**Jednostka sterująca piHPSDR do transceivera Hermes Lite2**



Kontroler piHPSDR jest kontrolerem zdalnym do transceivera Hermes Lite 2 i innych urządzeń.. Normalnie używam go po połączeniu z komputerem przy pomocy programu SDR Console. Ten duży komputer ma wystarczającą moc, aby obsługiwać oprogramowanie w trybie cyfrowym, rejestratory i wykorzystywać wszystkie niesamowite możliwości naszych radiotelefonów. Ale platforma piHPSDR ma wystarczającą moc, aby łatwo wyobrazić sobie siedzenie w moim ulubionym fotelu, mały kontroler oparty o stolik, nogi w górze, napoje i przekąski pod ręką, może mecz futbolu na dużym ekranie i możliwość swobodnego zajmowania się DX-em lub przeżuwaniem szmat. Tak czy inaczej, taka jest moja wizja i trzymam się jej, jak widać. Koszty materiałów oscylują w okolicach 500 złotych z wbudowanym dźwiękiem, co jest dalekie od 1200 dolarów za Maestro.

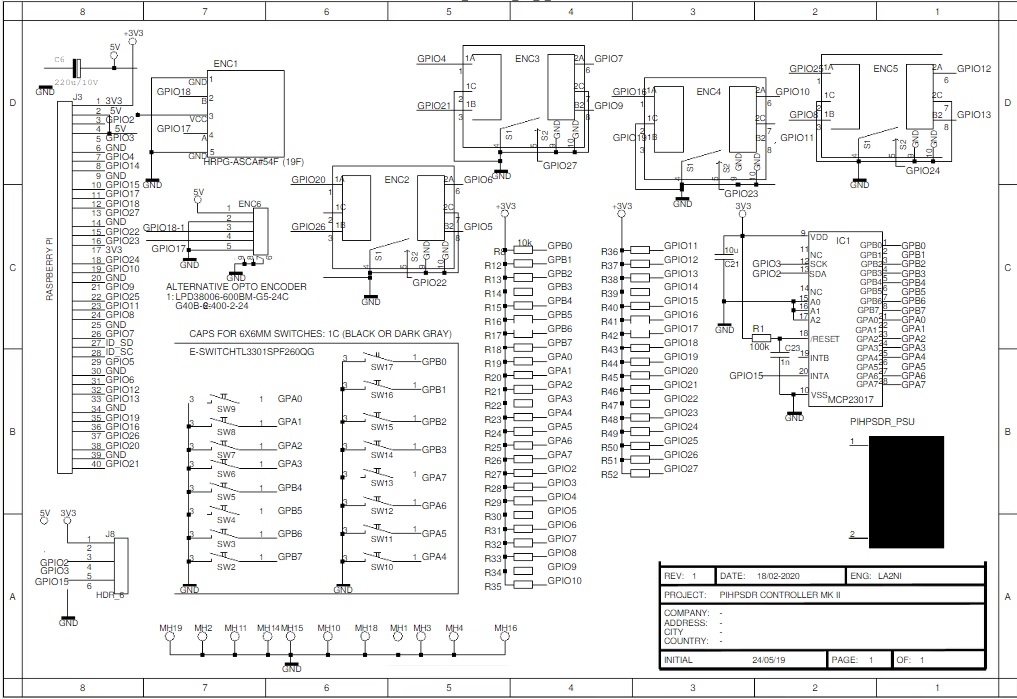
Opracowując mój projekt, chciałem czegoś czystego, kompaktowego, prostego w budowie, niedrogiego i niewymagającego projektowania płytek drukowanych. Oczywiście projekt musiał zawierać cztery enkodery obrotowe, ponieważ jest to *charakterystyczny* element konstrukcyjny. Jeśli chodzi o wybór enkoderów, jest ich naprawdę wiele. Chciałem, aby były montowane na panelu, a nie na płytce drukowanej. Koszt był zdecydowanie brany pod uwagę, w przeciwnym razie wybrałbym wyłącznie enkodery optyczne, ponieważ nic nie wydaje się tak dobrze wykonane jak dobrze wykonany enkoder optyczny. Jednak tylko pokrętło strojenia ma enkoder optyczny, pozostałe enkodery to bardzo tanie urządzenia. Przyciski też zostały zamontowane na płytkach uniwersalnych aby obniżyć koszty.

Jeśli chodzi o wymagania dotyczące zasilania, Raspberry Pi, wyświetlacz, interfejsy audio USB i inne elementy pobierają łącznie zaledwie 1,5 A przy pełnej mocy podczas pracy piHPSDR. Zasilacze Raspberry Pi o natężeniu 2,5 A z dobrym kablem wydają się być w porządku. Jeśli widzisz żółtą błyskawicę w prawym górnym rogu wyświetlacza, oznacza to, że Raspberry Pi wykryło stan zbyt niskiego napięcia. Do przechowywania danych karta SD o pojemności 32 GB była oczywistym wyborem, ponieważ pamięć masowa jest tania, więc nie ma sensu jej zaniżać.

Lista materiałów:

* Wyświetlacz 7” HDMI z panelem dotykowym
* Raspberry Pi 4
* 3 enkodery z przyciskiem
* 1 enkoder optyczny
* 16 przycisków podłączonych do MCP23017

Cała konstrukcja jest stworzona przy uzyciu ogólno dostępnych materiałów. Front panel wycięty z Dibond’u, boki obudowy, nówki, gałki, przyciski i ramki drukowane drukarką 3d z materiału PET-G. Zaprojektowane w programie Fusion360.

Schemat elektryczny:

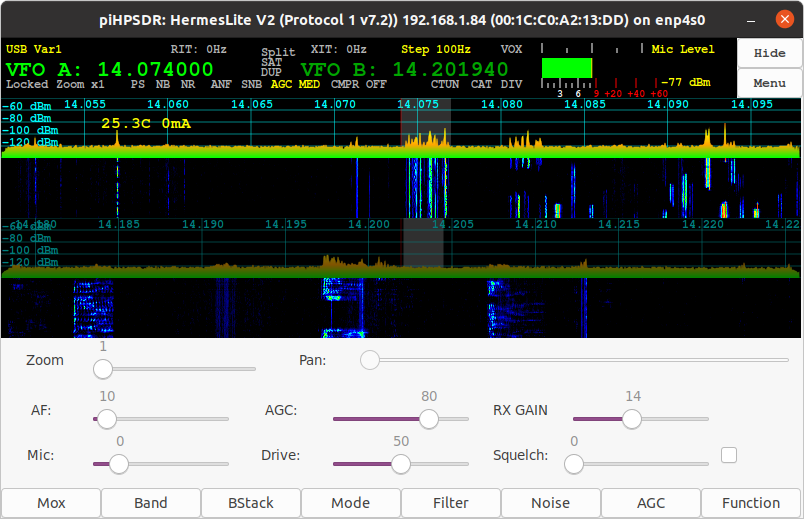
Okablowanie na podstawie powyższego schematu jest pokazane w poniższej tabeli.

|  |  |  |  |
| --- | --- | --- | --- |
| **Z** | | **DO** | |
| **KODER** | **SZPILKA** | **PIN ZŁĄCZA GPIO #** | **SYGNAŁ GPIO** |
| VFO | +5V | 4 | +5V |
| VFO | A | 11 | GPIO 17 |
| VFO | B | 12 | GPIO 18 |
| VFO | GND | Uwaga 1 | GND |
| Z | A | 37 | GPIO 26 |
| Z | B | 38 | GPIO 20 |
| Z | C | Uwaga 1 | GND |
| Z | 1 | Uwaga 1 | GND |
| Z | 2 | 22 | GPIO 25 |
| AGC | A | 40 | GPIO 21 |
| AGC | B | 7 | GPIO 4 |
| AGC | C | Uwaga 1 | GND |
| AGC | 1 | Uwaga 1 | GND |
| AGC | 2 | 26 | GPIO 7 |
| RF | A | 35 | GPIO 19 |
| RF | B | 36 | GPIO 16 |
| RF | C | Uwaga 1 | GND |
| RF | 1 | Uwaga 1 | GND |
| RF | 2 | 24 | GPIO 8 |

Uwaga 1: przewody uziemiające były połączone szeregowo od jednego pinu enkodera do drugiego i podłączone do pinu nr 39 (GND) złącza GPIO. Pozostałe piny uziemiające GPIO to 6, 9, 14, 20, 25, 30 i 34.

Na Raspberry Pi został zainstalowany system operacyjny Ubuntu i i aplikacja piHPSDR . Po uruchomieniu aplikacji wpisujemy adres IP transceivera i po uzyskaniu połączenia możemy pracować. Jednorazowo przed połączeniem konfigurujemy klawisze i encodery. Obsługa jest możliwa poprzez ekran dotykowy, myszkę i dostępne klawisze i enkodery na obudowie. Sygnał audio może być przekazywany z transceivera do konsoli. Oprogramowanie w pełni obsługiwać transceiver Hermes Lite. Zawiera predefiniowane ustawienia filtrów wejściowych.

Ekran aplikacji:



Źródło:

<https://github.com/g0orx/pihpsdr>

<https://g0orx.blogspot.com/>