**ApRemote – a web based Raymarine SeaTalk 1 autohelm remote control**

Source: <https://github.com/richardJG/APRemote>, Last commit, Nov 2020

Platform IO

**Libraries for build:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name & author** | **PlatformIO libdeps entry:** | **Version** | **Maintainer** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Moving to alternative web sockets library & code in ApRemote in PlatformIO

<https://m1cr0lab-esp32.github.io/remote-control-with-websocket/led-setup/>

Now complies :-)

String send from ApRemote to web client with data is:

{"hdg":0,"cts":0,"hdgInfo":"&nbsp;","rsa":"S0","sog":0.00,"cog":0,"awa":"0S","xte":"---","aws":0.00,"vlw":0.00,"dpt":0.00,"stw":0.00,"dtw":"---","btw":"---","left":"&nbsp;","right":"&nbsp;","led0":"led\_on","led1":"led\_off","led2":"led\_off","led3":"led\_off","alm":" "}

Adding Wi-Fi Access point capability

Rui Santo’s Tutorials: <https://RandomNerdTutorials.com/>

[ESP32 Access Point (AP) for Web Server | Random Nerd Tutorials](https://randomnerdtutorials.com/esp32-access-point-ap-web-server/)

**SeaTalk 1 interface circuit.**

### Hardware-Interface

### SeaTalk uses three wires, connected in parallel to all devices on the bus:

1. +12V    Supply, red
2. GND    Supply, grey
3. Data    Serial Data, yellow:
   1. +12V=Idle/Mark=1, 0V=Space/Data=0,
   2. 4800 Baud,
   3. pullup circuit in each device,
   4. talker pulls down to 0V (wired OR).

Original Raymarine interface (for 5V microcontroller):

Diagram, schematic

Description automatically generated

Reworked for 3.3V microcontroller:

Using EasyEDA simulation mode.

Diagram, schematic

Description automatically generated5V (VCC) related resistors have been scaled by 3.3/5 and nearest standard value selected.

Simulation showed circuit works with 11, 12 and 14V charge states of 12V lead acid battery.

A picture containing graphical user interface

Description automatically generated

**Info**

1. Great description of SeaTalk 1 electronics and messages …

<http://www.thomasknauf.de/seatalk.htm>

1. SeaTalk Autopilot Remote Control device based on an Arduino Pro Micro and a simple 433 MHz Key Fob.

<https://github.com/AK-Homberger/Seatalk-Autopilot-Remote-Control/blob/master/README.md>

1. Another ST Remote project:  
   This one is Arduino hosted, has “from first principles / no libraries” timer ISR to bit bank ST RX & TX.

<https://github.com/mattnj/SeaTalk-Tiller-Remote/tree/main>

1. A SeaTalk protocol library:

<https://github.com/jamesroscoe/seatalk>

**Autopilot SeaTalk sentences…**

On ST2000, you enter track mode by pressing Auto then +10 & -10 together. At each waypoint, acknowledge the turn by pressing +10 & -10 together. I’m thinking +10 & -10 together are equivalent to pressing track on an ST4000+ head unit … assuming the track button on the APR will enter track mode, and acknowledge turns at waypoints … need to test this at sea…

**Autopilot SeaTalk sentences in RichardJG’s AP remote software (in seatalk.cpp – sendCMD()**

                0x86, 0x21, 0x02, 0xfd,            // standby

                0x86, 0x21, 0x05, 0xfa,            // -1

                0x86, 0x21, 0x01, 0xfe,            // auto

                0x86, 0x21, 0x07, 0xf8,            // +1

                0x86, 0x21, 0x06, 0xf9,            // -10

                0x86, 0x21, 0x23, 0xdc,            // wind

                0x86, 0x21, 0x08, 0xf7,            // +10

                0x86, 0x21, 0x21, 0xde,            // tack (-1 + -10)

                0x86, 0x21, 0x03, 0xfc,            // track

                0x86, 0x21, 0x22, 0xdd};           // tack (+1 + +10

**Autopilot SeaTalk info (from Thomas’ site)**

**Items in bold are used in the original ApRemote code – i.e., above)**

**Bold blue** – try adding

86  X1  YY  yy  Keystroke  
                 Sent by autopilot (X=0: ST 1000+,  **X=2: ST4000+** or ST600R)  
     **X1  01  FE    Auto**     **X1  02  FD    Standby**  
     **X1  03  FC    Track**     X1  04  FB    disp (in display mode or page in auto chapter = advance)  
     **X1  05  FA     -1 (in auto mode)  
     X1  06  F9    -10 (in auto mode)**  
     **X1  07  F8     +1 (in auto mode)**  
     **X1  08  F7    +10 (in auto mode)**  
     X1  09  F6     -1 (in resp or rudder gain mode)  
     X1  0A  F5     +1 (in resp or rudder gain mode)  
     **X1  21  DE     -1 & -10 (port tack, doesn´t work on ST600R?)**  
     **X1  22  DD     +1 & +10 (stb tack)**     **X1  23  DC    Standby & Auto (enter wind mode)**     **X1  28  D7    +10 & -10 (in auto mode)**  
     X1  2E  D1     +1 & -1 (Response Display)  
     X1  41  BE    Auto pressed longer  
     X1  42  BD    Standby pressed longer  
     X1  43  BC    Track pressed longer  
     X1  44  BB    Disp pressed longer  
     X1  45  BA     -1 pressed longer (in auto mode)  
     X1  46  B9    -10 pressed longer (in auto mode)  
     X1  47  B8     +1 pressed longer (in auto mode)  
     X1  48  B7    +10 pressed longer (in auto mode)  
     X1  63  9C    Standby & Auto pressed longer (previous wind angle)  
     X1  68  97    +10 & -10 pressed longer (in auto mode)  
     X1  6E  91     +1 & -1 pressed longer (Rudder Gain Display)  
     X1  80  7F     -1 pressed (repeated 1x per second)  
     X1  81  7E     +1 pressed (repeated 1x per second)  
     X1  82  7D    -10 pressed (repeated 1x per second)  
     X1  83  7C    +10 pressed (repeated 1x per second)  
     X1  84  7B     +1, -1, +10 or -10 released

Data on boat with ST2000+ connected and in standby mode:

Rudder angle = 0.0

Heading=351

AP CTS = 0

Mode = 0

10 1 0 5D

11 1 C 7

20 1 0 0

23 1 13 42

50 2 33 4C 5

51 2 2 68 17

52 1 0 0

53 0 0

A2 84 0 34 34 3F 3F

26 4 0 0 0 0 40

27 1 22 1

9C 71 28 0

10 1 0 34

11 1 C 3

84 76 28 0 4 0 0 0 4

84 76 28 0 4 0 0 0 4

**Tests on boat**

* Standby, auto, wind, track buttons work.
* In Auto mode, +10/-10/+1/-1 buttons work.
* In Auto mode Tack buttons work.
* TFT compass display not working – error in GUI Slice … set external buffer to 6, now its updateable at run time.
* TFT LED’s not working – added GUISlice helpers to set radio button status. Removed call backs. Added code to update radios based on apMode.
* When in standby, not able to extend or retract tiller pilot piston remotely (+10 / -10) usually do this. Maybe try:   
       X1  82  7D    -10 pressed (repeated 1x per second)  
       X1  83  7C    +10 pressed (repeated 1x per second)  
  when in standby mode & test extend / retract
* Data screen shows 22 for heading – hdg was missing from TFT update in the main loop
* In Auto mode, the TFT APR main display shows an actual compass heading – The ST2000 does that in standby, but in Auto displays a course to steer (the +/- buttons can be seen changing that. The Web UI shows the right course to steer, with left and right arrows (assume on / off set heading?)
* Track causes ST2000 to enter T mode and ask for permission to turn, pressing track again doesn’t acknowledge this – may need to simulate +10 and -10 together. Try adding:  
  **X1  28  D7    +10 & -10 (in auto mode)**when in track mode (which might be a subset of auto mode?)
* When SeaTalk cable isn’t connected, and the ApRemote tries to send a command, it locks up – not great. I’m guessing this is because the electronics presents “bus busy” so the CheckBus() function blocks? – Need to separate SeaTalk send and SeaTalk receive functions into independent tasks. Need to handle SeaTalk send queue full (double beep & no action?)
* Suggest removing radio buttons and going with the ST2000 display – prefix heading info with C, A, W, T mode indicator.
* Data screen SOG = Chart plotter SOG … whole no’s not decimals. ST60 Multi shows tenths … but this might be a rolling average of the integer SOG??
* Electronics & S/W need a “beep” feedback on TFT button press (or web button press). Need a task that does a managed 0.5sec beep, or a 0.5 sec double (UI error) beep.
* Listen to SeaTalk “lamps” off / 1 / 2 / 3 and set TFT (& LED?) brightness accordingly.
* Add ability to control SeaTalk lamps from the APR TFT & maybe Web UI?

**Reliability issues**

* When SeaTalk data is coming in, TFT displays O.K.
* When SeaTalk data is coming in, and Wi-Fi is connected, system gets flaky … some random SeaTalk collisions reported (implying missed bytes by software serial receive. Once a tcp / network timing error caused a reset.

It seems system tasks (e.g. Wi-Fi / networking stack are pinned to core 0, and application tasks (the setup & loop) are pinned to core 1.

Need to:

* 1. Retest Wi-Fi reliability with no SeaTalk data coming in.
  2. Move SeaTalk task to the other core (clashes with the Wi-Fi code running on core 0?)

**Alternatives**

1. Ray control app – connects to wifi enabled mfd instruments?
2. **Raymarine** **S100** – autohelm remote approx. £400 – a basic unit

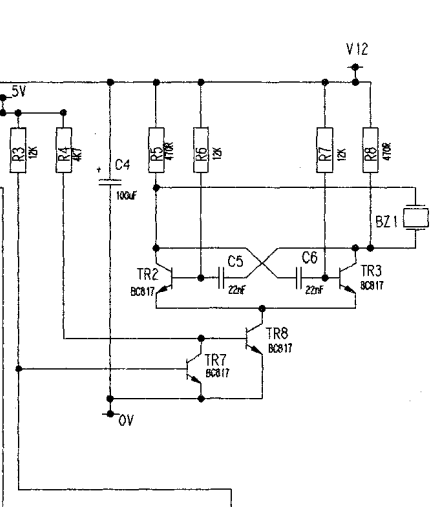
<https://www.force4.co.uk/item/Raymarine/S100-Autopilot-Wireless-Remote-with-Base-Station/AHD>

1. **Raymarine SmartController** with base station £540 – a more capable remote with data displays

<https://www.marinesuperstore.com/autopilots/autopilot-accessories/raymarine-smart-controller>

**Buzzer.**

Oscillator circuit for passive buzzer – snipped from PDF P115 of Raymarine autohelm 6000 service manual "C:\Users\dbarr\OneDrive\Poppy\Electrical\Raymarine Service\ST6000 Inboard Service Manual.pdf"



Circuit from ST60 service manual PDF page 77 "C:\Users\dbarr\OneDrive\Poppy\Electrical\Raymarine Service\st60 Service manual.pdf"

A diagram of a circuit

Description automatically generated