload

- X-CTU Program
- Special circuit or development board
- Firmware, command interface, test area, terminal all Windows-only

Firmware Upload Board