# TrapBox Overview

## Power Management

The TrapBox Board is intended to run off of battery power, and must consume a minimum of current when not powered on. The TrapBox Board (TBB) is powered up by pressing the Power On button (S1 on schematic). This will pull down the base of PNP transistor Q3, turning it on which will provide voltage to the Enable pin of voltage regulator IC1. When enabled, IC1 will power up microcontroller U1. One of the first things that U1 does when it starts executing code is to turn on the gate of Q4. Q4 will take over pulling down the gate of Q3 after the Button has been released. Powering off the TBB is done by pressing the Reset button (S2), which will cause the microcontroller to release the gate of Q4 long enough for the regulator IC1 to turn off.

## TBB Timing

The TBB acts as the Primary Transmitter (PTX) in the link between the TBB and the Remote. The microcontroller’s internal Timer2 is used to generate an interrupt once ever millisecond, and this interrupt is used to control the flow of the PTX. Within the ISR for the millisecond interrupt, a variable is decremented once/mS. When that variable gets to zero, it is reset to 50 and the decrementing repeats. This provides a 50mS loop which is used to control the sending of data to the remote (this occurs just after the 50mS variable gets to 0) and firing the ultrasonic transmitter (this occurs in the middle of the 50mS loop, at 25mS).

## Ultrasonic signal transmit/receive

When the above-mentioned 50mS loop gets to the halfway point (25mS), it calls a function to fire the ultrasonic transmitter. This starts a state machine that will last for a few milliseconds (maximum ~8.2mS) and will generate an 8-bit variable that represents the sonic travel time between the TBB and the nearest obstacle.

The state machine initially starts Timer1 to generate a 40kHz drive signal on PB2 (XMIT\_DRV). This will drive U2/U3, which are complementary MCP1401/MCP1402 gate drivers. These gate drivers will drive the Ultrasonic Transmitter ‘TX’ with a 40KHz signal of roughly twice the battery voltage. Every time the output drive toggles there is an interrupt generated. Within the ISR, the number of pulses is counted. After 8 pulses of the 40kHz drive, the output is turned off and Timer1 is reconfigured to generate an interrupt every 32uS. Ultimately the software will count the number of 32uS ticks between firing the transmitter and when a signal is detected at the receiver and use that to determine distance. 32uS was chosen as it’s plenty of resolution for this application and it will keep the count rate reasonable. Since the round-trip slowness of sound in air is 148uS/inch, every 4.625 counts of the 32uS interrupt = 1inch…Or to put that differently, the sum of 32uS ticks divided by 4.625 = the distance in inches. Immediately after firing the transmitter, the receiver will be seeing some signal that’s not legitimate (i.e. it’s directly coupled from the transmitter). This signal must be ignored and so the microcontroller will wait for 30 ticks of the 32uS interrupt before looking for a return signal. This means that the minimum distance that can be detected is (30/4.625 = ~6.5inches). Any distances shorter than this will appear as 6.5”. After 30 ticks have passed, the micro will enable an interrupt tied to the receiver amplifier/detector circuit. When a large enough signal is output from IC3A, it will bias the base of T3 enough to turn it on, pulling down the junction of T3/R31, RCVR\_IN. A low level on RCVR\_IN will trigger an interrupt if it occurs while the micro is listening for a receive signal. Once that interrupt occurs, the micro will store the total number of 32uS ticks in a register and that register will be sent to the Remote Box with the next transmission.

In the name of simplicity, the register that counts the number of timer ticks is 8-bit. This means that the maximum number of timer ticks that can be stored is 255, and so the maximum distance that can be recorded is (255/4.625 = ~55inches = ~4.6 feet). This could be extended if need be, but in the realm of live animal traps 4.6’ is more than sufficient. If the register that counts 32uS ticks reaches 255 before a receive interrupt is triggered, the register is set to 0, to indicate that nothing was returned to the receiver in the allotted window. LED2 on the TBB board will light up if a received signal comes in before the timer times out, or else the LED will be turned off.

Regardless of whether the ultimate number of 32uS ticks is due to a signal detected at the receiver or due to the receiver timing out and setting it to 0, after that ultimate value has been determined Timer1 is configured to generate 40Khz pulses so it is ready for the next time it gets activated.

## Data Transmission

The TBB acts as the PTX (Primary Transmitter) in the NRF24L01+ link between the TBB and the Remote Box Board (RBB). Every 50mS, the TBB will call functions to acquire information from the ultrasonic transducer (the number of 32uS rollovers) and an ADC value representative of the voltage of the TBB battery. These two 8-bit values, along with a counter variable (cnty) are transmitted via NRF link to the RBB. After every data transmission, cnty is incremented by 1. When it gets to 255, it will roll over and start counting again. This is a simple way to scan for dropped packets by ensuring that the cnty variable on the RBB only increments by 1 every time data is received.

The NRF module is configured to re-transmit data up to 7 times if it doesn’t receive an acknowledgement packet (Ack-Pac) from the NRF module on the RBB. This is done without intervention from the TBB after initial configuration. After initializing a transmission to the RBB, the TBB will wait for the NRF module to either receive an Ack-Pac from the remote, or reach its maximum number of retries and give up. If it receives an Ack-Pac, it will scan a bitfield sent from the RBB, paying attention to 3 bits. These will let the TBB know if it should enable/disable the on-board LED, activate the Auxiliary Output, or activate the trap’s door in order to close it.

## On-board LED

The on-board LED is a Lumileds Rebel Power White LED. Its anode is connected to battery input via a current limiting resistor while its cathode is tied to the drain of an N-Channel FET. When the TBB receives a signal telling it to enable the LED, the microcontroller on the TBB will turn on the gate of FET Q5, turning on the LED.

## Auxiliary Output

The Auxiliary Output was initially intended to activate a camera’s shutter release, but can be configured to perform other tasks fairly easily. When the TBB detects a command to activate the Aux Output, it will check and see if the output is busy. If it is already busy, the TBB will ignore the command. If it is not busy, the TBB will activate the ‘Shutter’ output which will turn on NPN-BJT T2, and it will set up the millisecond timer to generate a callback to ‘camera done’ in 1 second. When 1 second has passed, the ‘camera done’ function will turn off T2 and clear the flag stating that the Aux output is busy, freeing it up for the next call to activate the output.

## Set Trap

This document will discuss the later versions of TBB where the ‘Set Trap’ functionality is done via a solenoid. In earlier versions of TBB, setting of the trap was accomplished via hobby servo. The solenoid is an all-around better option for simplicity, reliability and cost. Stepping up of the battery voltage to a level high enough to drive the solenoid is accomplished via a boost converter driven by Timer0. When the TBB sees a command to set the Trap, it first makes sure that the Set Trap function isn’t currently active. If it is active, the new command is ignored, otherwise it will enable the output of Timer0-B/PD5. This output is a PWM signal with a 40% DC and a frequency of ~131kHz, which will drive a gate driver that in turn controls a FET to switch the Boost Converter. As the microcontroller generates the PWM signal to drive the Boost Converter, it counts how many times the FET on the converter has been pulsed. It initially pulses the FET 15 times while there is no load on the output of the Boost, which allows the voltage to rise to ~16VDC, and this voltage is connected to one side of the solenoid coil. After 15 pulses, the micro enables output PD6 which will turn on a FET that connects the other side of the solenoid coil to ground, in order to activate the solenoid/close the trap. PD6 is kept on for ~500mS, after which both FET’s are turned off and the solenoid is released.