GoodWatch User's Guide

Travis Goodspeed and Friends

October 29, 2018

Howdy y'all,

The GoodWatch is a clone of the Casio 3208 watch module, replacing that module in the CA-53W and CA-506 series watches. It is built around the CC430F6137 RFSoC, allowing for years of battery life with convenient C programming. The second model adds a 70cm amateur radio transceiver.

1 Usage

Applets are accessed in a ring, with the Mode button on the right side of the watch switching between them. Sometimes the Mode button will also have a secondary function, in which case you'll press it twice to move to the next mode.

Clock

The watch begins in the Clock applet, and resets to that applet after a few minute from most other applets.

Pressing the Program button switches to a programming mode, where the hour, minute and date can be changed; the seconds are held at zero until the minute is finished being set.

Hold 9 to show the day of the week or ÷ to show the date in YYYYMMDD format. Hold 7 for a self-test, which ought to display `ALL GOOD` but might otherwise indicate a trouble code. Hold 4 to show the Git hash of the firmware revision, 5 to show the date of manufacture, or 6 to toggle a blinking CPU load indicator.

Stopwatch

The Stopwatch applet allows you to time things. Press + to start or stop the timer, and `0` to reset it.

RPN Calculator

The calculator operates in Reverse Polish Notation mode. The current implementation operates on 32-bit integers, but floating point support will come in time. Pressing Mode the first time will reset the stack to zero; a second press moves to the next applet.

There's little room here for a complete tutorial for RPN, but gist of it is that you push numbers onto the stack by typing them in, and each operator will pop the most recent values from the stack, pushing the result. Press `=` to push the current number into the stack and begin another; you can think of this as the Enter or Space key from an HP calculator.

Hex Viewer, Disassembler

The hex viewer application display eight nybbles on the screen. The left four are the address, while the right four are the little-endian value at that address. The 7, 8, 9, and ÷ buttons increment nybbles of the address, while the 0, ., =, and + buttons decrease the address nybbles.

You will find that unmapped regions of memory return `0x3FFF`, an unconditional branch to self. The bootloader at `0x1000` is unlocked for your reading pleasure, but a troublesome region at `0x1B00` is blacklisted because reads from that region trigger a reboot. Flash runs 32kB starting at `0x8000`, Info Flash is at `0x1800`, and RAM is at `0x1C00`. The interrupt table is at the very end of the address space.

Holding the 4 button will disassemble the current instruction, as best the 7-segment display can handle. AM and PM show the two bits of the `As` field, and ÷ shows the one bit of the `Ad` field.

2 Construction

3 Installation

4 Source Code