NIXIE CLOCK *REDUX*

IN-12 VARIANT (ИН-12Б)

Drew Maatman

1. Warnings

2. Clock Overview

3. Clock Features

- The tubes can be configured to display:
    - Time (24 hour or AM/PM)
    - Date (MM/DD/YY)
    - Weekday (Numbered from 1 to 7)
    - Alarm Time (24 hour or AM/PM)
    - Alarm Arm (Armed or Disarmed)
    - 24 Hour or AM/PM Mode
    - Time/Date/Weekday/Alarm Setting Modes (set through pushbuttons)
    - Tube Brightness Setting (10% to 100%)
    - Tube RGB Backlight Color (build option)
- Additional Features:
    - USB Serial Port
    - Tube Elapsed Time Counter (build option)
        - Nixie tubes age, this can track how long voltage has been applied, as well as how many times voltage has been cycled
    - Clock Elapsed Time Counter (build option)
        - Tracks total time clock has input power applied, power cycles
    - Capacitive Touch Pushbuttons for Power On/Off and user interface
    - Battery Backed Real Time Clock (build option)
    - Internal Telemetry (build option)
    - Input Power Protection
        - Reverse Polarity Protection
        - Input Under/Overvoltage Protection
        - Input Current Limit
    - Tube RGB Backlight Brightness (adjustable over USB)

4. Pushbutton Interface

5. USB/Serial Interface

A. Establishing a Connection

The Nixie Clock can be communicated with through its USB port. It enumerates as an FTDI virtual COM port on Windows, Macintosh, and Linux machines. The following serial settings should be used:

- 115.2kBaud
- 8-bit Data Length
- No Parity
- 1 stop bit
- No flow control

A serial client such as PuTTY or TeraTerm can be used on Windows machines. Macintosh and Linux machines allow serial communication through the command line using the `screen` command. If using PuTTY, ensure that `Implicit CR in every LF` is enabled, `Local Echo` is set to `Force on`, and `Local Line Editing` is set to `Force on`.

B. Beginning Communication

On reset (power on or software/hardware reset), the Nixie Clock will print startup messages and debug information about the startup code as it configures itself. This will not appear if the serial terminal is opened after boot is complete. Another easy way to determine if a connection is established is to call the `*IDN?` command.

C. Serial Color Scheme

- Help messages and neutral responses appear in yellow
- System parameters and affirmative responses appear in green
- Measurement responses appear in cyan
- Urgent/interrupt messages appear in magenta
- Errors and negative responses appear in red
- User input appears in white

D. Help command

All supported serial commands, as well as arguments that can be passed with them, can be printed with the `Help` command. A one-line description of the command is also printed.

E. Setting the Time

The internal real-time clock and calendar (RTCC) can be set over the USB interface. The real-time clock and calendar tracks time, date, and weekday.

All RTCC settings can be adjusted using the `Set RTCC` command, which requires an argument passed for what should be adjusted, along with the value for what' should be adjusted. For example:

`Set RTCC: Time: 09:30:25` would set the current time to 9:30 and 25 seconds, AM

`Set RTCC: Time: 21:30:25` would set the current time to 9:30 and 25 seconds, PM (regardless of time format displayed, RTCC time must be entered in a 24-hour format)

`Set RTCC: Date: 04/24/2023` would set the current date to April 24th, 2023 (only years between 2000 and 2099 are supported. All four digits of year must be entered)

`Set RTCC: Weekday: Friday` would set the current weekday as Friday. The RTCC weekday is set with a string, while it is displayed on the Nixie tubes as a number (1 to 7)

F. Selecting what is Displayed

The clock can be configured to display time information through the USB interface.

`Set Display Mode: Time` would configure the clock to show the current time (most use cases)

`Set Display Mode: Date` would configure the clock to show the current date

`Set Display Mode: Weekday` would configure the clock to show the current weekday (numbered 1 through 7)

`Set Display Mode: Alarm` would configure the clock to show the time that the alarm is set for

G. Setting the Alarm

`Set Alarm: 09:30:25` would set the alarm time for 9:30 AM and 25 seconds. This must be entered in 24-hour time format. This does not actually arm the alarm to sound.

H. Arming the Alarm

`Arm Alarm: Arm` would enable the alarm to sound at the time set by the `Set Alarm` command. Once the alarm sounds, it can be disarmed/silenced by pressing any pushbutton, or by sending the `Arm Alarm: Disarm` command. There is no snooze function.

I. Setting the Tube RGB Backlight

`Set Backlight Color: Red` would set the RGB tube backlight, if installed, to be a solid red color. Supported colors include Black (off, default), Red, Green, Blue, Cyan, Magenta, Yellow, and White. Any 24-bit RGB hex code can also be sent for custom colors (FFFFFF would correspond to white, for instance)

`Set Backlight Brightness: 90` would set the RGB tube backlight, if installed, to a brightness level of 90%.

J. Turning the clock On and Off

`Set Power: On` enabled the clock. `Set Power: Off` disables the clock. Whenever the clock is turned on, Time is always selected to be displayed, and the RGB backlight is disabled.

K. Supported Serial Commands

All supported serial commands and their help messages are listed below:

```
    Help: Prints help message for all supported serial commands

    Reset: Executes an MCU software reset

    Clear Screen: Clears the serial port terminal

    *IDN?: Prints identification string

    Repository?: Prints project Git repo location

    Host Status?: Prints status of MCU host device (IDs, WDT, DMT, Prefetch, Cause
of Reset, up time)

    Peripheral Status? <peripheral_name>: Prints status of passed host peripheral.
Available peripherals:

        Interrupts

        Clocks

        PMD

        WDT

        DMT

        Prefetch

        DMA

        ADC

        ADC Channels

        I2C Master

        RTCC

        Timer <x> (x = 1-9)

    Error Status?: Prints the status of various error handler flags

    Clear Errors: Clears all error handler flags

    Platform Status?: Prints current state of surrounding circuitry, including
PGOOD, clock elapsed time, I2C slaves
```

Live Telemetry: Toggles live updates of system level telemetry

Time and Date?: Prints the current system time and date

Set RTCC: <parameter>: <parameter args>: sets a time parameter within the Real Time Clock and Calendar. Available parameters:

Date: <mm>/<dd>/<yyyy>: Sets the RTCC date

Time: <hh>:<mm>:<ss>: Sets the RTCC time. (Must be 24 hr time format)

Weekday: <weekday>: Sets the RTCC weekday

Unix Time: <decimal unix time>, <hour offset from UTC to local time>: sets the RTCC to the supplied UNIX time with hour offset from UTC

Set Backlight Color: <color/hex>: Sets the tube backlight color. Colors include Red, Green, Blue, Yellow, Magenta, Cyan, White, and any 24 bit hex color (eg FFFFFF)

Set Backlight Brightness: <percentage>: Sets the brightness of the tube backlight

Set Power: <On/Off>: Turns the clock on or off

Set Display Mode: <Time, Date, Weekday, or Alarm>: Sets the display to show different clock functions

Display Lamp Test: Tests all IN12 display Elements. This should look like a giant blur on each digit

Set Display Brightness: <percent>: Sets the IN12 display to the entered brightness as a percentage

Set Time Format: <24/AM_PM>: Sets time display format. This only impacts tube display, not USB user interface

Alarm Status?: Prints clock alarm settings

Set Alarm: <hh>:<mm>:<ss>: Sets the alarm time. (Must be 24 hr time)

Arm Alarm: <Arm/Disarm>: Arms or disarms the clock alarm

IN-12 Status?: Prints status of devices on IN-12 Carrier Board, as well as carrier SPD data. This includes tube elapsed time counter.

6. Project History

A. Rev A

The first incarnation of this project was completed in October of 2017. It ran on a PIC16LF1519 8-bit microcontroller, supporting IN-14 tubes. It featured a 1.65F supercapacitor stack backup circuit (charge pump, LDO) that allowed the PIC16 to continue timekeeping for up to a day after being unplugged. The PIC16 utilized extensive sleep modes and low power techniques to continue operation with this supercapacitor stack. This design was only able to display and track time, it could not track date or weekday, and did not have an alarm. Time was tracked directly in RAM as standard C variables. Due to a low pin count on the PIC16, binary coded decimal encoding (and hardware decoding) was used to control all the multiplexing signals. Like the rev B design, the rev A was constructed using two printed circuit board. The top circuit board was entirely passive, only hosting the tubes. All power, processing and multiplexing was done on the lower PCB. There was no USB port, telemetry, or debug functionality. Also like the rev B design, the rev A featured an input protection circuit for under and overvoltage lockout, and reverse polarity protection. It sports a similar, yet less refined boost converter design to the rev B project. Due to the extreme step-up ratio required to boost the +24V input to the +180V strike voltage, the boost converter is a hybrid topology, with a standard non-synchronous boost design, but with a diode/capacitor voltage doubler added to the switch node. The regulator regulates to +90V, and the diode/capacitor double doubles this to +180V strike voltage. Both happen to be needed to properly drive nixie tubes. The upper and lower PCBs were connected with three different 0.1” headers, which made mating the two PCBs difficult within the enclosure.

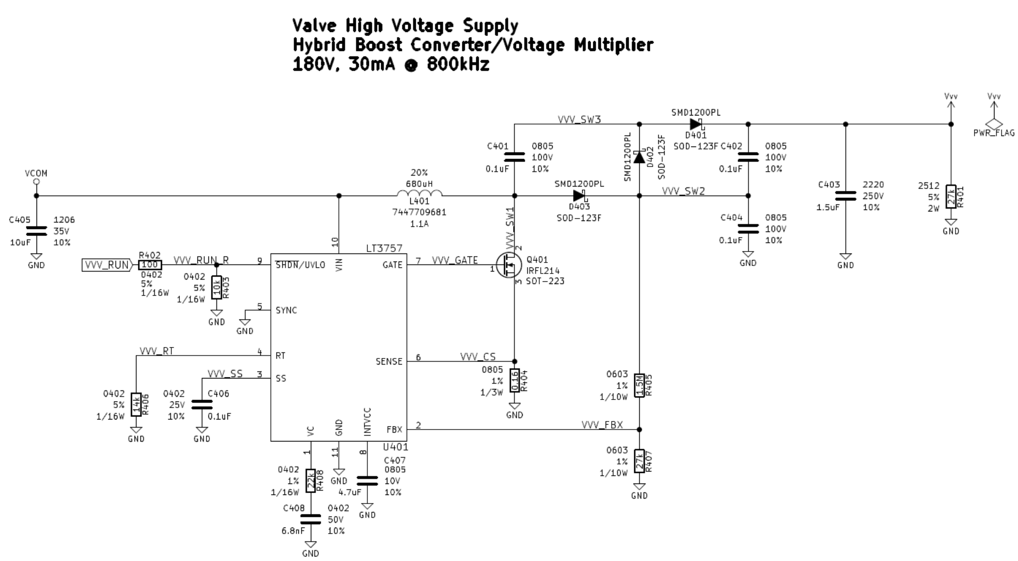

FIGURE 1: REV A BOOST CONVERTER

B. Rev B

The rev B design has many similar elements to rev A, but has been extensively redesigned. The boost converter remains, although with more refined component selection. This version runs on a PIC32MZ EF 200MHz 32-bit microcontroller, which is extremely, extremely overkill. It is designed to be much more flexible from a user-interface and functionality standpoint, as well as from a configuration standpoint. Power and processing live on the lower PCB, but multiplexing and user interface support live on the upper PCB. This means that multiple tube types can be supported by swapping out the upper PCB. A single tall stacking height PMC connector bridges the two PCBs.

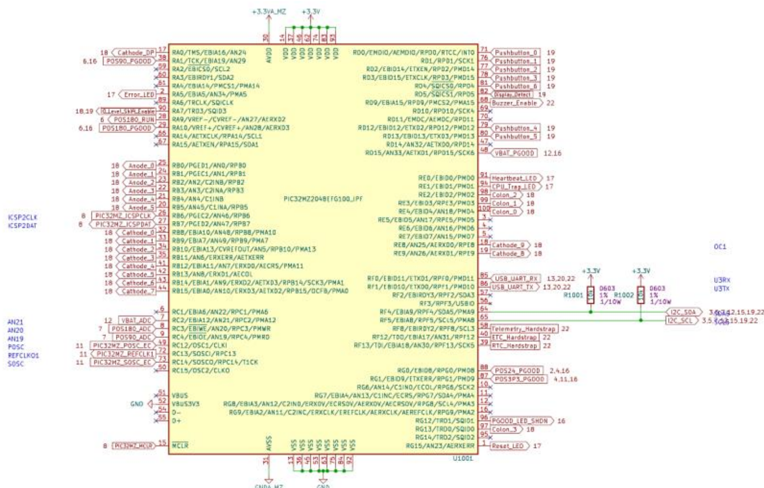

Figure 2: Rev B Microcontroller

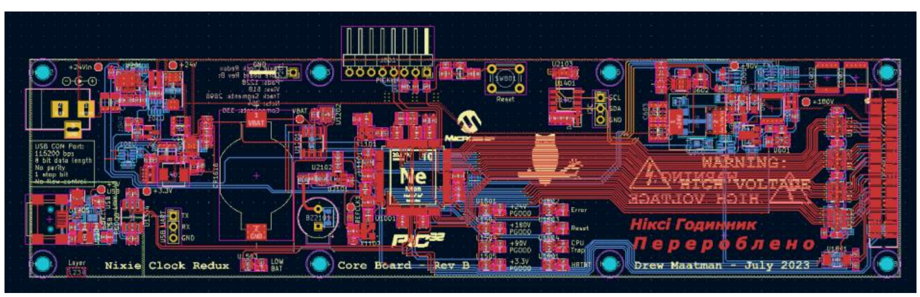

Figure 3: Rev B Lower/Core PCB

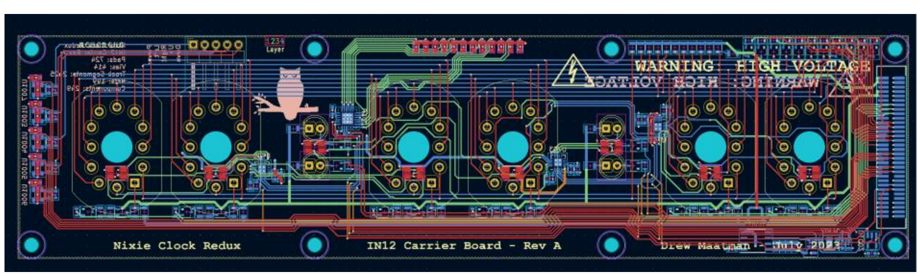

Figure 4: Rev B Upper/Carrier PCB (IN-12 Variant)

7. Nixie Tube Working Principle

A. (Not) a Vacuum Tube

Despite appearing like a vacuum tube, Nixie tubes do not contain a vacuum. Sealed inside the tube is neon gas. This is what gives nixie tubes their orange glow when energized. Some more modern tubes also contain a mixture of mercury, which reduces sputtering – see below. This mercury creates a small blueish glow along the cathode in addition to the orange glow due to neon ionization.

B. Ionization

Nixie tubes have anodes and cathodes, much like modern PN junction semiconductors. When a high voltage is applied from anode to cathode, electrons are stripped from the valence electron bands of the neon gas inside the tube. When a neon atom is stripped of an electron, it becomes a positive charged ion (ionization), which wants to flow towards the cathode – which is at a lower voltage potential. This is similar in principle to a neon tube sign. When these neon atoms hit the cathode, metal atoms are displaced from the cathode into the space within the tube (called sputtering). This is what forms an electric current, and allows charge to flow. This electric current is relatively small (~2.5mA). The light within a nixie tube is emitted through both ionization and sputtering.

C. Common Anode

Much like some LED common-anode seven segment displays, most Nixie tubes have a single anode, constructed as a mesh, which has the high voltage applied to it through a single current limiting anode resistor. To illuminate a number within the tube, one of many cathodes is grounded, or driven to 0V. This creates the voltage potential needed for neon ionization. Most tubes have 10 cathodes and a single anode – the 10 cathodes correspond to the digits 0 to 9. The cathodes are physically arranged front to back within the tube, so that when any cathode is illuminated, they appear in the same location. Some tubes also have a decimal point cathode, which typically is rated to a much lower current. Other more rare tubes have common electrical symbols such as Ω or µ, since Nixie tubes were used within electrical measurement equipment. IN-12 tubes have their cathodes in the order 3 8 9 4 0 5 7 2 6 1. The digit 5 is an upside down 2 to reduce unique materials. The cathodes are arranged in this manner to minimize more forward cathodes from obscuring the rear cathodes.

D. Cold Cathode

Nixie tubes do not heat up very much. There is no heating element within the tube, unlike most vacuum tubes electrical engineers would be familiar with. Other graphical tubes, such as VFDs, do require a heating element. A lack of a heater does not mean they are efficient, though. IN-12 nixie tubes dissipate approximately 0.36W each when a single cathode is lit.

E. Strike Voltage

To light a nixie tube, a high voltage must be applied. This first voltage application is known as the strike voltage. This clock design utilizes a strike voltage of +180V, which is the output of an internal boost converter on one of two internal printed circuit boards. This high DC voltage gives the neon atom valence electrons the motivation they need to be stripped from the neon atoms, turning the atoms into neon ions. The nixie tube sustains the strike voltage across its anode to the selected grounded cathode for about 100µs before the neon gas is sufficiently ionized to begin conduction. There is no current flow during this time.

F. Sustain Voltage

After about 100µs of strike voltage application, the voltage across the tube drops to about 140V. This is called the sustain voltage. Conduction begins and the cathode begins to glow. The tube will hold the sustain voltage across it if the cathode is grounded and the anode voltage (pre-resistor) is sustained.

G. Negative Resistance

The drop from strike voltage to sustain voltage, and how this corresponds to an increase in current draw (from 0mA to around 2.5mA when conduction begins) gives a nixie tube a *negative resistance* characteristic, since voltage drop is reducing while current draw is increasing.

H. Multiplexing

Only one nixie tube within this clock is on at a time. This is because the circuitry required to have all tubes on at the same time would be much more complex and require a set of 10 or 11 high voltage bipolar transistors for each tube, which is not economical. In addition, nixie tubes have limited life. Having all the tubes on constantly would wear down the lifetime more quickly.

Instead, a multiplexing technique is used. Each tube has its own anode driven individually, and the cathodes for each tube are all tied together. To turn one number in one tube on, the anode for that tube has +180V applied to it with a PNP transistor, and the shared cathode signal for the number desired is grounded with an NPN transistor.

The timing for multiplexing is carefully controlled such that all 6 tubes (and two sets of neon bulb colons) are cycled through in $1/60^{th}$ of a second, to give a display refresh rate of 60Hz. The changes are too fast for the human eye to distinguish, the human brain blends the light for all tubes together, so that all digits appear to be on at the same time.

The timing is generated with hardware timers within the microcontroller (brains of the clock), and the code is interrupt based, meaning that the timing is deterministic, accurate and exact.

I. Anti-Ghosting

Nixie tubes are highly capacitive. Because the anodes and cathodes are driven with either an NPN or PNP transistor (open-collector or open-emitter instead of a push-pull or totem pole circuit topology), when a tube is energized, it holds residual charge on the anodes and cathodes (instead of being discharged to ground). If this charge has nowhere to bleed to, it will move to the subsequent tube that is next in the multiplexing sequence. This would cause adjacent tubes to appear to have the same numbers falsely illuminated. This phenomenon is called ghosting. The cleanest approach to mitigating ghosting it to terminate the anode and cathode signals that are shared across all tubes together through a high value resistor. This “termination” connection is then tied to half of the strike voltage, or +90V. This is a low enough voltage to not energize the tube, but high enough to minimize switching losses when turning tubes on and off. The boost converter which generates the +180V strike voltage was also carefully designed to break out half of the output voltage magnitude for this termination level. This is a similar working principle to termination in DDR4 memory in modern computer design, just on a roughly ~100x voltage magnitude scale, and much, much slower in frequency.

8. Nixie Tube History

A. From Wikipedia:

“Nixie tubes were invented by David Hagelbarger. The early Nixie displays were made by a small vacuum tube manufacturer called Haydu Brothers Laboratories, and introduced in 1955 by Burroughs Corporation, who purchased Haydu. The name Nixie was derived by Burroughs from "NIX I", an abbreviation of "Numeric Indicator eXperimental No. 1", although this may have been a backronym designed to justify the evocation of the mythical creature with this name. Hundreds of variations of this design were manufactured by many firms, from the 1950s until the 1990s. The Burroughs Corporation introduced "Nixie" and owned the name Nixie as a trademark. Nixie-like displays made by other firms had trademarked names including Digitron, Inditron and Numicator. A proper generic term is cold cathode neon readout tube, though the phrase Nixie tube quickly entered the vernacular as a generic name.”

B. In Soviet Russia:

C. IN-12A/B: