Rebel Alliance Mod v1.0 Nov-10, 2013

By: PA3ANG, NA8E,K4JK and AE6RQ (beta tester)

INTRODUCTION

The Rebel Alliance Mod is a MPIDE /UNO32 sketch for the OPEN SOURCE Ten-Tec Rebel Model 506. The sketch is based on the Rebel Base sketch as delivered with the Rebel when buying the Transceiver from Ten-Tec.

The Rebel Base sketch is basically the standard firmware of the Rebel transceiver and controls all basic functions. The Rebel Alliance Mod sketch represents firmware with added functionality. This functionality enriches the Rebel to create a complete CW Transceiver for the 20 and 40 meter HAM bands.

Note: The sketch is not Ten-Tec approved nor have the programmers any relation with Ten-Tec other than being their customer.

The added functionality is:

- Iambic A/B Keyer
- Keyer speed control using on-board potentiometer
- Keyer speed control using paddle (menu U1)
- Automatic detection of Straight key / Paddle
- CQ and BEACON generator
- Switching between 20 and 40 meters (**requires additional hardware**)
- Band memory when switching between bands
- Display support for:
    - 20 x 4 LCD – 4BIT interface
    - 20 x 4 LCD – I2C interface
    - 16 x 2 LCD – 4BIT interface
    - 16 x 2 LCD – I2C interface
    - Nokia 5110
    - USB Terminal (PuTTY, Screen, Hyperterm)
- Frequency Announce in morse code
- CW Decoder (**requires additional hardware**) Not fully implemented

Note: Use of the sketch is at own risk. The programmers can be held responsible. The sketch is open source and may be freely distributed under GNU General Public License rules.

MPIDE Software

The sketch is created and can be loaded into the Rebel by using **the MPIDE software**. This Multi Platform IDE software is Arduino compatible and capable to compile UNO32 sketches . The software can be downloaded from : http://chipkit.net/started/.

The software will install on your PC and recognize the USB port of the connected Rebel.

LIBRARIES

The Rebel Alliance Mod sketch uses several library. These can be downloaded from : https://github.com/pa3ang/Tentec506/tree/master/lib.

You have to place the libraries into the mpide folder in /hardware/pic32/libraries

The extra needed libraries are:

- LiquidCrystal – if you are using a LCD display
- LiquidCrystal_I2C1602V1 – if the LCD is interface with the rebel using I2C
- Adafruit_PCD8544_Nokia_5110_LCD_library – if you are using the Nokia 5110 display
- Adafruit_GFX – if you are using the Nokia 5110 display
- Morse_EnDecoder – for the CW decoder feature

The library Wire.h is part of the MPIDE package.

Rebel Alliance Mod place holder.

The .pde file can be found at Github or in the Ten-Tec Rebel Model 506 Yahoo Group.

- https://github.com/pa3ang/Tentec506
- https://github.com/pstyle/Tentec506
- http://groups.yahoo.com/neo/groups/TenTec506Rebel/files/User%20Upload%20folder/Rebel%20Alliance%20Mod

FEATURE SELECTION

The sketch has been build around so called FEATURE selects.

```
#define FEATURE_DISPLAY              // LCD display support (include one of the Model AND INTERFACE options below)
#define FEATURE_LCD2004               // Classic LCD display using either 4 I/O lines or I2C.
//#define FEATURE_LCD1602             // Classic LCD display using either 4 I/O lines or I2C.
#define FEATURE_LCD_4BIT             // Select the LCD Display interface either I2C or 4BIT NOT BOTH!
//#define FEATURE_LCD_I2C             // I2C backpack interface.
```

If you have a LCD display then uncomment the line #define FEATURE_DISPLAY and select either LCD2004 (for a 20x4 LCD) or LCD1602 (for a 16x2 display) and then select the type of digital interface (4 bit or I2C). Note that these LCDs can display only upper case characters. Attempts to print lower case or comma will result in unwanted behavior.

Or you can select the Nokia by uncommenting #define FEATURE_DISPLAY and FEATURE_LCD_NOKIA5110

```
#define FEATURE_DISPLAY              // LCD display support (include one of the Model AND INTERFACE options below)
#define FEATURE_LCD_NOKIA5110        // If using a NOKIA5110 Display.
```

Alternativily you can use the PC to display the Rebel settings by using a simple terminal program like PuTTY, Hyperterm or Linux Screen.

```
#define FEATURE_DISPLAY              // LCD display support (include one of the Model AND INTERFACE options below)
#define FEATURE_TERMINAL             // Use a dumb terminal program as display ………
```

The CW Decoder software is implemented and does work. However you need additional hardware such as a NE567 tone decoder and preferably a Limiting amplifier and filter to ensure solid audio / tone detection.

```
#define FEATURE_CW_DECODER            // With additional NE567 or similar tone decoder
```

The Keyer can be selected by uncommenting the FEATURE_KEYER and FEATURE_SPEEDCONTROL_A7 or FEATURE_SPEEDCONTROL_U1

```
#define FEATURE_KEYER                  // Keyer based on code from OpenQRP.org.
#define FEATURE_SPEEDCONTROL_A7   //Analog speed control (uses onboard trimpot connected to A7)
#define FEATURE_SPEEDCONTROL_U1   //Control the speed with the paddle by selecting U1 menu.
```

Other Features can be selected or unselected by either uncomment or comment the line

```
#define FEATURE_BEACON _CQ            // Use USER Menu U3 and U2 to run message.
//#define FEATURE_SERIAL              // Enables serial output.  Only used for debugging at this point.
//#define FEATURE_CAT_CONTROL        // Enables CAT based on Kenwood All Frequency set and read

#define FEATURE_BANDSWITCH            // Software based Band Switching.  Press FUNCTION > 2 seconds
#define FEATURE_FREQANNOUNCE         // Announce Frequency by keying side tone (not TX). Press SELECT > 2 seconds
```

Note: Unlike other features which can be configured in, even if you do not intend to use them, the FEATURE_BANDSWITCH should only be selected when the additional bandswitching hardware is installed.

IAMBIC KEYER

The Rebel Alliance Mod firmware auto-detects a straight key when the 3,5mm jack center conductor is grounded due to insertion of a straight key plug. In that case the FEATURE_KEYER is switched off.

The setup is simple. In the sketch it is possible to swap the Dah and Dit and to select between Iambic A and B version. The speed is either controlled by the trimpot A7 or fixed .

```
//-------###########  SET CW SPEED HERE (If you dont use the analog control)  #######----------
int ManualCWSpeed = 15; //  <---- SET MANUAL CW SPEED HERE

#ifdef FEATURE_KEYER
  //  keyerControl bit definitions
  #define     DIT_L      0x01     // Dit latch
  #define     DAH_L      0x02     // Dah latch
  #define     DIT_PROC   0x04     // Dit is being processed
  #define     PDLSWAP    0x08     // 0 for normal, 1 for swap
  #define     IAMBICB    0x10     // 0 for Iambic A, 1 for Iambic B

  //Keyer Variables
  unsigned char       keyerControl;
  unsigned char       keyerState;
  int ST_key = 0;        //This variable tells TX routine whether to enter use straight key mode
  enum KSTYPE {IDLE, CHK_DIT, CHK_DAH, KEYED_PREP, KEYED, INTER_ELEMENT };
#endif // FEATURE_KEYER
```

SPEED CONTROL

This code provides 3 options to adjust keyer speed control:

1. No control, but fixed value
   //-----------############ SET CW SPEED HERE (If you dont use the analog control) #######-------------
   int ManualCWSpeed = 15; // <---- SET MANUAL CW SPEED HERE

2. With trimpot A7 (on the Rebel board! Choose FEATURE_SPEEDCONTROL_A7

3. With the paddle when in menu U1
   Choose FEATURE_SPEEDCONTROL_U1 and comment SPEEDCONTROL_A7!
   When in menu U1 the speed can be set by pressing the Dit and Dah paddle. On the display the speed is updated immediately. When ready unselect menu U1 by pressing the FUNCTION key.

CQ AND BEACON KEYER.

```
#ifdef FEATURE_BEACON
  // Simple Arduino CW Beacon Keyer
  // Written by Mark VandeWettering K6HX
  #define     BEACON          ("VVV DE PA3ANG/BEACON  JO32AM")           // Beacon text
  #define     CQ              ("CQCQCQ DE PA3ANG PA3ANG PA3ANG PSE K")     // CQ text
  #define     CW_SPEED        20                                 // Beacon Speed    is fixed !!
  #define     BEACON_DELAY    10                                 // in seconds
```

The CQ text will be send when selecting U2. First there is a CQDELAY time of 2 seconds and then the CQ message will be transmitted. You can stop at any time during transmission by pressing the DIT or DAH lever. During CQDELAY the sequence can be stopped by unselecting U2.

The same applies for the Beacon which is selected and transmitted by selecting U3.

All parameters of these messages are fixed including content, CW send speed, and pause time between repetitions.

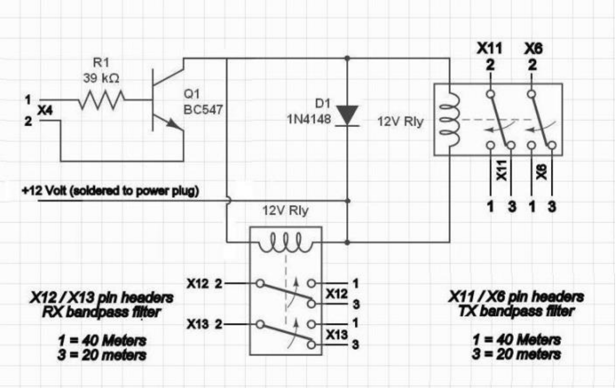
Note: the CQ and BEACON speed is fix and has no relation with the IAMBIC Keyer speed!

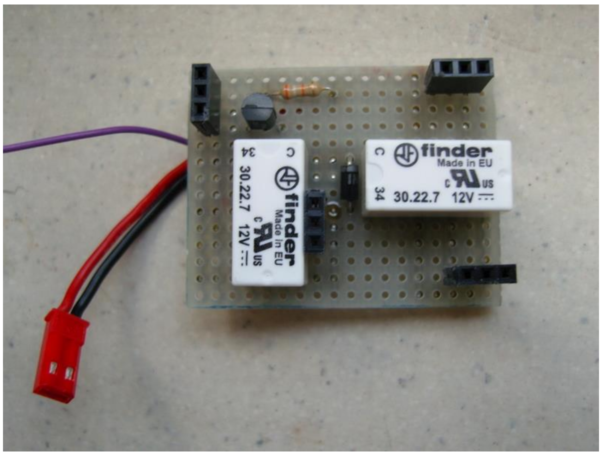
BANDSWITCH

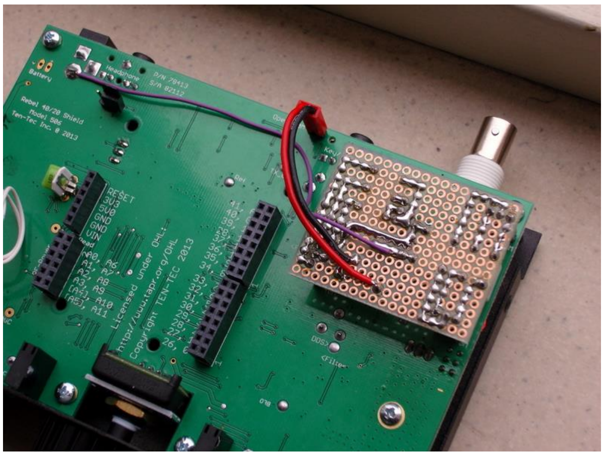
The standard Rebel shipped by Ten-Tec uses five, 2-pin headers to specify the operating band (40m or 20m). This standard Ten-Tec method is recognized by the Rebel Alliance Mod sketch.

Rebel Alliance Mod sketch also includes support for relay-switching between the 20 and 40 meter band by using the FUNCTION button. (press 2 seconds or longer). This bandswitching function requires additional hardware as shown by the schematics and photos below.

The Rebel Alliance Mod firmware will memorize the frequency when switch between bands.







Note that the #define #FEATURE_BANDSWITCH in the software should only be uncommented when this additional bandswitching hardware is installed, even if the software otherwise compiles without errors.

DISPLAY SUPPORT

The Rebel Alliance Mod firmware can support 3 type of physical display, 2 type of LCD display interfaces and a simulated Dsiplay on the PC using a dumb terminal program like PuTTY, Hyperterm, Realterm or Screen (Linux, RPi!).

The LCD displays are interfaced either with a so called 4 BIT parallel interface or with a I2C interface.

Pinning 4BIT:

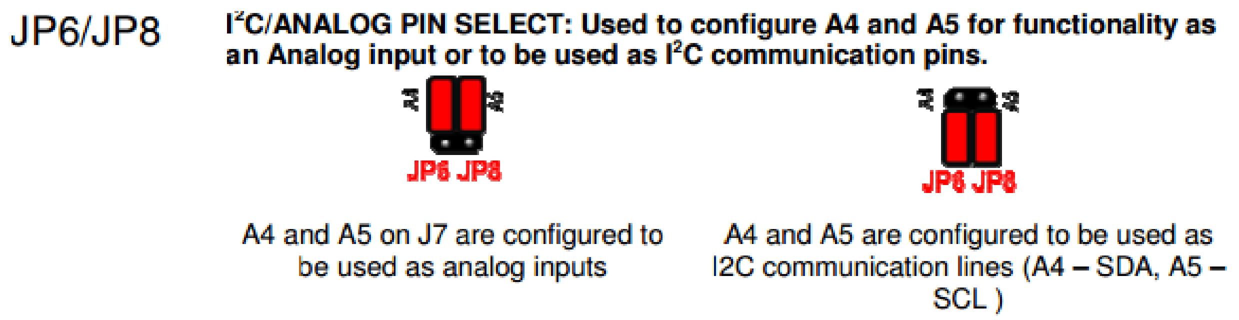
- LCD RS pin to digital pin 26
- LCD Enable pin to digital pin 27
- LCD D4 pin to digital pin 28
- LCD D5 pin to digital pin 29
- LCD D6 pin to digital pin 30
- LCD D7 pin to digital pin 31
- LCD R/W pin to ground
- 10K resistor:
- ends to +5V and ground
- wiper to LCD VO pin (pin 3)

Note: Consult your LCD schematic for connection (and voltage) for the backlight LED.

Pinning I2C:

- SDA to analog pin A4
- SCL to analog pin A5
- GND and +5 Volt

Note: UNO32 board JP6/JP8 – A4/A5 Signal Select Jumpers. These jumpers are used to switch pins 9 and 11 on connector J7 between analog inputs A4 and A5 or the I2C signals SDA and SCL. On my Ten-Tec shipped Rebel these JP6/JP8 had the I2C selected.



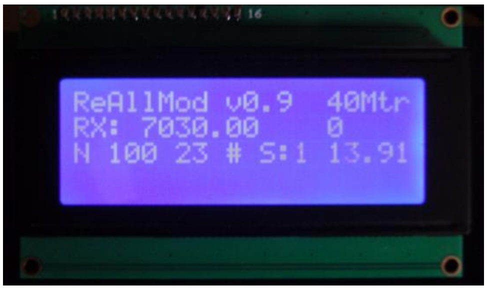
JP6/JP8

I^{2}C/ANALOG PIN SELECT: Used to configure A4 and A5 for functionality as an Analog input or to be used as I^{2}C communication pins.

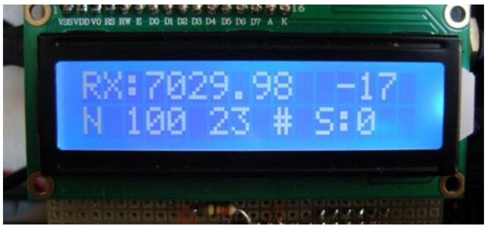
A4 and A5 on J7 are configured to be used as analog inputs

A4 and A5 are configured to be used as I2C communication lines (A4 – SDA, A5 – SCL)

Display types

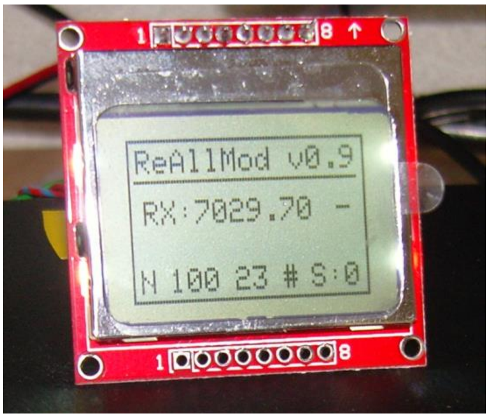


20 characters and 4 lines. The 4th line is used for the CW Decoder



16 characters and 2 lines. This is a minimal display option.

Note: there are Ebay types who can’t handle lower case characters. In that case the txt0 – RebelAllMod must be changed to txt0 – REBELALLMOD. (all upper case)



Graphical display with currently a 4 line layout with 13 characters

Pinning Nokia 5110:

- pin 30 - Serial clock out (SCLK)
- pin 29 - Serial data out (DIN)
- pin 28 - Data/Command select (D/C)
- pin 27 - LCD chip select (CS)
- pin 26 - LCD reset (RST)

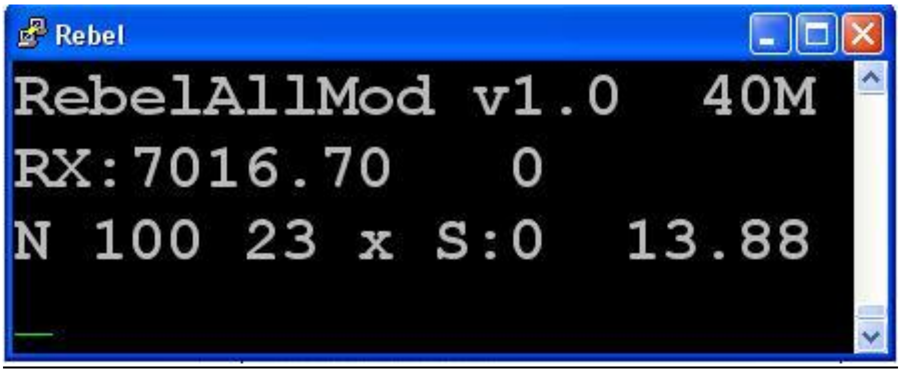
In anticipation of future features, Rebel Alliance Mod developers have included support for the Nokia 5110 display based on its fast display speed and its ability to support graphics.

TERMINAL

Selecting FEATURE_TERMINAL as display will enable passing the information via the USB connector to a connected PC. Either Windows or Linux and even Raspberry Pi.

The terminal program on the PC should be capable to understand ANSI (cursor position and clear screen commands) and could be virtually anything from simple to advanced.

Note: If using the FEATURE_TERMINAL it's not possible to use FEATURE_CAT_CONTROL.



```
RebelAllMod v1.0   40M
RX:7016.70    0
N 100 23 x S:0  13.88
```

PuTTY

FEATURE_SERIAL

This is the original Serial Dump fro the Rebel_Base sketch. It will give some basic information and timing characteristics.

CAT CONTROL

The Rebel Alliance Mod sketch has a very basic CAT Control function based on the Kenwood protocol. The sketch can receive the **FA** command (frequency set) and send the frequency back based on the **IP** command.
Both commands are used by Logger32, which happens to be my logbook program.

FREQUENCY ANNOUNCE

If you do not have a display attached to the Rebel, you still can have a accurate Frequency 'readout' by pressing the SELECT button more than 2 seconds. This will start a frequency announce in morse code of the frequency rounded to 100Hz.

CW DECODER

The Rebel alliance Mod sketch has a build in Morse Decoder. In fact the Morse EnDecoder library is capable to Encode and decode morse code.

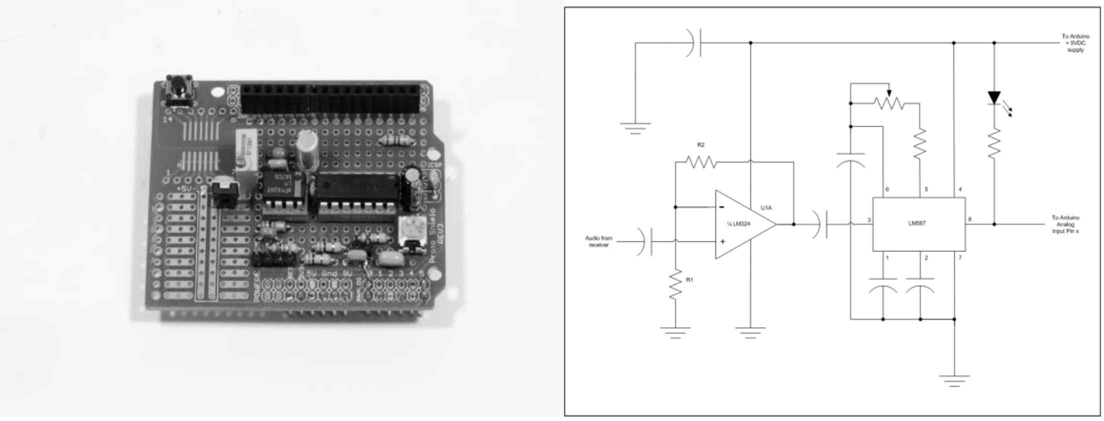
-

The current version supports display of the decoded characters on the 4th line of the 20 x 4 display!

In order to use the decoder feature an additional shield needs to be made. This shield should have the following functions:

- Amplifier to boost the signal from analog pin A6
- Limiter / AGC to have a stable signal
- 800Hz filter
- Tone decoder (NE567)

Several prototypes of such circuits have been made to date, but none have been able to achieve a stable decoding with between weak and strong signals. We suggest that you monitor the Rebel Yahoo! Group for information on future development of the CW decoder hardware.



Prototype of partly workable solution.