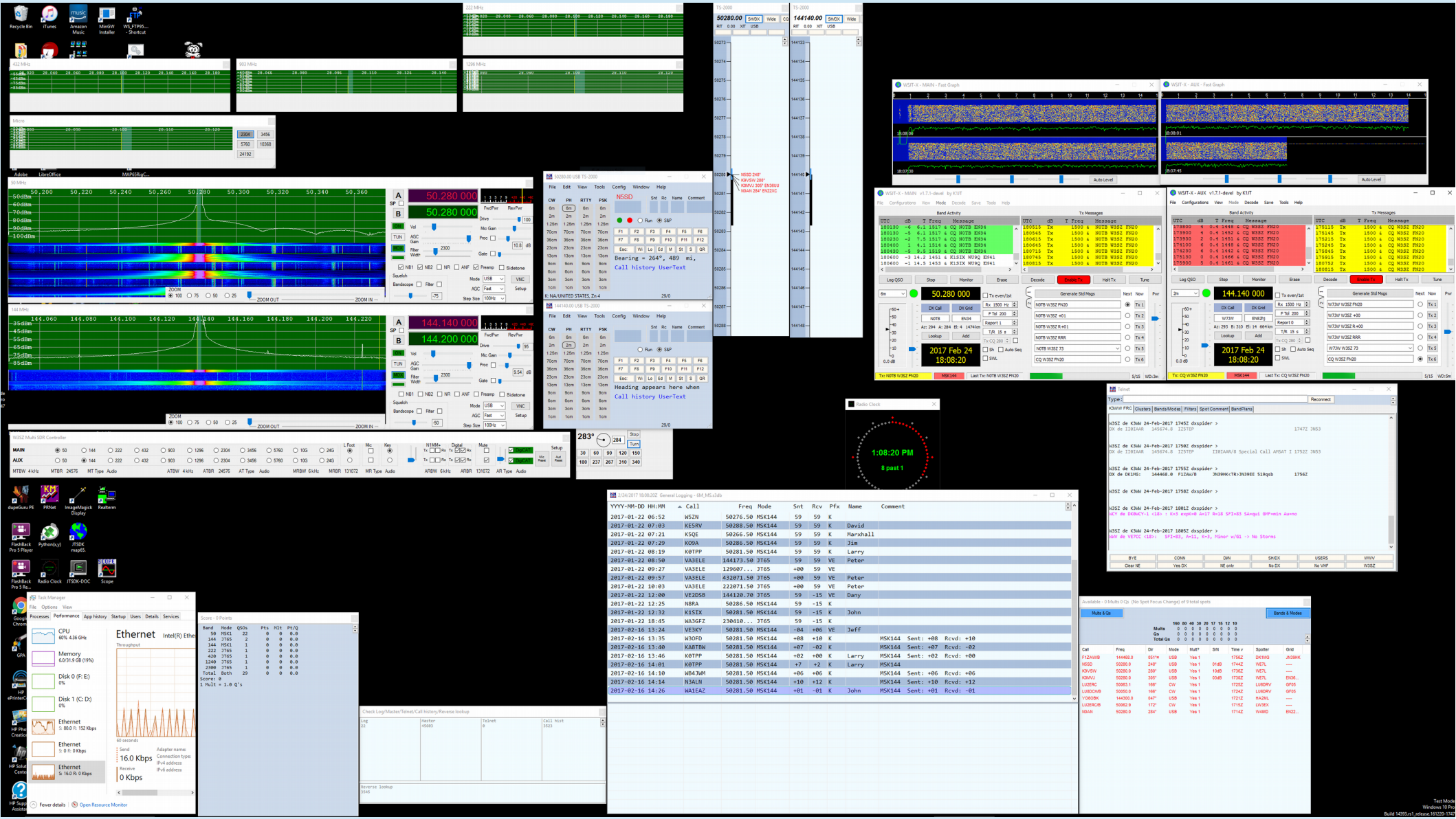
# Station Automation
# --W3SZ



# Device Bandswitching

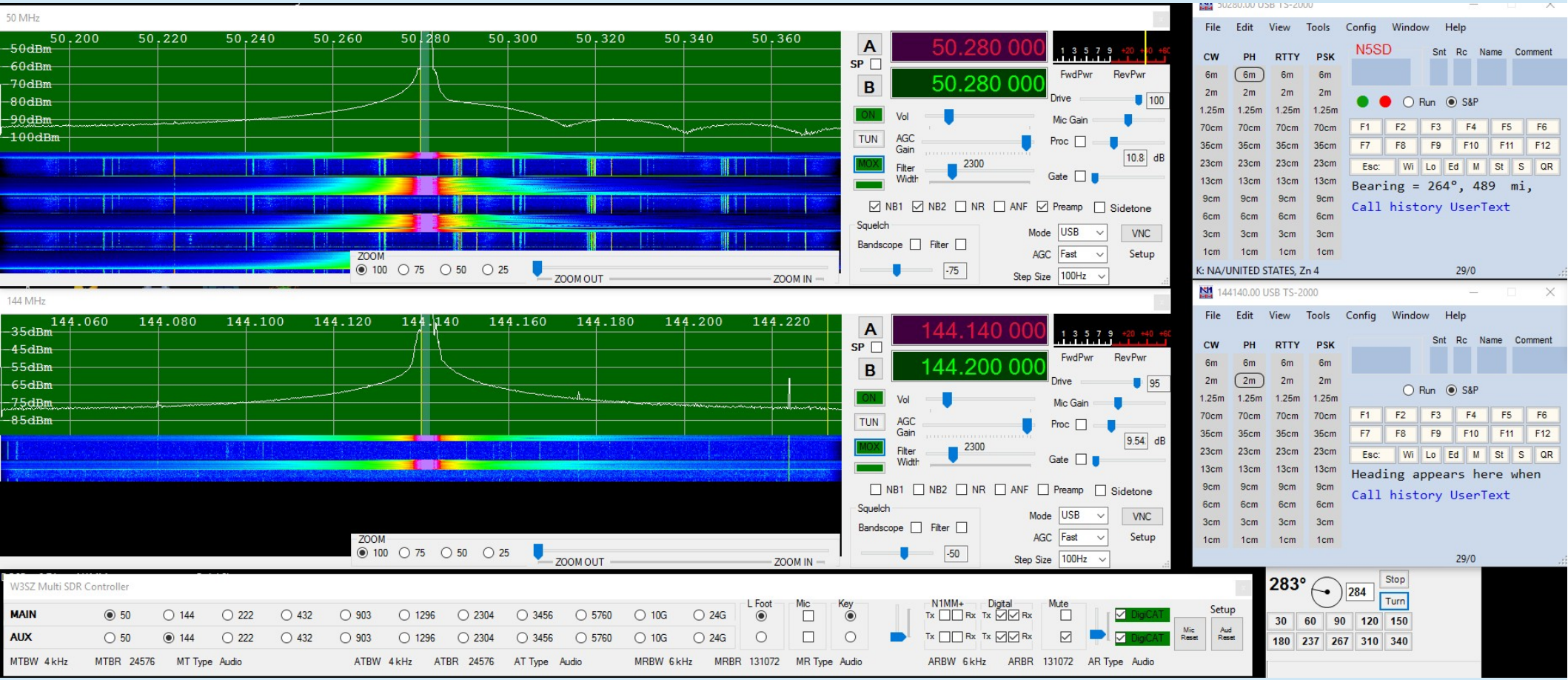
- Band switching IF radio, transverters is only part of the story
- If have more than 1 IF radio, then need to switch Mic (or digital transmit audio), receive audio, Footswitch/PTT, CW Key between IF radios
- Device bandswitching is potentially fundamentally different if you are using SDR vs conventional radio
  - For SDR can potentially use virtual connections, switch in software without need for electromechanical switching
  - For conventional radio, need to use electromechanical switching (relays or solid state switches)

# Device Bandswitching

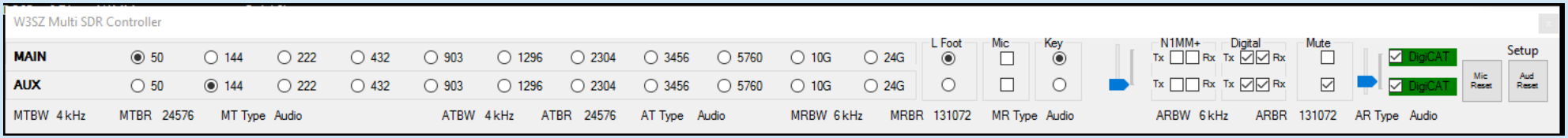
- If you are running Vista or newer OS:
  - N1MM will provide receive audio switching between Radios 1 and 2
  - N1MM will handle switching pre-recorded transmit audio messages between Radios 1 and 2
- Still need to deal with switching CW key, “live” microphone transmit audio between IF Radios
- N1MM options for receive audio switching may not be sufficient for all operators
- So something beyond plain vanilla N1MM is required for adequate device bandswitching

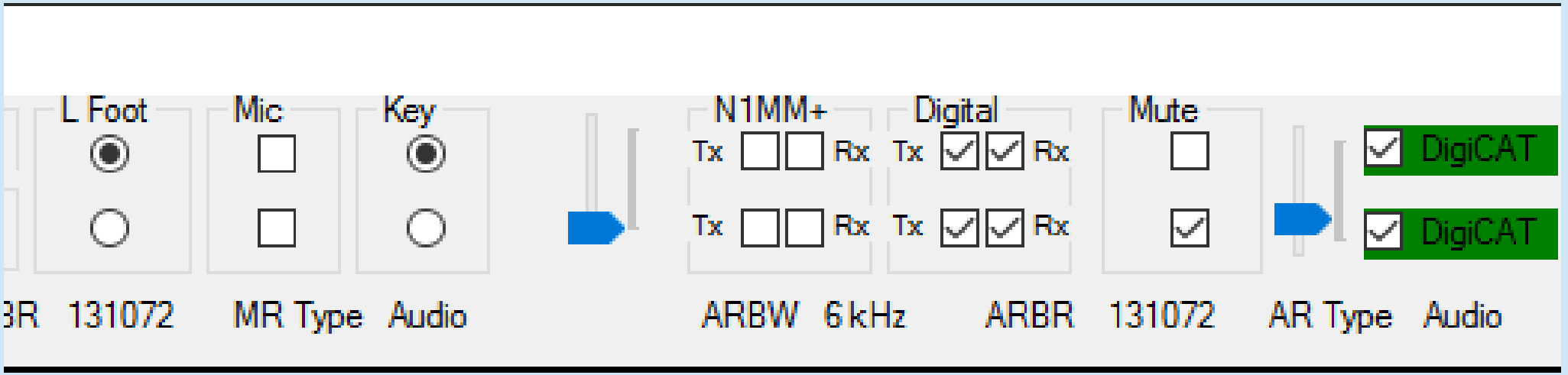
# Device Bandswitching

- At W3SZ: All done in software:



# Device Bandswitching




**Any Band / IF Radio can be assigned to either Main or Aux “radio” by the click of a button at any time**

**Footswitch, Mic, CW Key, Digital Audio can be selected for either Main or Aux “radio” at any time by the click of a button or checkbox**

**Rx Audio is always present for radios in both Main and Aux positions**

# Device Bandswitching

- Unfortunately, many stations don’t have the capability of controlling everything in software
- Stations exist at many different levels of technology / computerization
- What you need to do / what you can do to automate device bandswitching depends on many station-specific factors

# Device Bandswitching Maybe Not That Simple?

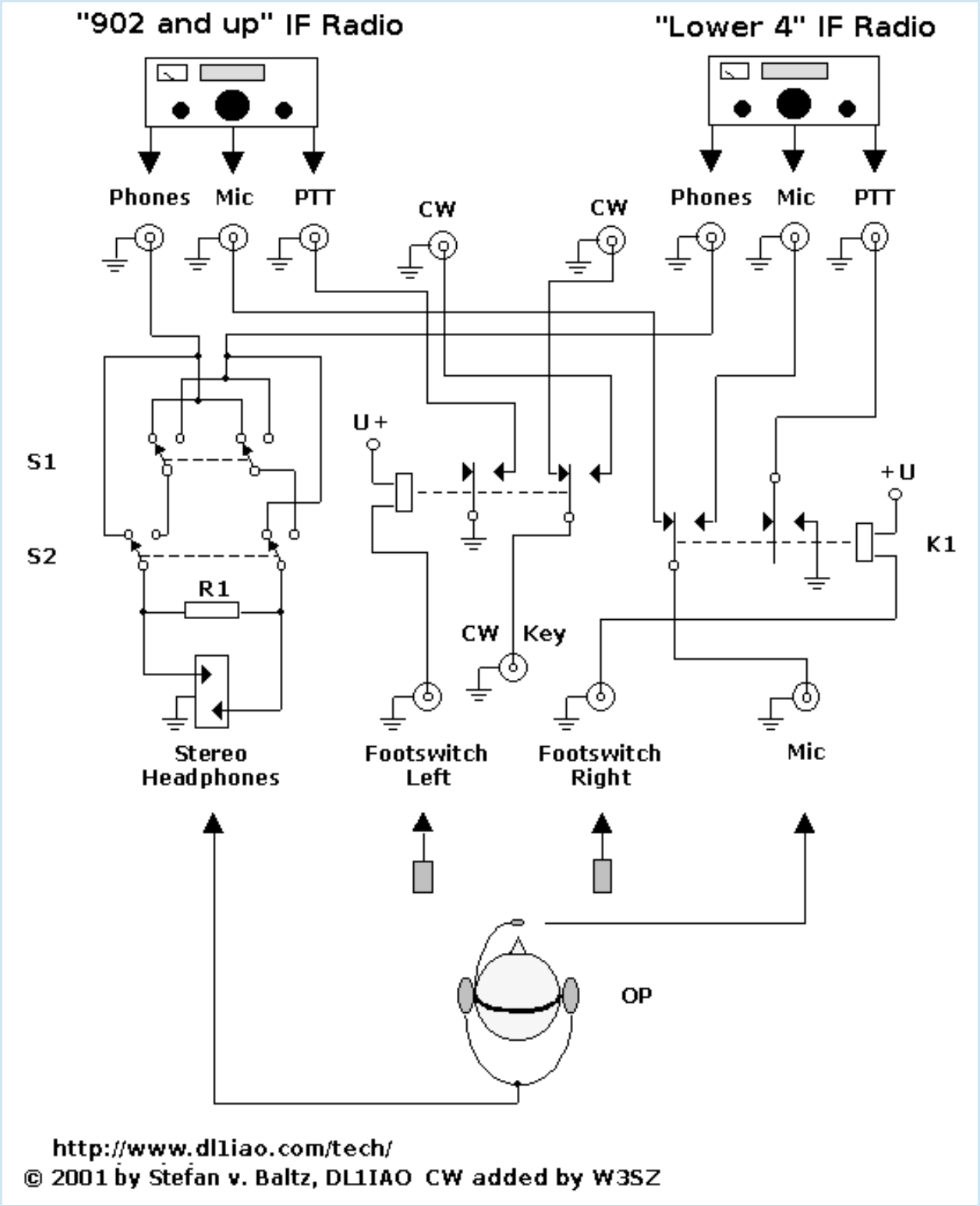
- CAT-Controlled (some or all of the IF radios)?
- Logging program: running SO2R or not?
- Logging program: MK2R+ / OTRSP compatible?
- Logging program: will it handle audio switching?
- Number of IF radios?
- Do radios have built-in codecs?
- SDR or conventional radio?
- Hardware or virtual serial ports?
- If SDRs, can everything be done in software?

# Device Bandswitching

- Electromechanical Device Switching using switches and/or relays rather than virtual audio and virtual ports and doing the switching in software, is the first (and sometimes final) step for many
- The mechanics of Electromechanical Device Bandswitching are fundamentally no different than bandswitching the transverters
  - You want mic or digital transmit audio, receive audio, footswitch/PTT, CW Key to follow your attention to the IF radio of your choice

# Electromechanical Device Bandswitching

- Can be done with relays and switches alone
  - No computer or software needed
- Can be done using Arduino or other MCU or SBC
  - Generally done in conjunction with logging software
- In either case, you want mic or digital transmit audio, receive audio, footswitch/PTT, CW Key to follow your attention to the IF radio of your choice as you change bands




# Electromechanical Device Bandswitching With Two IF Radios, No Computer

Disadvantages:

No IF Radio / Transverter Bandswitching

No CAT Control, thus
No Radio-Logger Synchronization

# Electromechanical Device Bandswitching

- Arduino (or other MCU) can be used with code like that shown for IF Radio / transverter bandswitching
  - If you are running SO1V with N1MM this is easy
    - Just tell the Arduino which IF radio gets the mic / CW key / PTT / receive audio connection for each band
    - Get band information from N1MM UDP output

# Arduino N1MM Ethernet Device Bandswitch SO1V

- Runs on an Uno
- Reads UDP “Radio” packets from N1MM
- N1MM Setup as per BBB Ethernet IF/Transverter device
- .ino file is here:
  - http://w3sz.com/ArduinoDeviceBandSwitcherEthernet.ino
- .pdf file is here:
  - http://w3sz.com/ArduinoDeviceBandSwitcherEthernet.pdf
- Code Handout pages 61-64

# Device Bandswitching SO1V mode N1MM

```
void SetBand(String commandOut)
{
  if(commandOut == "50" || commandOut == "14" || commandOut == "22" || commandOut == "43")
  {
    //This is low band radio, set relays off to connect to this radio

  digitalWrite(MicPin, LOW);
  digitalWrite(TxDigitalAudioPin, LOW);
  digitalWrite(CW_KeyPin, LOW);
  digitalWrite(LeftFootswitchPin, LOW);
  digitalWrite(RightFootswitchPin, LOW);
  digitalWrite(ReceiveAudioOnePin, LOW);
  digitalWrite(ReceiveAudioTwoPin, LOW);
  }
  else if (commandOut == "90" || commandOut == "12" || commandOut == "23" ||
  commandOut == "34" || commandOut == "57" || commandOut == "10" || commandOut == "24")
  {
    //This is microwave radio, set telays to ON to connect to this radio
  digitalWrite(MicPin, HIGH);
  digitalWrite(TxDigitalAudioPin, HIGH);
  digitalWrite(CW_KeyPin, HIGH);
  digitalWrite(LeftFootswitchPin, HIGH);
  digitalWrite(RightFootswitchPin, HIGH);
  digitalWrite(ReceiveAudioOnePin, HIGH);
  digitalWrite(ReceiveAudioTwoPin, HIGH);
  }
```

**Code Handout pp63-64**

# Device Bandswitching Complications

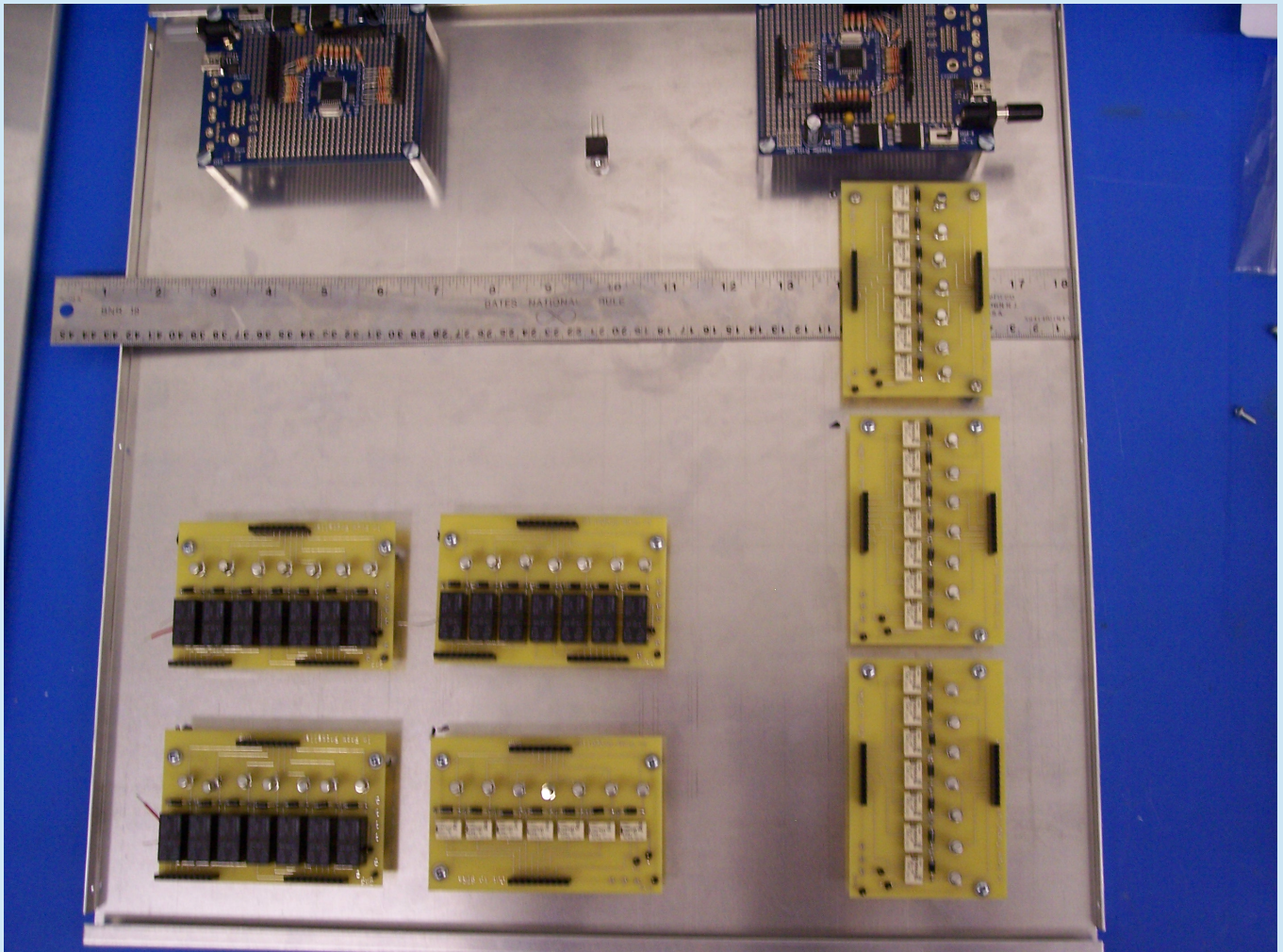
- Who runs SO1V these days??
- Don’t always WANT mic / CW key / PTT / receive audio to “follow the radio”
  - e.g when on 432 for liaison for 10 GHz contact
    - Want mic on 432, CW key on 10 GHz
    - Want receive audio on BOTH radios
    - Want separate PTT / footswitch for each radio
  - But other times, e.g when running the bands without need for liaison, DO want the mic / CW key / PTT / receive audio to “follow the radio”
- So need more system complexity in order to accommodate both situations

# Device Bandswitching

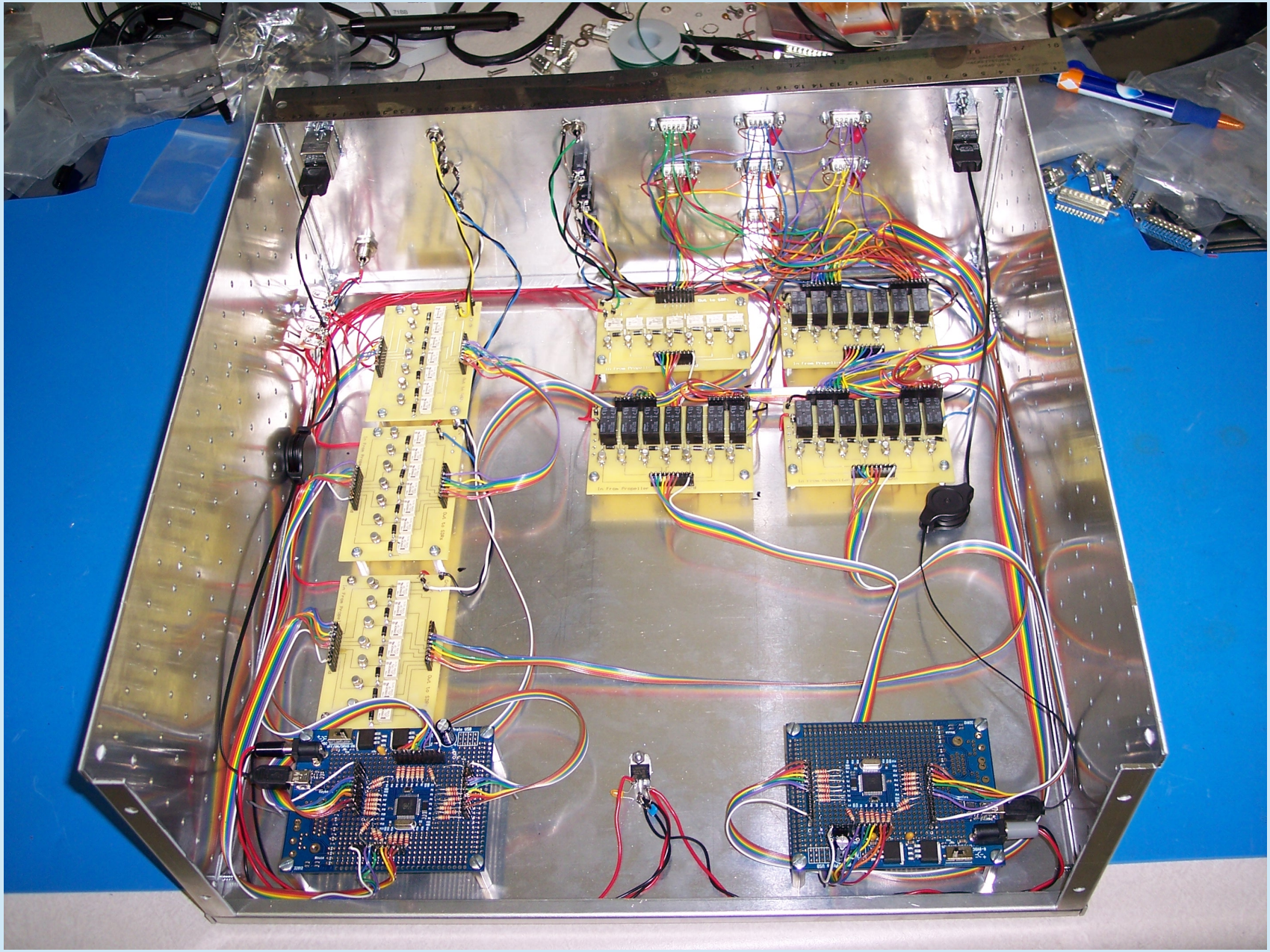
- Before I developed my “everything in software” approach, I used electromechanical device switching of mic, 3 receive audio channels, 2 foot switches, CW key between 7 IF radios (one radio each for 50, 144, 222, 432, 903, and 1296 MHz and one shared by 2-24 GHz)
- I used two Parallax Propeller MCUs controlled by 2 USB ports

# Device Bandswitching

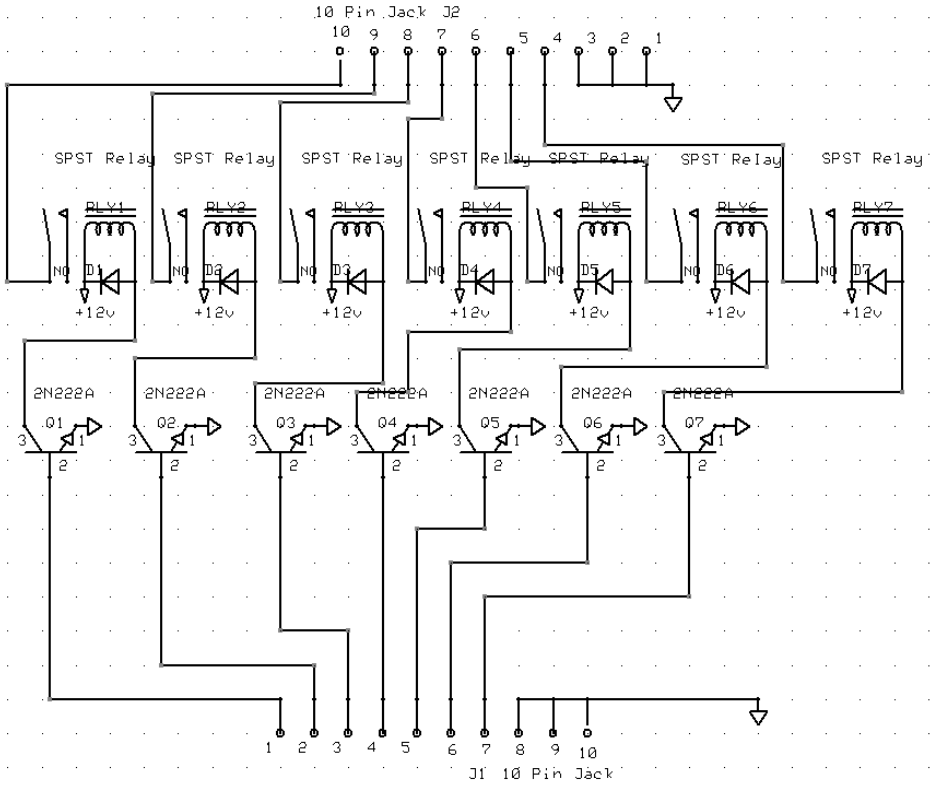
- 3 boards for receive audio switching, 4 boards for mic, 2 footswitches, CW key switching



# Device Bandswitching

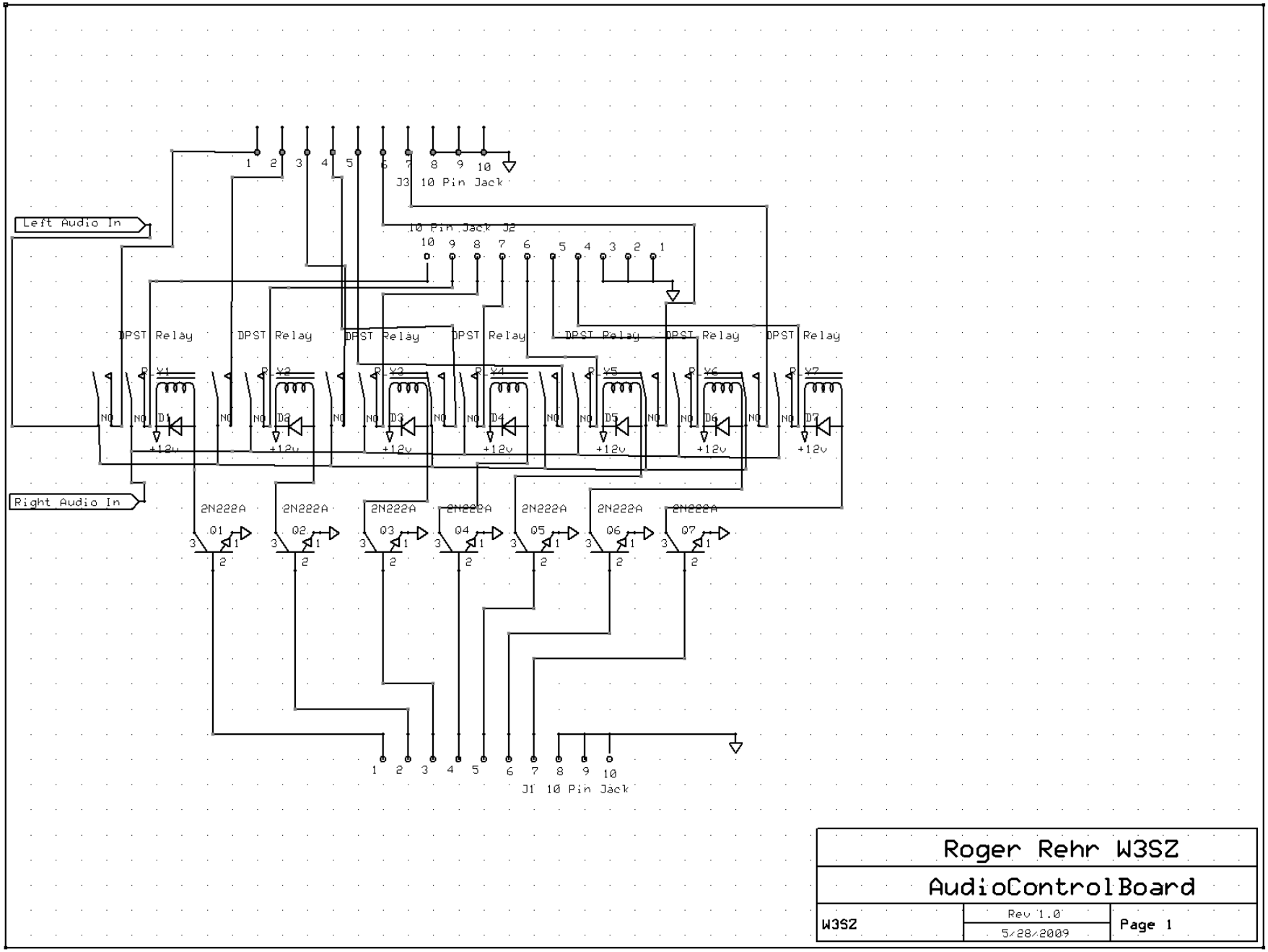


# Device Bandswitching

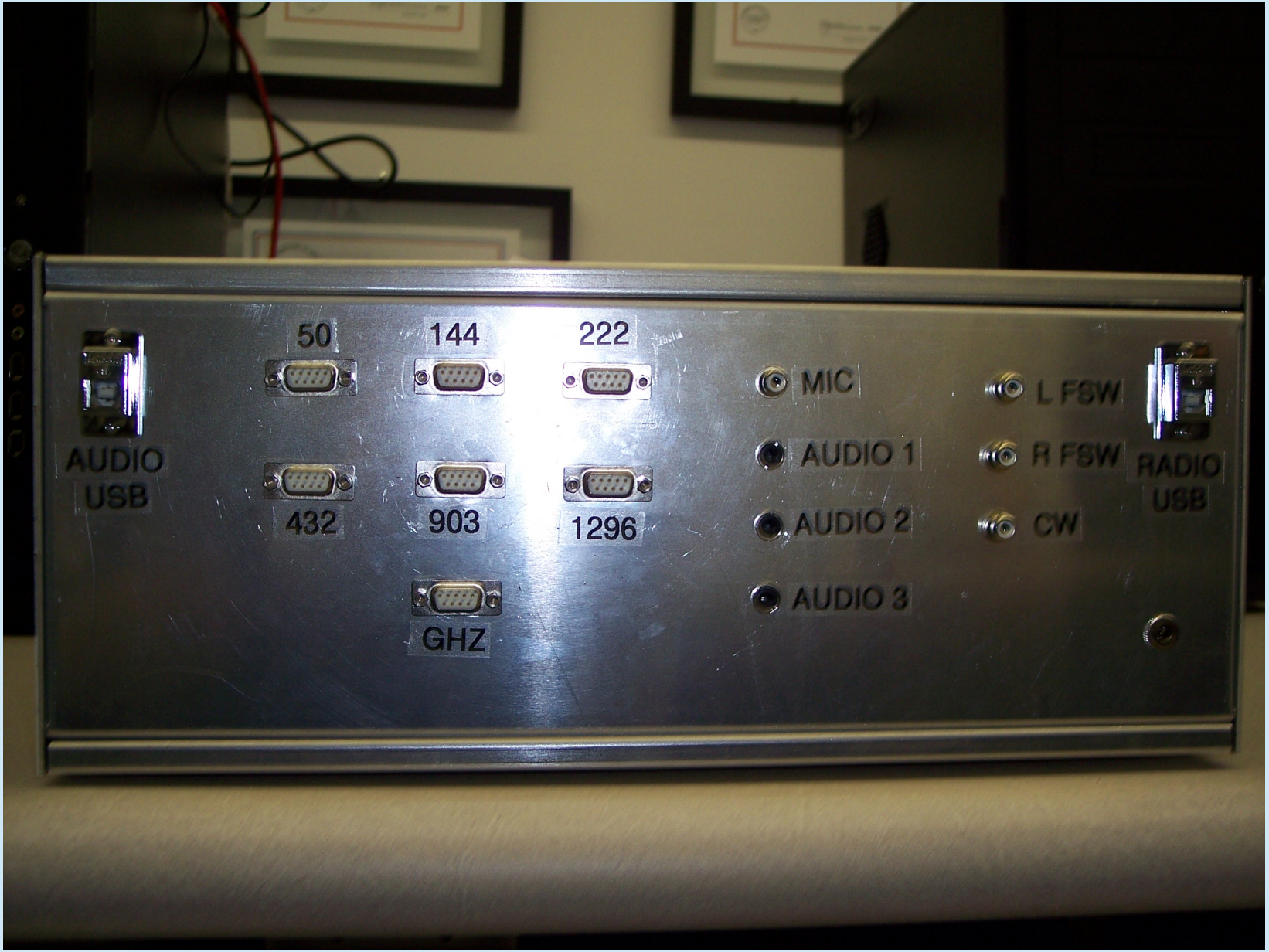



| | | |
|---|---|---|
| Roger Rehr W3SZ | | |
| RadioControlBoard | | |
| W3SZ | Rev 1.0<br>5/28/2009 | Page 1 |

# Device Bandswitching

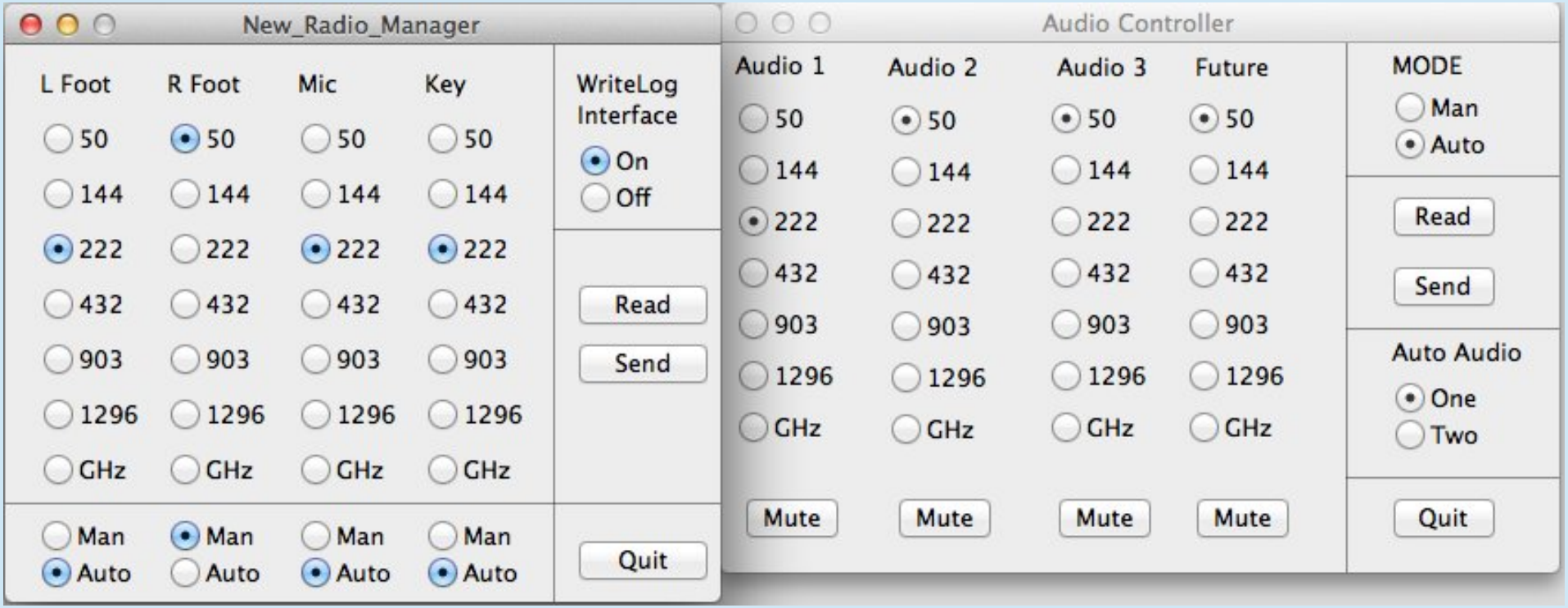


# Device Bandswitching



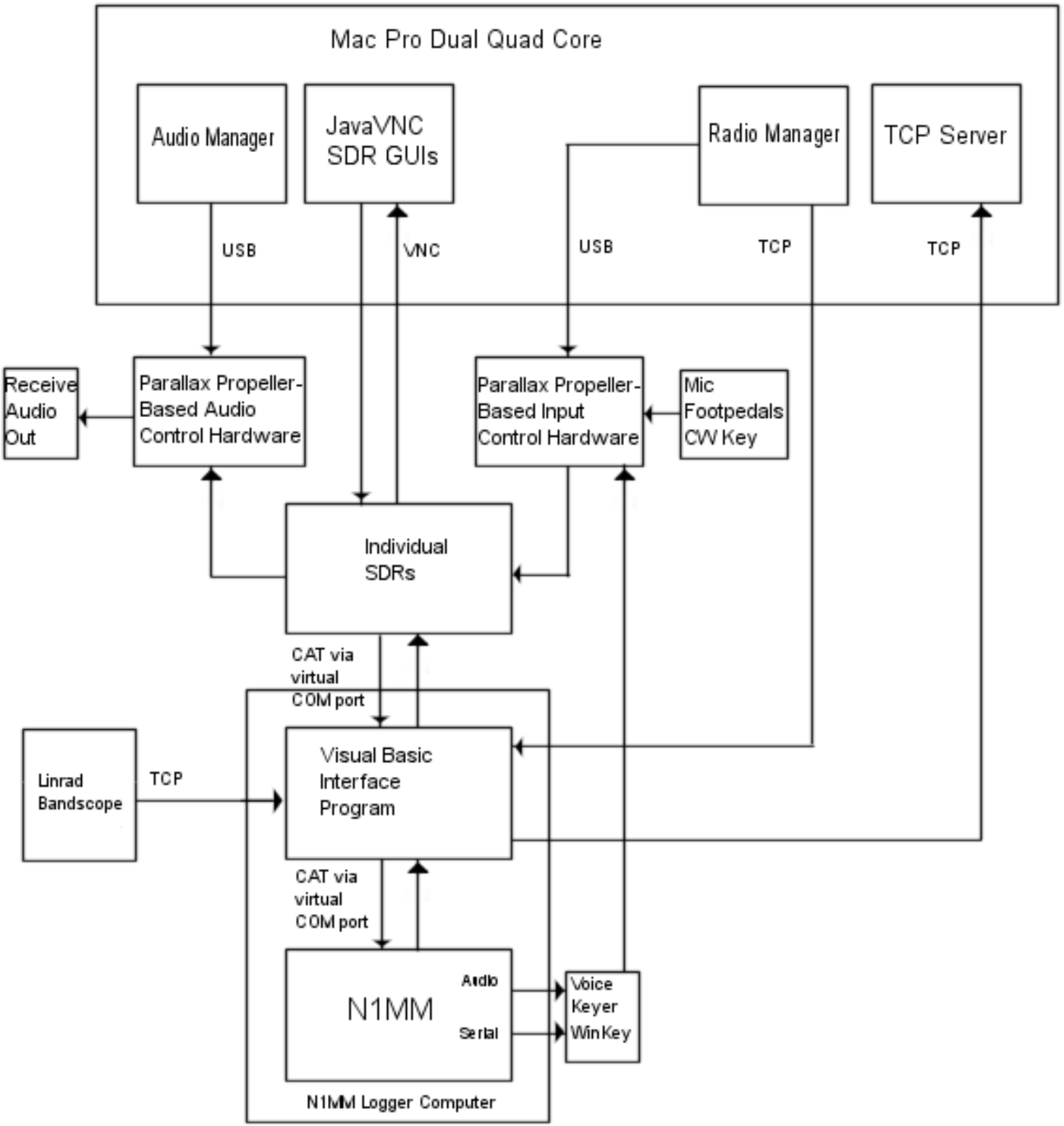
# Device Bandswitching

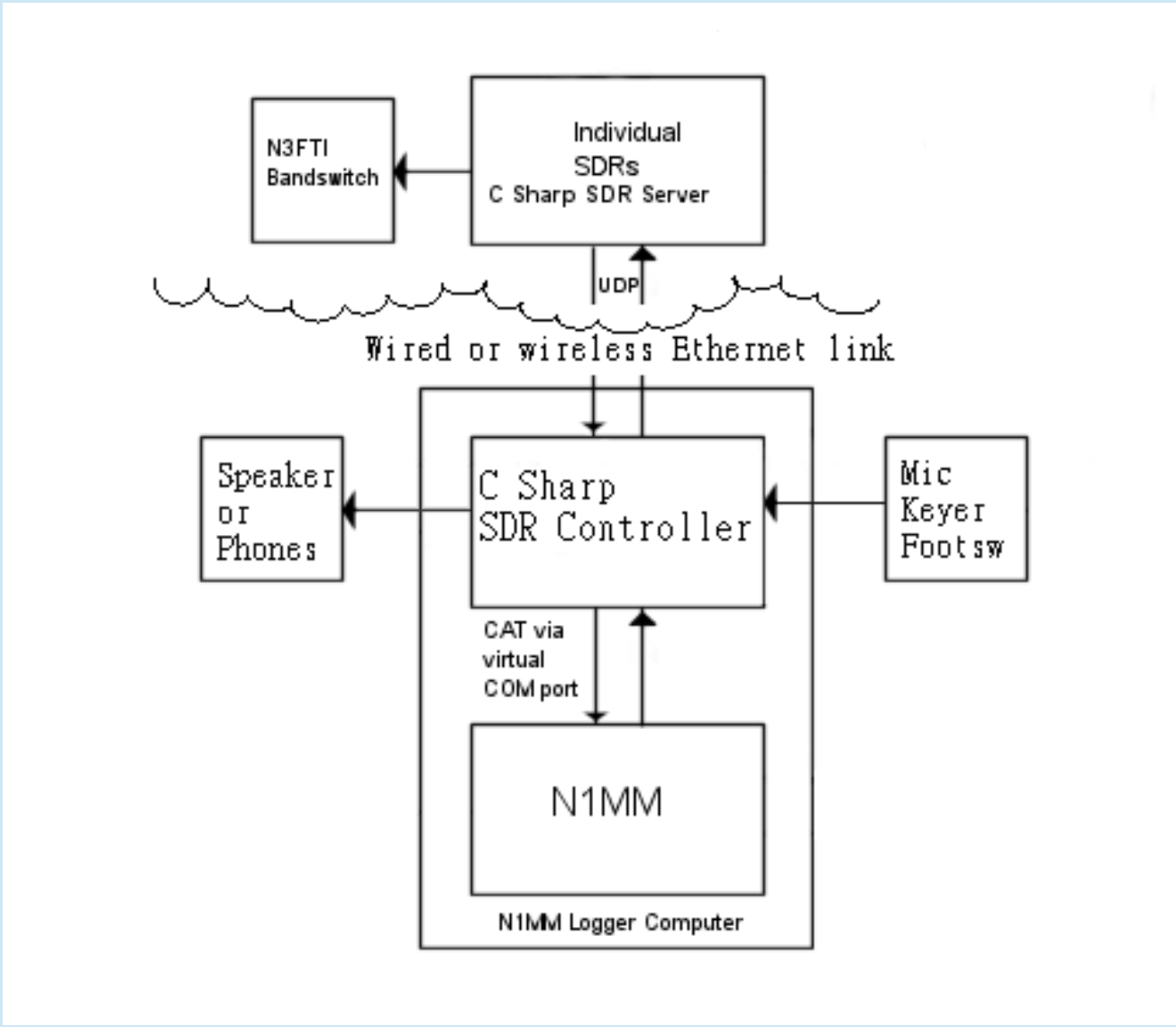
# Device Bandswitching



# Device Manager

- Details of OS X code are here:
  - http://www.nitehawk.com/w3sz/osxhpsdrserver.htm
- Propeller Spin code is here:
  - http://w3sz.com/AudioController.spin
  - http://w3sz.com/RadioManager.spin
- PCB and SCH files are here:
  - http://w3sz.com/RadioControlBoardNew.pcb
  - http://w3sz.com/RadioAudioControlBoard.pcb
  - http://w3sz.com/RadioControlBoardNew.sch
  - http://w3sz.com/RadioAudioControlBoard.sch


Mac Pro Dual Quad Core
Audio Manager
JavaVNC SDR GUIs
Radio Manager
TCP Server
USB
VNC
USB
TCP
TCP
Receive Audio Out
Parallax Propeller-Based Audio Control Hardware
Parallax Propeller-Based Input Control Hardware
Mic Footpedals CW Key
Individual SDRs
CAT via virtual COM port
Linrad Bandscope
TCP
Visual Basic Interface Program
CAT via virtual COM port
N1MM
Audio
Serial
Voice Keyer WinKey
N1MM Logger Computer


N3FTI
Bandswitch
Individual
SDRs
C Sharp SDR Server
UDP
Wired or wireless Ethernet link
Speaker
or
Phones
C Sharp
SDR Controller
Mic
Keyer
Footsw
CAT via
virtual
COM port
N1MM
N1MM Logger Computer

# Device Bandswitching
# “Typical” Station’s Capability:

- CAT Control
- SO2R
- N1MM or WriteLog or DXLabs with Win 7 or newer
- 2 IF Radios
  - 50, 144, 222, 432 MHz
  - 902 MHz and up

# Device Bandswitching
# “Typical” Capability → SO2R Box

- Legacy LPT Devices
  - Top Ten Devices “DX Doubler”
  - Array Solutions “SO2R Master”
- USB-Serial Port Devices
  - MicroHAM MK2R+  $939
  - OTRSP compatible devices like YCCC SO2RBox
  - SO2RDuino by Paul Young, K1XM
    - http://ncjweb.com/features/julaug10feat.pdf
    - http://ncjweb.com/bonus-content/NCJJulAugBonus.ZIP
    - Depending on your junkbox, cost could be $5

# USB-Serial SO2R Boxes:

- Provide Mic, CW Key, PTT, Receive Audio switching between Radio 1 and Radio 2
- Provide Band data for IF Radio / Transverter bandswitching
- Simultaneous transmit on 2 radios is not always possible
  - e.g. MK2R+ allows dual transmission if NOT in SO2R mode, but NOT if in SO2R mode
- Simple SO2RDuino provides for only 1 PTT signal
  - also does NOT separate out control of Mic, CW Key, PTT

# Super MEGA SO2RDuino

- Extension of SO2RDuino using Mega 2560 R3 to give:
  - Two PTT inputs
  - Allows both Radio 1 and Radio2 to transmit at same time using PTT A for one and PTT B for the other
  - Ability to indepently set status to either “Follow Focus Radio” or “Manual Assignment to Radio 1 or Radio 2” for each of:
    - Receive Audio (Follows Rx Focus radio)
    - PTT A (Follows Tx Focus radio)
    - PTT B (Follows Tx Focus radio)
    - Mic (Follows Tx Focus radio)
    - CW Key (Follows Tx Focus radio)
  - IF Radio / Transverter bandswitching for up to 16 bands. Each Radio has 16 band outputs, which are set up using “Antennas” Configuration page in N1MM as previously described.

# Super MEGA SO2RDuino

- If you are confused about “Entry Focus”, “Tx Focus”, “Rx Focus” go here:
  - http://n1mm.hamdocs.com/tiki-index.php?page=SO2R+-+Single+Operator+Two+Radio+Operation&highlight=SO2R#The_SO2R_Dots_LEDs_
- Basically:
  - GREEN DOT = Rx Focus = Entry Focus
  - RED DOT = Tx Focus
    - DOES **NOT** ALWAYS = Rx Focus or Entry Focus

# The Original SO2RDuino by Paul Young K1XM

# Concept Illustration of Super MEGA SO2RDuino

# Concept Illustration of Super MEGA SO2RDuino

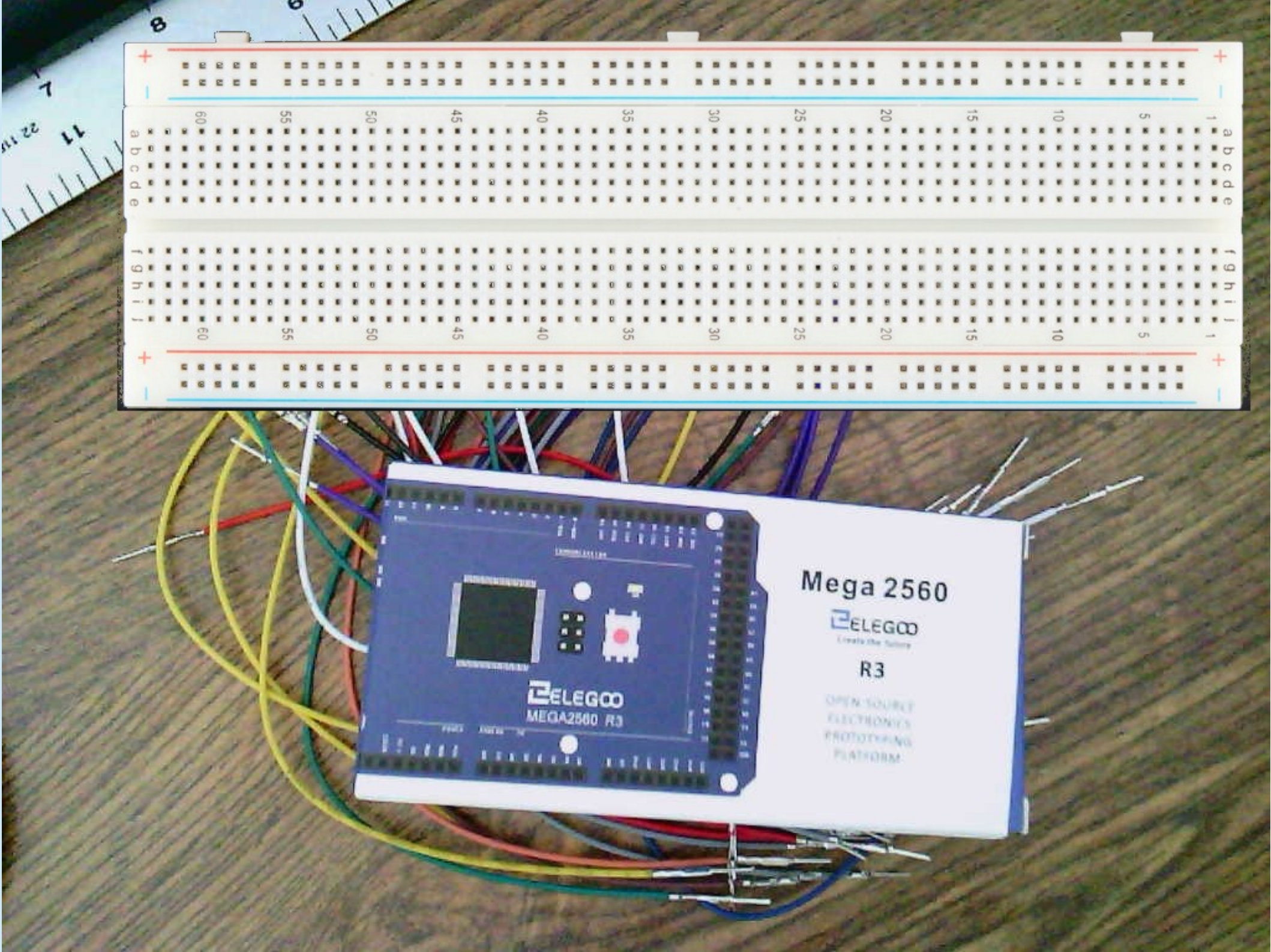
# Super MEGA SO2RDuino

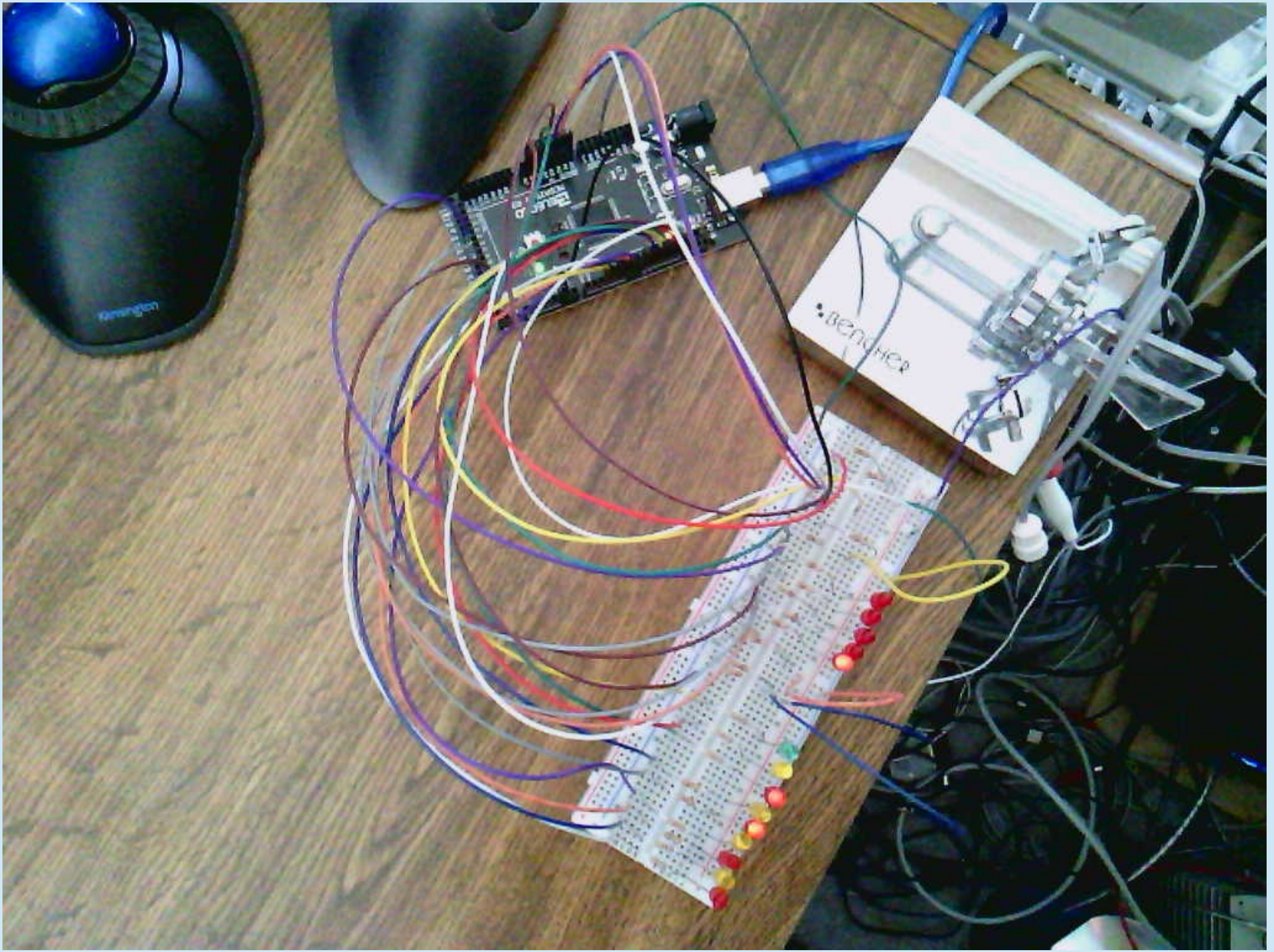
- Code is based on Paul Young's SO2RDuino code at:

  http://ncjweb.com/bonus-content/NCJJulAugBonus.ZIP
- Super MEGA SO2RDuino code is at:

  http://w3sz.x10.mx/SO2RDuino_ToINO_New_Mega.ino

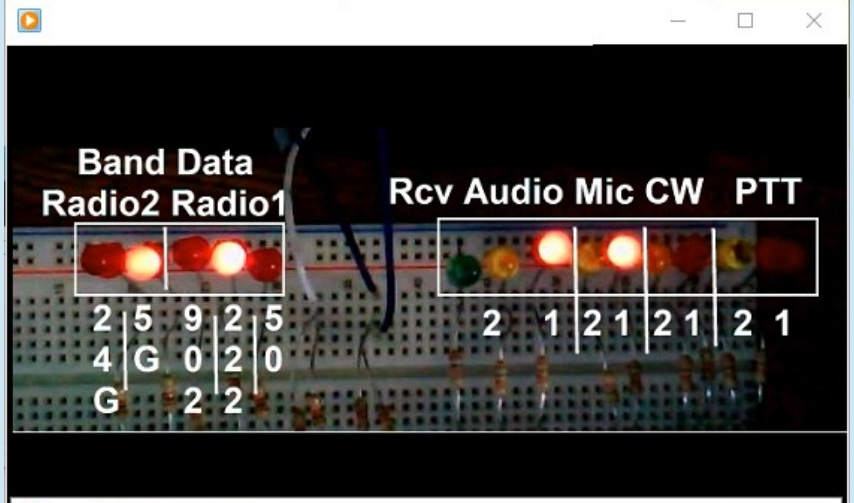
  http://w3sz.x10.mx/SuperMEGASO2RDuino.pdf
- Code is 30 pages long so we won't review it here
- It uses the concepts and code presented in the other projects that ARE discussed in detail here today
  - The code is well documented


Mega 2560
ELEGOO
R3
ELEGOO
MEGA2560 R3



# Super MEGA SO2RDuino

- Two videos follow, which also help cement the concepts of what we are doing with N1MM/SO2RBox combination
- First video discusses and demonstrates function of box
- Second video demonstrates manual setting of radio-CW key association




222125.06 CW TS-2000 Radio 1

File Edit View Tools Config Window Help

| CW | PH | RTTY | PSK |
|---|---|---|---|
| 6m | 6m | 6m | 6m |
| 4m | 4m | 4m | 4m |
| 2m | 2m | 2m | 2m |
| 1.25m | 1.25m | 1.25m | 1.25m |
| 70cm | 70cm | 70cm | 70cm |
| 33cm | 33cm | 33cm | 33cm |
| 23cm | 23cm | 23cm | 23cm |
| 13cm | 13cm | 13cm | 13cm |
| 9cm | 9cm | 9cm | 9cm |
| 6cm | 6cm | 6cm | 6cm |
| 3cm | 3cm | 3cm | 3cm |
| 1cm | 1cm | 1cm | 1cm |

Left Click, QSY to 902100

Snt Rcv Grid

Run S&P 11

F1 Qrl? F2 Exch F3 Tu F4 F5 His F6 Repeat
F7 Spare F8 Agn? F9 Nr? F10 Call? F11 F12 Wipe
Esc: Stop Wipe Log Edit Mark Stor Spot QRZ

Heading appears here when enabled.
Call history UserText appears here when

CW Speed = 11 286/66 52,206

6 cm USB TS-2000 Radio 2

File Edit View Tools Config Window Help

| CW | PH | RTTY | PSK |
|---|---|---|---|
| 6m | 6m | 6m | 6m |
| 4m | 4m | 4m | 4m |
| 2m | 2m | 2m | 2m |
| 1.25m | 1.25m | 1.25m | 1.25m |
| 70cm | 70cm | 70cm | 70cm |
| 33cm | 33cm | 33cm | 33cm |
| 23cm | 23cm | 23cm | 23cm |
| 13cm | 13cm | 13cm | 13cm |
| 9cm | 9cm | 9cm | 9cm |
| 6cm | 6cm | 6cm | 6cm |
| 3cm | 3cm | 3cm | 3cm |
| 1cm | 1cm | 1cm | 1cm |

Snt Rcv Grid

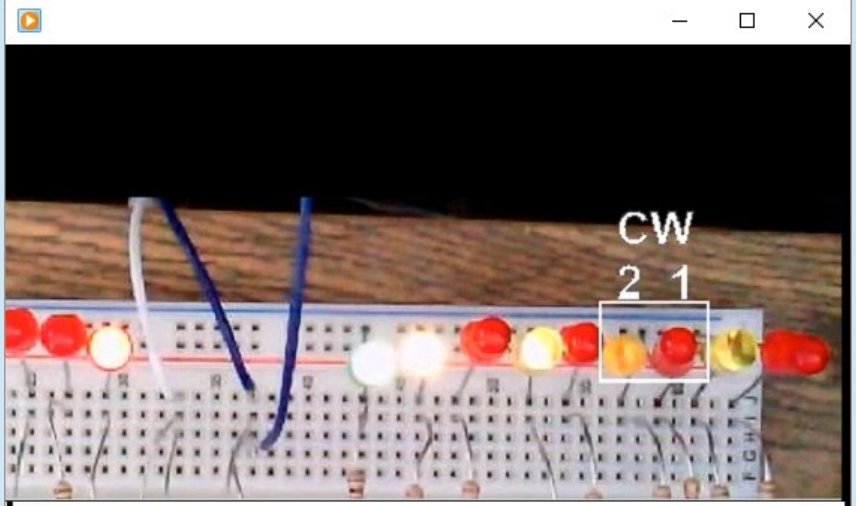
Run S&P

F1 S&P F2 Exch F3 Spare F4 F5 His F6 Spare
F7 Rpt F8 Agn? F9 Zone F10 F11 F12 Wipe
Esc: Stop Wipe Log Edit Mark Stor Spot QRZ

Heading appears here when enabled.
Call history UserText appears here when

CW Speed = 42 286/66 52,206

E:\StationAutomation\PackRatsMiniTalk\5_SHORT_Super_MEGA_SO2RDuinoWithN1MM.optimizedSHORTforCONFERENCE.wmv




50125.00 CW TS-2000 Radio 1

File Edit View Tools Config Window Help

| CW | PH | RTTY | PSK |
|---|---|---|---|
| 6m | 6m | 6m | 6m |
| 4m | 4m | 4m | 4m |
| 2m | 2m | 2m | 2m |
| 1.25m | 1.25m | 1.25m | 1.25m |
| 70cm | 70cm | 70cm | 70cm |
| 33cm | 33cm | 33cm | 33cm |
| 23cm | 23cm | 23cm | 23cm |
| 13cm | 13cm | 13cm | 13cm |
| 9cm | 9cm | 9cm | 9cm |
| 6cm | 6cm | 6cm | 6cm |
| 3cm | 3cm | 3cm | 3cm |
| 1cm | 1cm | 1cm | 1cm |

Snt Rcv Grid

Run S&P 10

F1 Qrl? F2 Exch F3 Tu F4 F5 His F6 Repeat
F7 Spare F8 Agn? F9 Nr? F10 Call? F11 F12 Wipe
Esc: Stop Wipe Log Edit Mark Stor Spot QRZ

Heading appears here when enabled.

Call history UserText appears here when

CW Speed = 10 286/66 52,206

6 cm CW TS-2000 Radio 2

File Edit View Tools Config Window Help

| CW | PH | RTTY | PSK |
|---|---|---|---|
| 6m | 6m | 6m | 6m |
| 4m | 4m | 4m | 4m |
| 2m | 2m | 2m | 2m |
| 1.25m | 1.25m | 1.25m | 1.25m |
| 70cm | 70cm | 70cm | 70cm |
| 33cm | 33cm | 33cm | 33cm |
| 23cm | 23cm | 23cm | 23cm |
| 13cm | 13cm | 13cm | 13cm |
| 9cm | 9cm | 9cm | 9cm |
| 6cm | 6cm | 6cm | 6cm |
| 3cm | 3cm | 3cm | 3cm |
| 1cm | 1cm | 1cm | 1cm |

Snt Rcv Grid

Run S&P 20

F1 Qrl? F2 Exch F3 Tu F4 F5 His F6 Repeat
F7 Spare F8 Agn? F9 Nr? F10 Call? F11 F12 Wipe
Esc: Stop Wipe Log Edit Mark Stor Spot QRZ

Heading appears here when enabled.

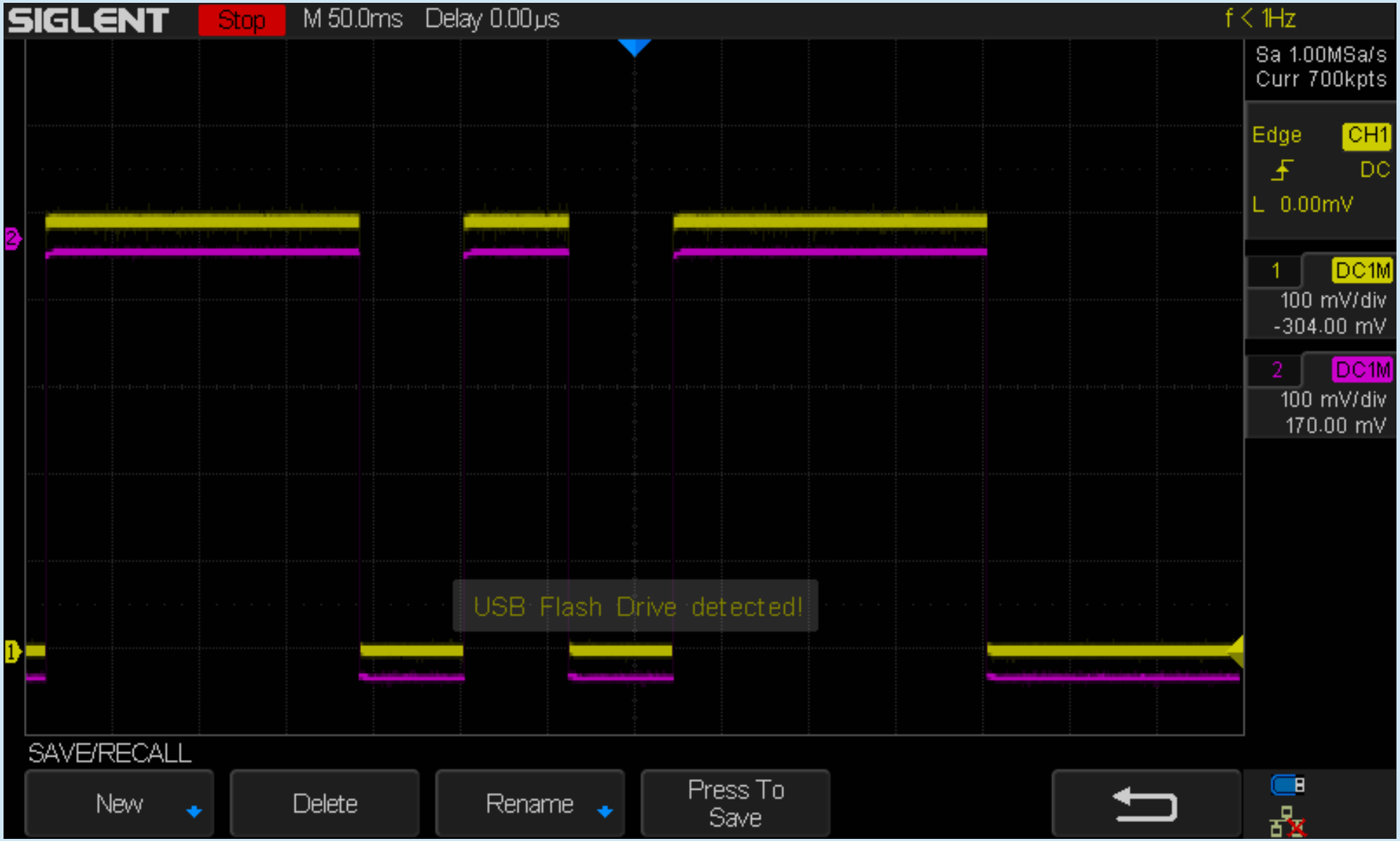
Call history UserText appears here when

286/66 52,206

E:\StationAutomation\PackRatsMiniTalk\6_SuperMegaWithN1MM_ManualCWSetRadio.wmv

# Is the MEGA Fast Enough?
# No Interrupts Used – 18 wpm

# Is the MEGA Fast Enough?
# No Interrupts Used – 18 wpm



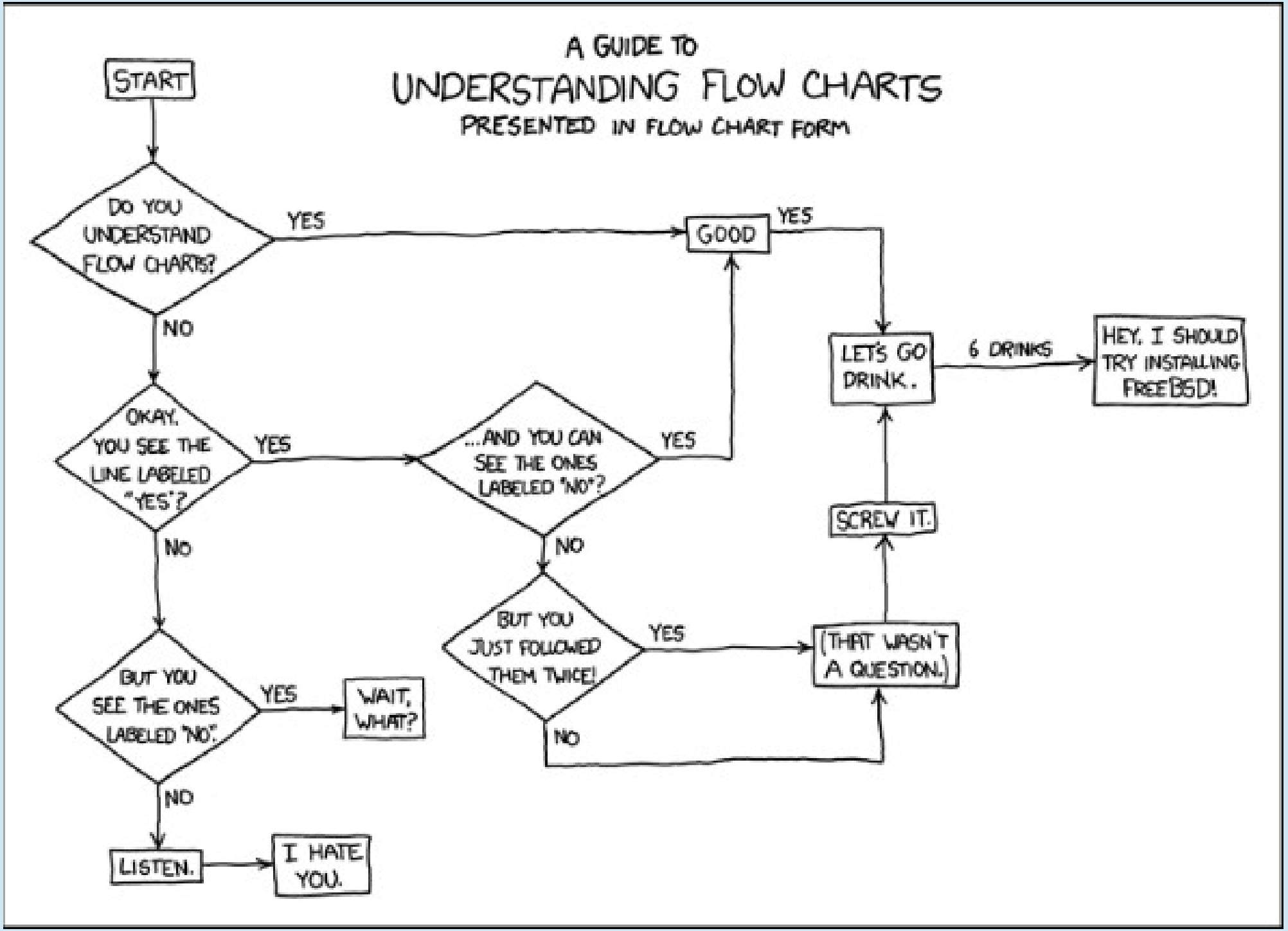
# Is the MEGA Fast Enough?
# No Interrupts Used – 42 wpm

# Station Automation Coding

- Very Simple:
  - Got Some Input
  - Did Something With It
  - Produced Some Output

# Station Automation Coding

1) Included libraries containing external functions
2) Defined variables and constants
3) Setup ()
    - Define and initialize GPIO pins
    - Defined, started, serial port, Ethernet port
4) Loop()
    - Received input from ports
    - Parsed / processed data to extract desired information
    - Used information derived from data to bandswitch using GPIO pins
5) From within Loop(), called other functions() as needed


A GUIDE TO
UNDERSTANDING FLOW CHARTS
PRESENTED IN FLOW CHART FORM
START
DO YOU UNDERSTAND FLOW CHARTS?
YES
GOOD
YES
NO
LET'S GO DRINK.
6 DRINKS
HEY, I SHOULD TRY INSTALLING FREEBSD!
OKAY, YOU SEE THE LINE LABELED "YES"?
YES
...AND YOU CAN SEE THE ONES LABELED "NO"?
YES
SCREW IT.
NO
NO
BUT YOU JUST FOLLOWED THEM TWICE!
YES
(THAT WASN'T A QUESTION.)
BUT YOU SEE THE ONES LABELED "NO".
YES
WAIT, WHAT?
NO
NO
LISTEN.
I HATE YOU.