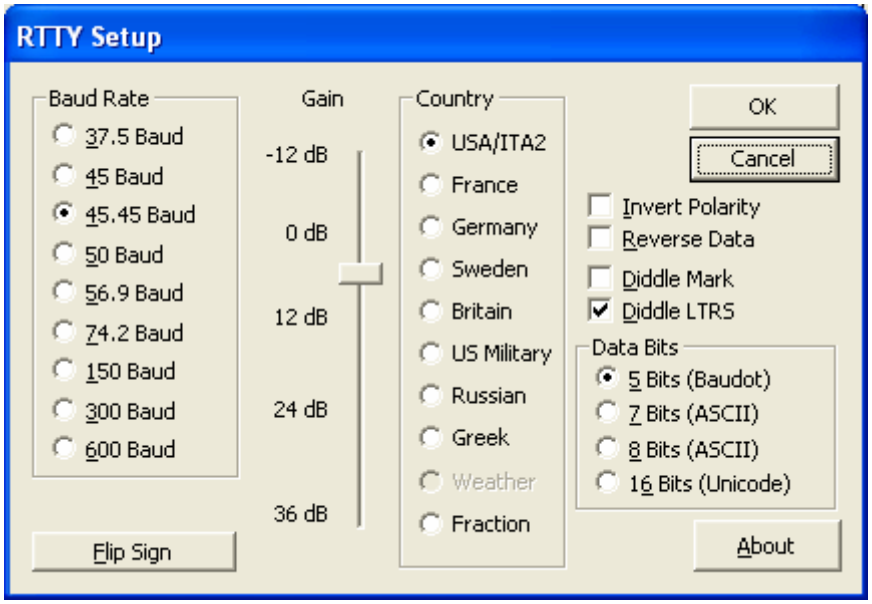
# HamHub™ RTTY Interface

Designed by
TIMEWAVE TECHNOLOGY INC.
1025 Selby Ave, Suite 101, St Paul, MN 55104-6533
Phone: (651) 489-5080 ● Email: sales@timewave.com

The Timewave HamHub™ RTTY Interface is for receiving and transmitting RTTY via the Soundcard on a PC. It follows all the standard international conventions.

## Setup

When the setup is selected, either via clicking on Setup in HHTRX or by choosing Soundcard Setup from the File menu in PK-Term, the following dialog appears:




### *Baud Rate*

The baud rate for receiving and transmission can be set to several speeds. The faster data is transmitted, the more error there will be on the receiving end. Generally, 300 baud is the highest that this should be set. The standard for ham radio in the United States is 45.45 baud, the speed the old Western Union teletypes used. A new emerging popular speed is 300 baud ("400 Speed").

*Standard Baud Rates*

| Baud | Speed (WPM) | Name |
|---|---|---|
| 45.45 | 65 | Western Union |
| | 61.33 | 60 Speed |
| | 60.61 | 45 bauds |
| 50.00 | 66.67 | European/50 bauds |
| 56.92 | 76.68 | 75 Speed |
| | 75.89 | 57 bauds |
| 74.20 | 100.00 | 100 Speed |
| | 98.98 | 74 bauds |
| 100.0 | 133.33 | 100 bauds |

Source "The ARRL Handbook for Radio Communications, 2004"

## Flip Sign

When this is chosen, then sign for the incoming Baudot will be flipped. This means that the reception will switch between the Figures and the Letters. See *What is Baudot?* on page 4 below for details.

This can also be done if this button is on the toolbar:

Note: The FIGS↔LTRS flip will be done immediately, not when the OK is selected.

## Gain

The incoming signal may be too strong or too weak to be received properly. Adjust the gain for your optimal reception. Ideally, the receiver gain should be adjusted; this is only for fine tuning.

## Country

When using Baudot data reception (see *What is Baudot?* on page 4 below), this will specify the conversion for displaying and transmitting the data.

## Invert Polarity

This will reverse the high and low data bits. Commonly used for government data transmission.

## Reverse Data

This will invert the data word, sending the MSB first. Commonly used for government data transmission.

## *Diddle Mark/Diddle LTRS*

When sending Baudot, there are two conventions for the diddle, Mark and LTRS. The diddle is what is sent when there is no data. A Mark tone will send a constant low frequency tone, whereas the LTRS will send a LTRS figure shift repeatedly. The Mark tone was more popular in the past, but in recent years the LTRS has become more popular as it synchronizes the data much better.

If neither of these are checked, then no diddle will be sent.

## *Data Bits*

This specifies the number of data bits used to encode the data. There are four possible settings: 5-bit Baudot, 7-bit ASCII, 8-bit ASCII, and 16-bit Unicode.

### Baudot (5-bit)

This is the teletype standard since WWII. See *What is Baudot?* on page 4 below.

### ASCII (7-bit)

This is a US only standard. It encodes the Alphabet and punctuation, both upper and lower case. It does NOT support non-English languages. This was used for a short time in the late 60s and early 70s and is nearly non-existent today.

### ASCII (8-bit)

The most common of the modern standards, this is the full ASCII code set.

### Unicode (16-bit)

This is used to send data in any Unicode supported language. This includes Hebrew, Japanese, Chinese, Korean, Thai, Greek, Russian, and several other alphabets. Because of the word size, this is also the slowest of all the standards.

Official Unicode is 32-bit, but there is a sub-standard that only uses the low word of the Unicode character and so is 16-bits. This is used by Microsoft® Windows™.

## *About*

Displays the About box.

## What is Baudot?

Sometimes called the Murray code (British), this was developed during WWII and became instantly popular following the war. This was widely used on Teletypes (called TTYs).

It is a 5-bit standard that allows 32 differentiating characters to be sent. To allow for 26 characters and 10 numbers plus carriage returns, the concept of a SHIFT was created. Following a LTRS (letters) SHIFT, all the characters would be interpreted as alphabetical. Following a FIGS (figures) SHIFT, all the characters would be interpreted as numeric or punctuation. Naturally, this doesn't allow for the extra characters that are found in non-English languages. Also, special interest parties wanted specialized character sets for themselves. Thus was born the code page.

The code page is essentially just a different translation table for each language or special interest. For instance, a character 26 means several different things depending on the code page. For the U.S., it is an '&', for British '@', Sweden 'Ä', German 'Ö', French '%', Russian 'Ш', etc.

Note: Late 20th century computers used 8-bit code pages for different languages. However, the only standards were ASCII and IBM. There are literally thousands of manufacturer defined code pages (especially on UNIX systems). It would be impossible to support each one. ASCII is what HH RTTY supports. The newest computers, both UNIX and Windows based, use Unicode so code pages are not necessary.

| Code | ITA2 LTRS | ITA2 FIGS | France FIGS | Germany FIGS | Sweden FIGS | Britain FIGS | Russia LTRS | Russia FIGS | Greece LTRS | Greece FIGS | US Mil FIGS | Weather FIGS | Fraction FIGS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| 4 | | | | | | | | | | | | - | |
| 23 | Q | 1 | 1 | 1 | 1 | 1 | Я | 1 | Ψ | 1 | 1 | 1 | 1 |
| 19 | W | 2 | 2 | 2 | 2 | 2 | В | 2 | Ω | 2 | 2 | 2 | 2 |
| 1 | E | 3 | 3 | 3 | 3 | 3 | Е | 3 | Е | 3 | 3 | 3 | 3 |
| 10 | R | 4 | 4 | 4 | 4 | 4 | Р | 4 | Р | 4 | 4 | 4 | 4 |
| 16 | T | 5 | 5 | 5 | 5 | 5 | Т | 5 | Т | 5 | 5 | 5 | 5 |
| 21 | Y | 6 | 6 | 6 | 6 | 6 | Ы | 6 | Υ | 6 | 6 | 6 | 6 |
| 7 | U | 7 | 7 | 7 | 7 | 7 | У | 7 | | 7 | 7 | 7 | 7 |
| 6 | I | 8 | 8 | 8 | 8 | 8 | И | 8 | I | 8 | 8 | 8 | 8 |
| 24 | O | 9 | 9 | 9 | 9 | 9 | О | 9 | О | 9 | 9 | 9 | 9 |
| 22 | P | 0 | 0 | 0 | 0 | 0 | П | 0 | П | 0 | 0 | 0 | 0 |
| 8 | cr | cr | cr | cr | cr | cr | cr | cr | cr | cr | cr | cr | cr |
| 3 | A | - | - | - | - | | А | - | А | - | - | ↑ | - |
| 5 | S | ' | ' | ' | ' | [1] | С | ' | Σ | ' | BEL | BEL | BEL |
| 9 | D | WRU | WRU | WRU | WRU | [2] | Д | Ч | Δ | WAU | $ | ↗ | $ |
| 13 | F | ! | É | Ä | Å | [1]/ | Ф | Э | Φ | | ! | → | ¼ |
| 26 | G | & | % | Ö | Ä | [3]/ | Г | Ш | Γ | | & | ↘ | & |
| 20 | H | # | | Ü | Ö | [5]/ | Х | Щ | Н | | # | ↓ | # |
| 11 | J | BEL | BEL | BEL | BEL | [7]/ | Й | Ю | Χ | BEL | ' | ↙ | ' |
| 15 | K | ( | ( | ( | ( | [9]/ | К | ( | Κ | ( | ( | ← | ½ |
| 18 | L | ) | ) | ) | ) | / | Л | ) | Λ | ) | ) | ↖ | ¾ |
| 2 | lf | lf | lf | lf | lf | lf | lf | lf | lf | Lf | lf | lf | lf |
| 17 | Z | + | + | + | + | . | З | + | Ζ | + | " | + | " |
| 29 | X | / | / | / | / | £ | Ь | / | Ξ | / | / | / | / |
| 14 | C | : | : | : | : | ( | Ц | : | Θ | : | : | ◯ | ⅛ |
| 30 | V | = | = | = | = | ) | Ж | = | | = | ; | ◯ | ⅜ |
| 25 | B | ? | ? | ? | ? | ? | Б | ? | Β | ? | ? | ⊕ | ⅝ |
| 12 | N | , | , | , | , | - | Н | , | Ν | , | , | ◯ | ⅞ |
| 28 | M | . | . | . | . | ' | М | . | Μ | . | . | . | . |
| 31 | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS | LTRS |
| 27 | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS | FIGS |

There are several other Baudot conversions defined, but they are obsolete and not used anymore.

Please see The ARRL Handbook for Radio Communications, 2004, chapter 12 "Text Modes" for further information.