

# RMG PROGRAMMABLE LO

for

902 MHz TRANSVERTER

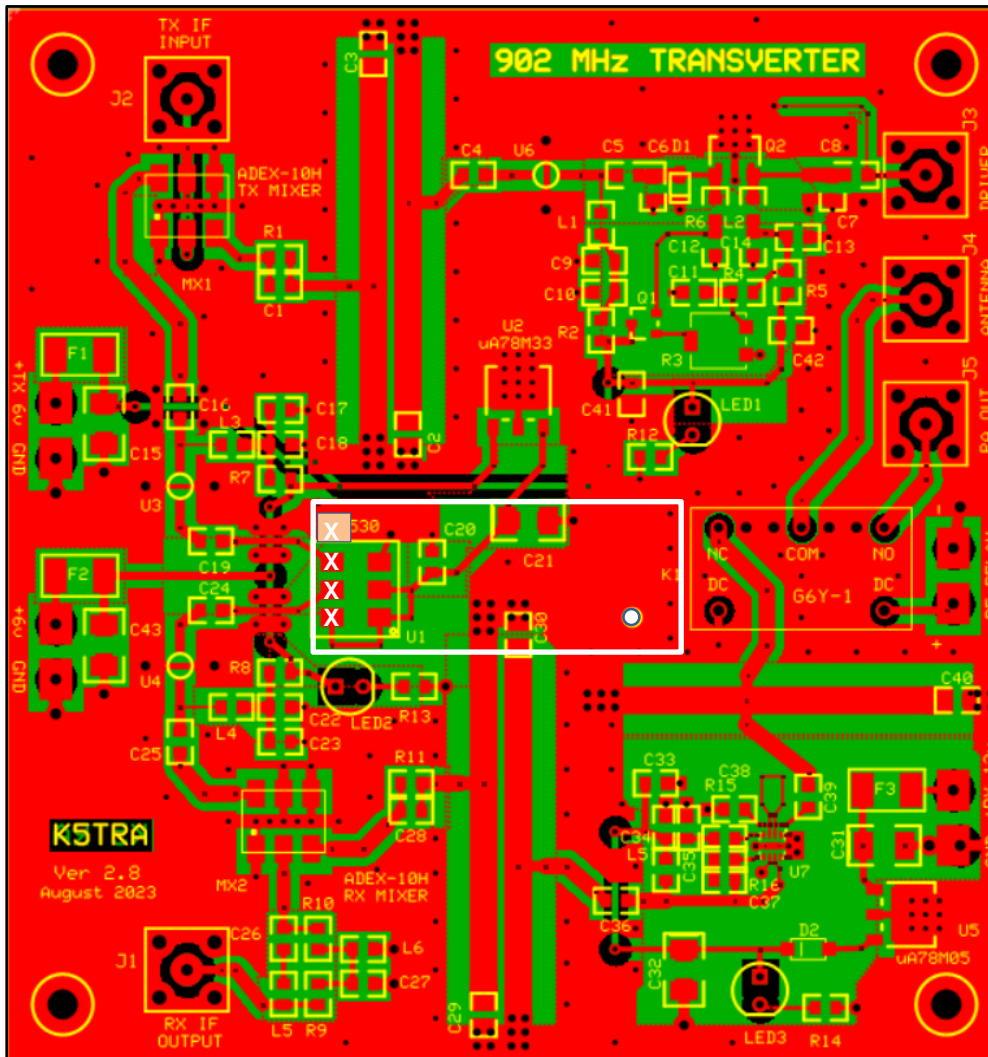
*[and other transverters & beacons]*

*AB5SS / KE0FF / K5TRA*

# HOW IT WORKS

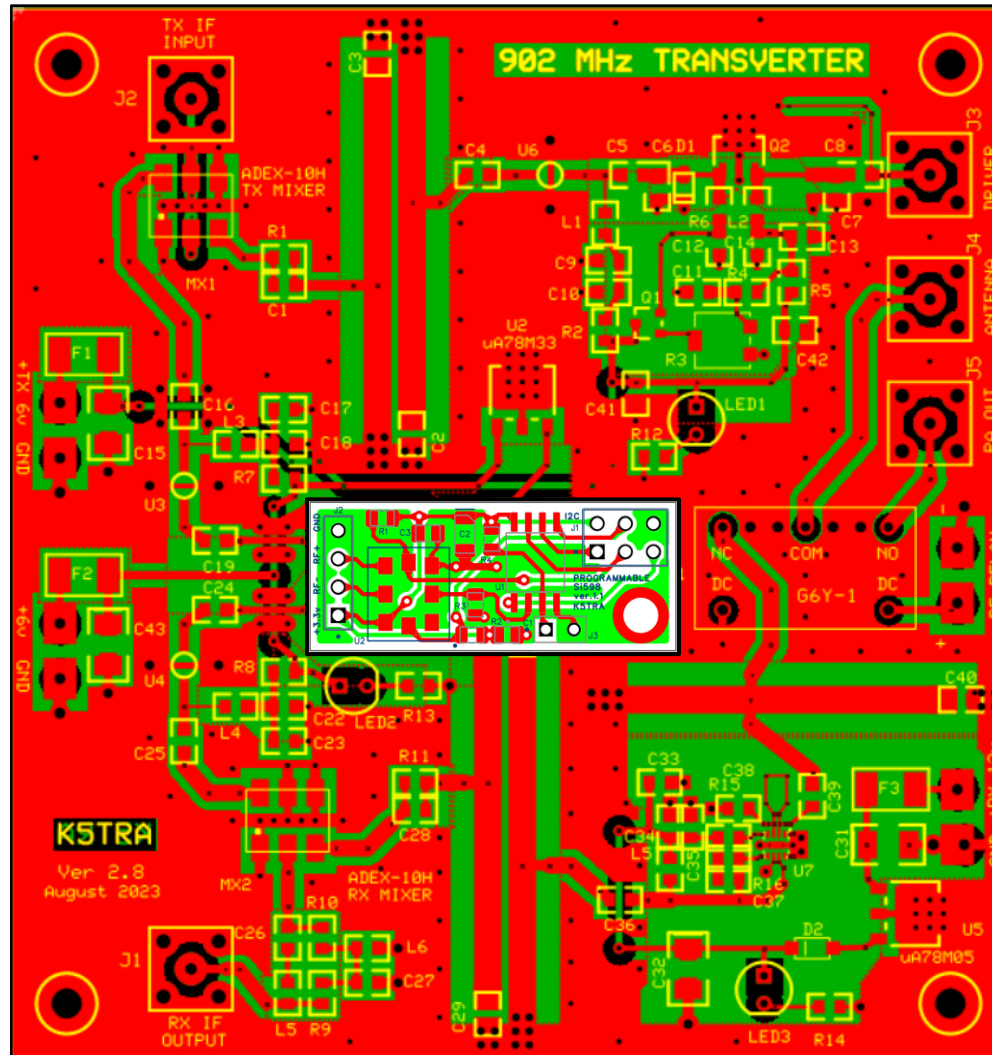
- Si598 programmable oscillator
- When powered, the Si598 must be programmed to the desired frequency.
- ATtiny85-20 microcontroller communicates via I2C with the Si598
- Set the frequency and then go to sleep
- A second frequency can be programmed and selected on control line.

# LO BOARD CONTACTS & MOUNTING



- 4-PIN header below LO board contacts RF board
- Ground contact (top): remove (scrape) solder mask
- 4 PINs are soldered to pads on RF board
- Mounting hole to 4-40 standoff and screw through RF board

# LO BOARD ON TRANSVERTER RF BOARD

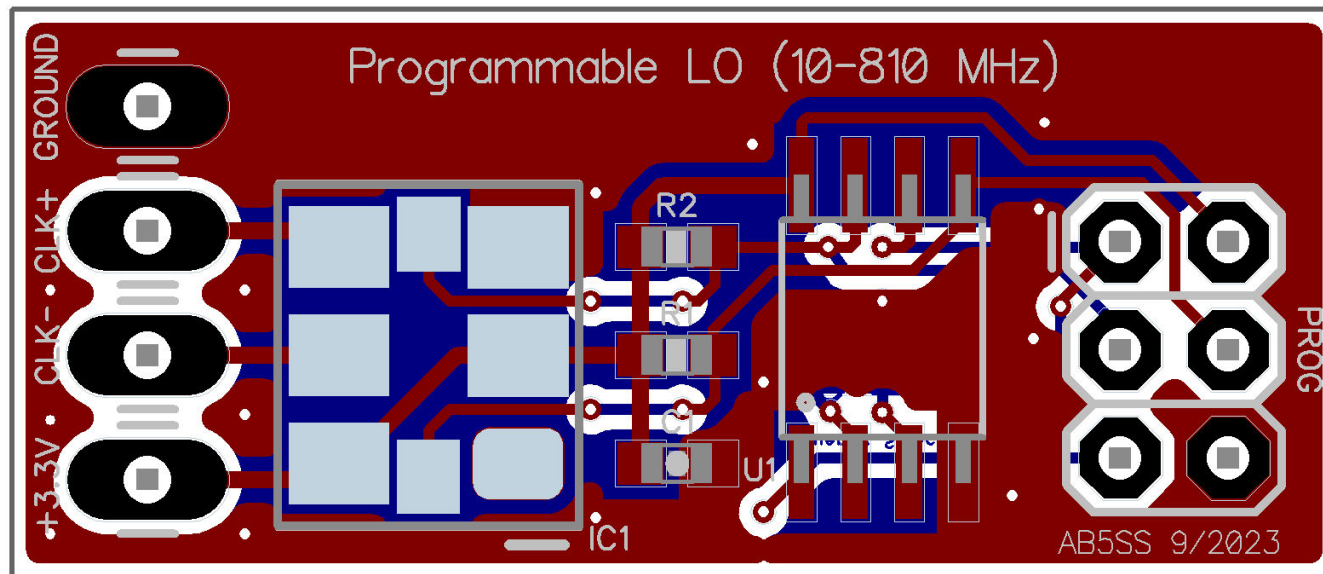


# EVOLUTION OF THE LO BOARD

- John Maca (**AB5SS**) deserves huge props for the original idea of using an ATtiny85-20 MCU and a Si598 programmable LO for the 902 MHz RMG transverter, as an alternative to the factory programmed Si530.
- The original prototype board was done by John (using Eagle) and the original code to program it, in Arduino IDE.
- ATtiny85 MCU programming can be done with either Arduino IDE or Atmel/Microchip Studio.
- The AVR ISP MKII programmer can be setup to work with either IDE or Studio; not both on the same PC.
- Joe Haas (**KE0FF**) has written code for the Atmel (Microchip) Studio to program the ATtiny85-20 and Si598 circuit. Joe did an outstanding job.
- Tom Apel (**K5TRA**) has created two boards (using DipTrace). The same ports on the MCU were used for I2C lines to Si598 (SDA and SCL) as in John's prototype board; so, Joe's code in Studio will work with all of the boards. That code also controls the Si598 'Output-Enable' and provides for a second frequency.

# AB5SS PROTOTYPE BOARD

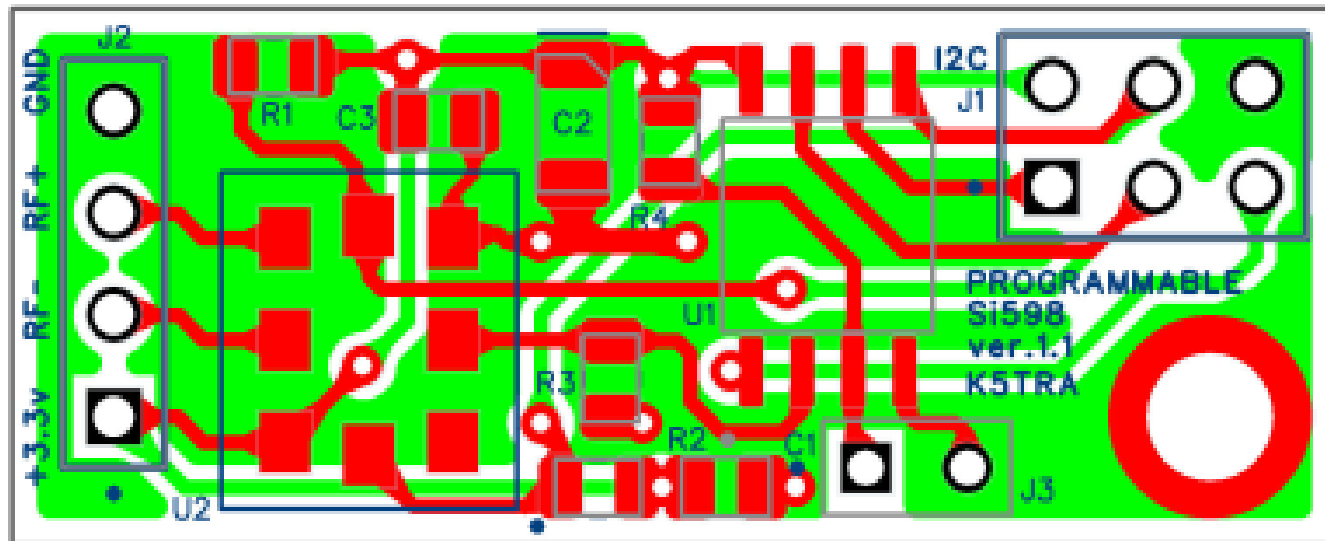
- Output pin pitch is 0.115".
- One 0.1uF V+ bypass.
- Potential future control pads on backside of board.
- Bare-bones circuit.



(1.225" x 0.525")

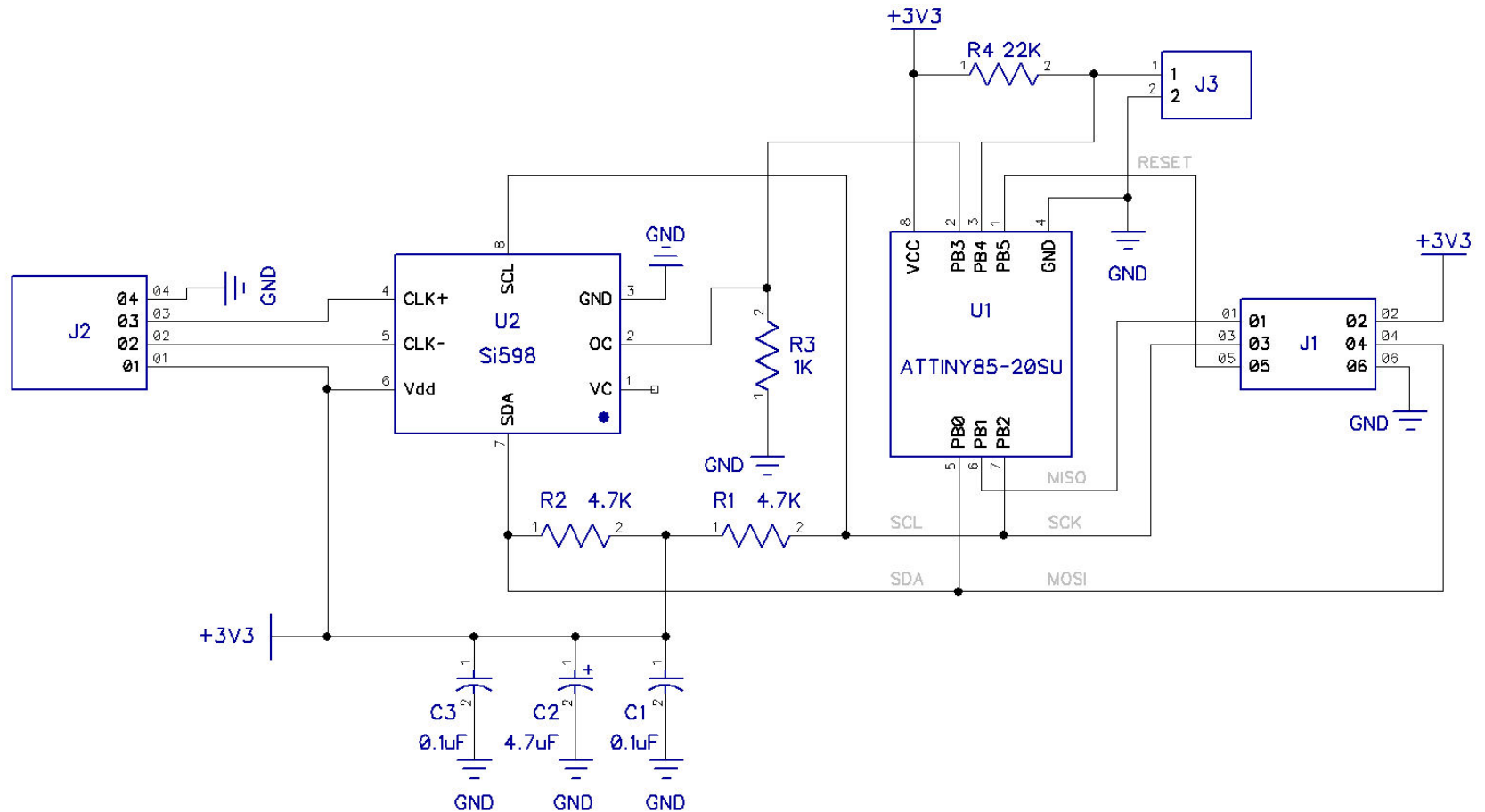
# 902 MHz TRANSVERTER LO DAUGHTER BOARD

- Output pin pitch matches the Si530 pad pitch (0.100") with a 4-pin header attached to the bottom side of the board.
- Mounting hole for 4-40 screw to standoff (to transverter RF board).
- More bypass capacitance on V+.
- Si598 Output-Enable is also controlled by the MCU (PB3).
- A selectable second frequency is provided through PB4.



(1.300" x 0.525")

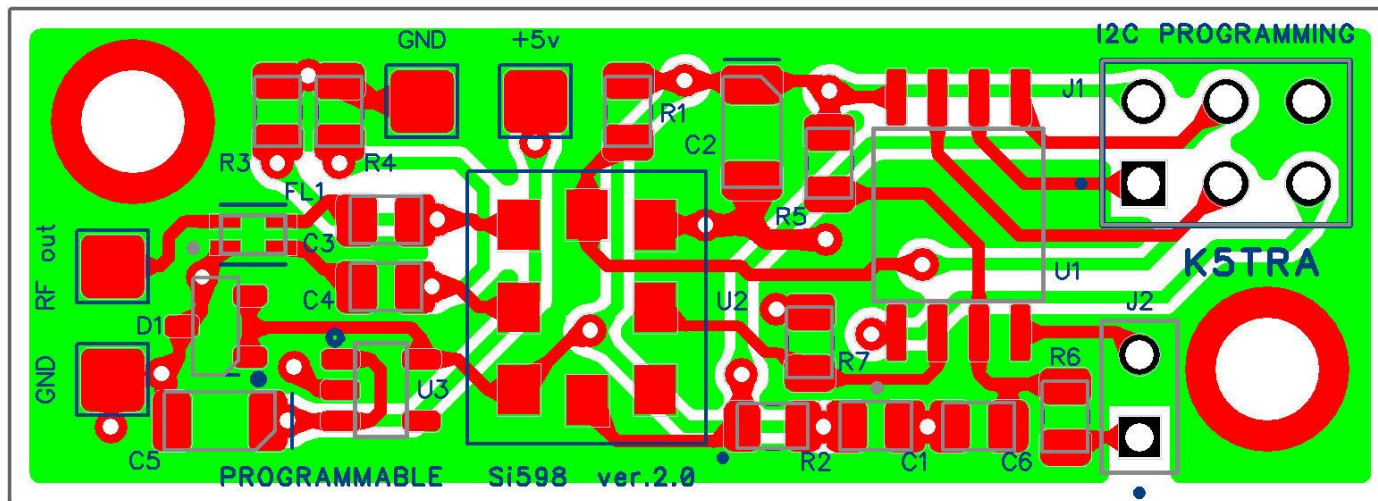
# LO DAUGHTER BOARD SCHEMATIC





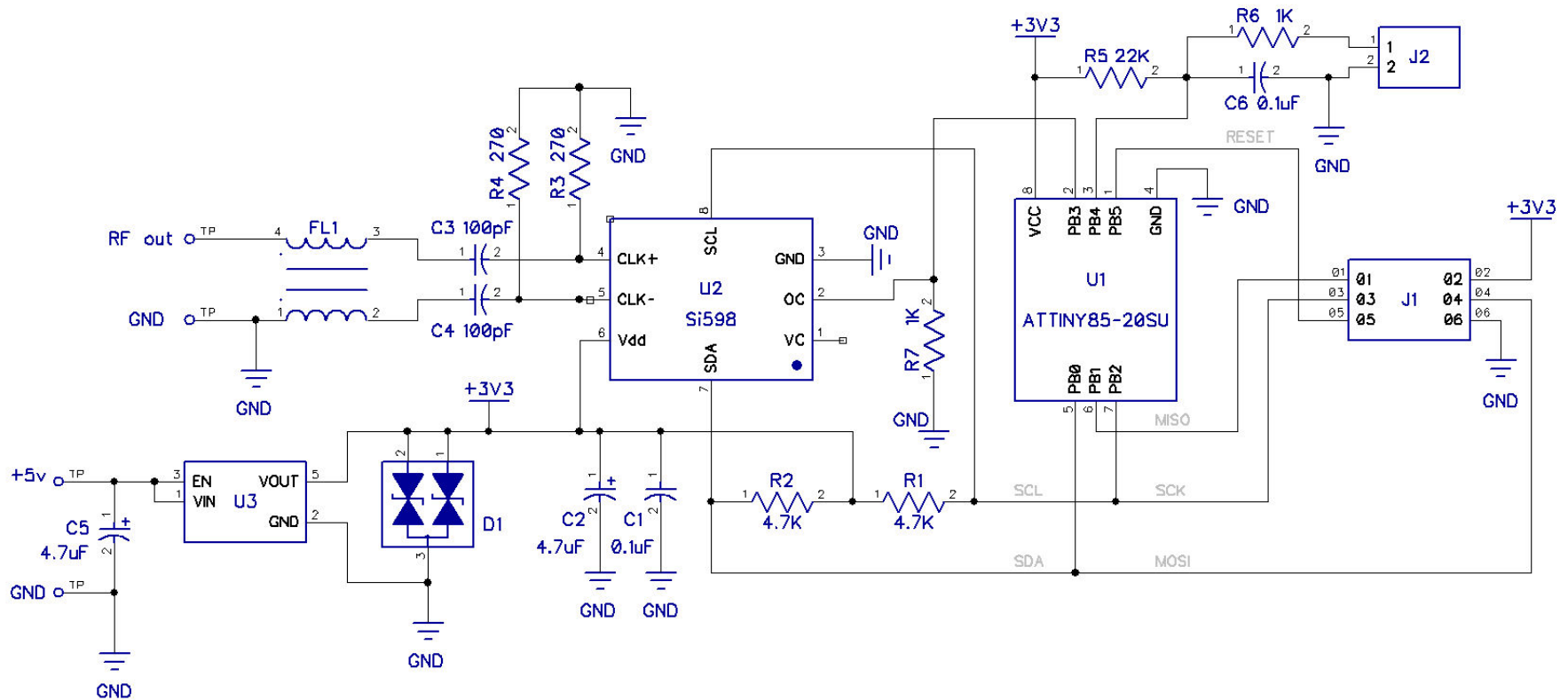
# GENERAL PURPOSE 2-FREQUENCY BOARD

- Differential output is coupled to single-ended output with DC blocking capacitors and balun.
- Onboard LDO (and very quiet) 3.3v regulator. This will be low dissipation when powered from external +5v.
- ESD protection on +3.3v bus.
- Two mounting holes for 4-40 standoffs.
- Si598 Output-Enable is also controlled by MCU (PB3).
- A selectable second frequency provided through PB4. Additional RC filtering on control interface (J2, 2-pin header) to assist software de-bounce.

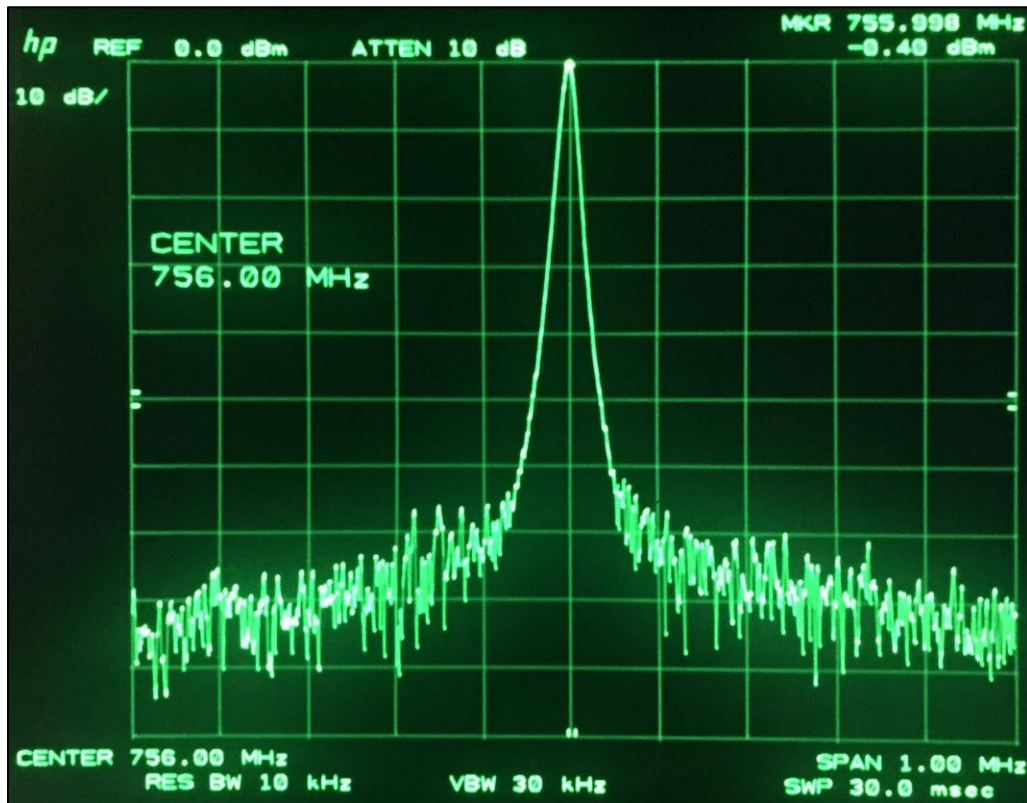


(1.675" x 0.600")

# GENERAL PURPOSE BOARD SCHEMATIC

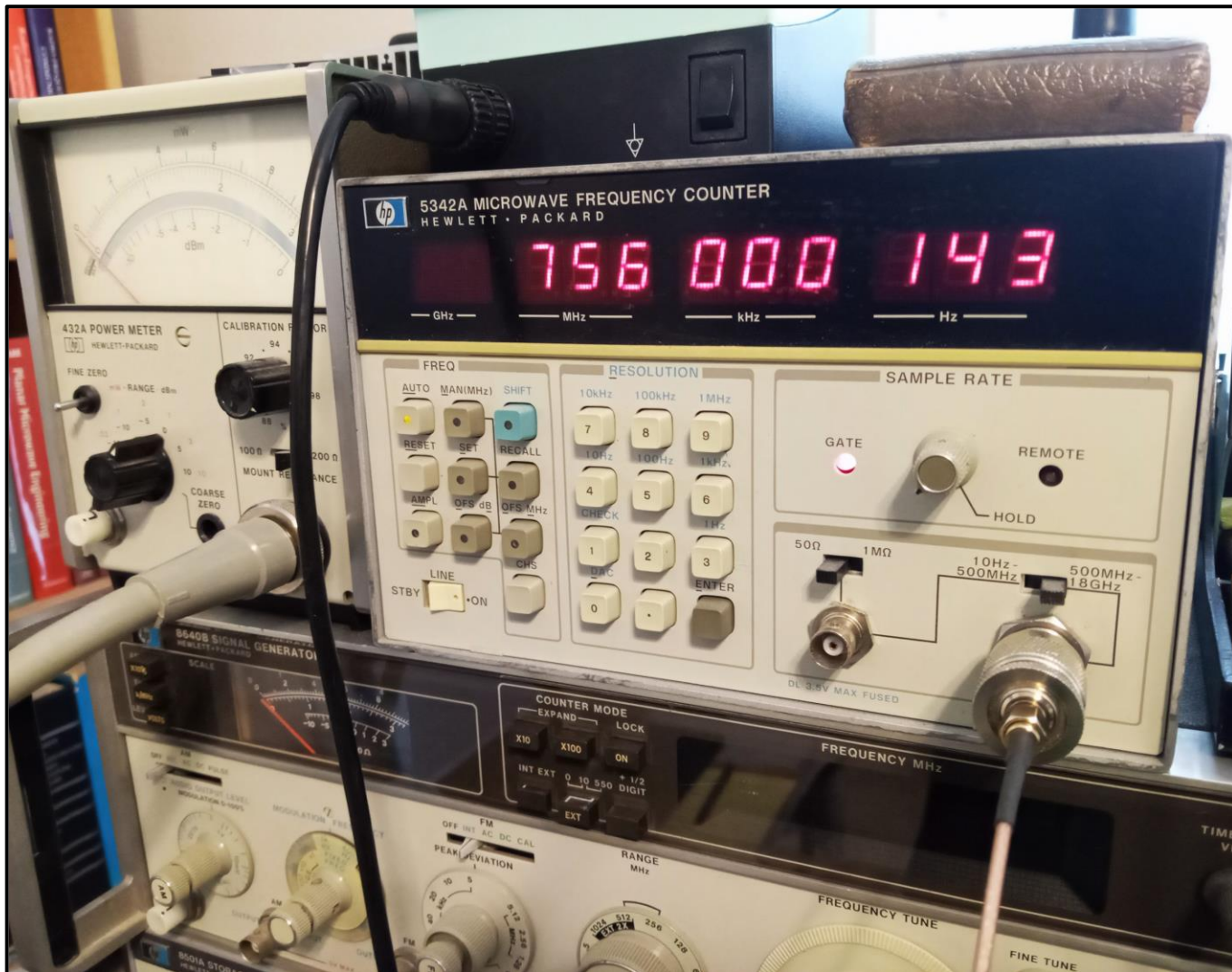


# TEST BREADBOARD OUTPUT SPECTRUM



- Breadboard yields ~ 0 dBm single-ended output (each side of differential output)
- Programming can be “tweaked” to tune the frequency to very tight tolerance

# TEST BREADBOARD OUTPUT FREQUENCY



# I2C SERIAL PROGRAMMING

## 756 MHz - CHANNEL 1 (902 MHz to 146 MHz)



## 758 MHz - CHANNEL 0 (902 MHz to 144 MHz)





# ATtiny85 FUSE SETTINGS

AVRISP mkII (00B018070B86) - Device Programming

Tool: AVRISP mkII | Device: ATtiny85 | Interface: ISP | Apply | Device signature: 0x1E930B | Read | Target Voltage: 3.2 V | Read | ⚙️

Interface settings  
Tool information  
Device information  
Oscillator calibration  
Memories  
**Fuses**  
Lock bits  
Production file

Fuse Name	Value
✓ EXTENDED.SELFPRGEN	<input type="checkbox"/>
✓ HIGH.RSTDISBL	<input type="checkbox"/>
✓ HIGH.DWEN	<input type="checkbox"/>
✓ HIGH.SPIEN	<input checked="" type="checkbox"/>
✓ HIGH.WDTON	<input type="checkbox"/>
✓ HIGH.EESAVE	<input checked="" type="checkbox"/>
✓ HIGH.BODLEVEL	Brown-out detection disabled ▾
✓ LOW.CKDIV8	<input checked="" type="checkbox"/>
✓ LOW.CKOUT	<input type="checkbox"/>
✓ LOW.SUT_CKSEL	Int. RC Osc. 8 MHz; Start-up time PWRDWN/RESET: 6 CK/14 CK + 64 ms ▾

Fuse Register	Value
EXTENDED	0xFF
HIGH	0xD7
LOW	0x62

☒ Auto read  
☒ Verify after programming

Copy to clipboard

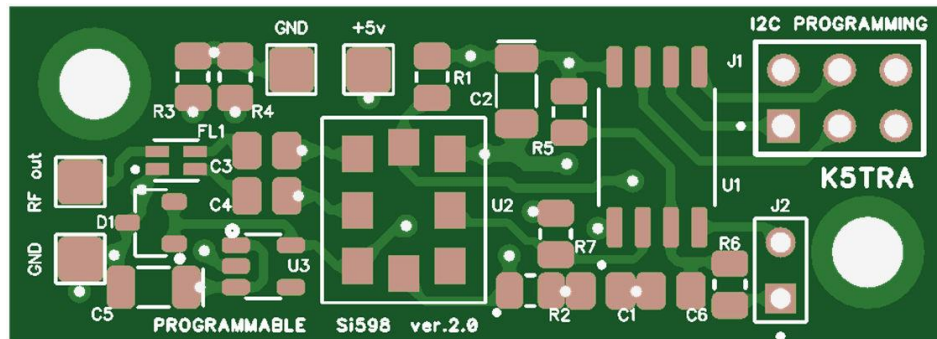
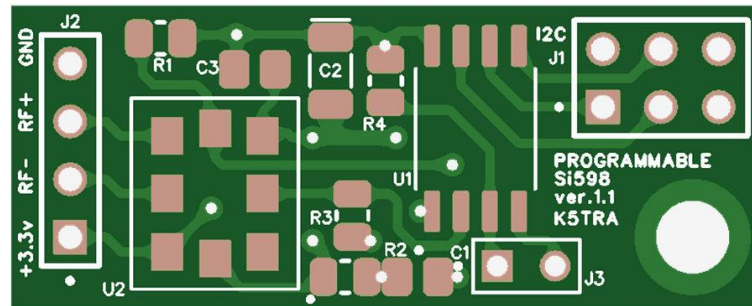
Program | Verify | Read

Starting operation verify registers  
Verify register EXTENDED...OK  
Verify register HIGH...OK  
Verify register LOW...OK  
Verify registers ... OK

▾ Verify registers ... OK

***THESE ARE DESIRED (DEFAULT) SETTINGS***

# QUESTO E' TUTTO



- Atmel/Microchip Studio programming: <https://github.com/ke0ff/PgmXtal>
- Arduino IDE programming: [ijmaca@gmail.com](mailto:ijmaca@gmail.com)
- Layout GERBER files: [tom@k5tra.net](mailto:tom@k5tra.net)