

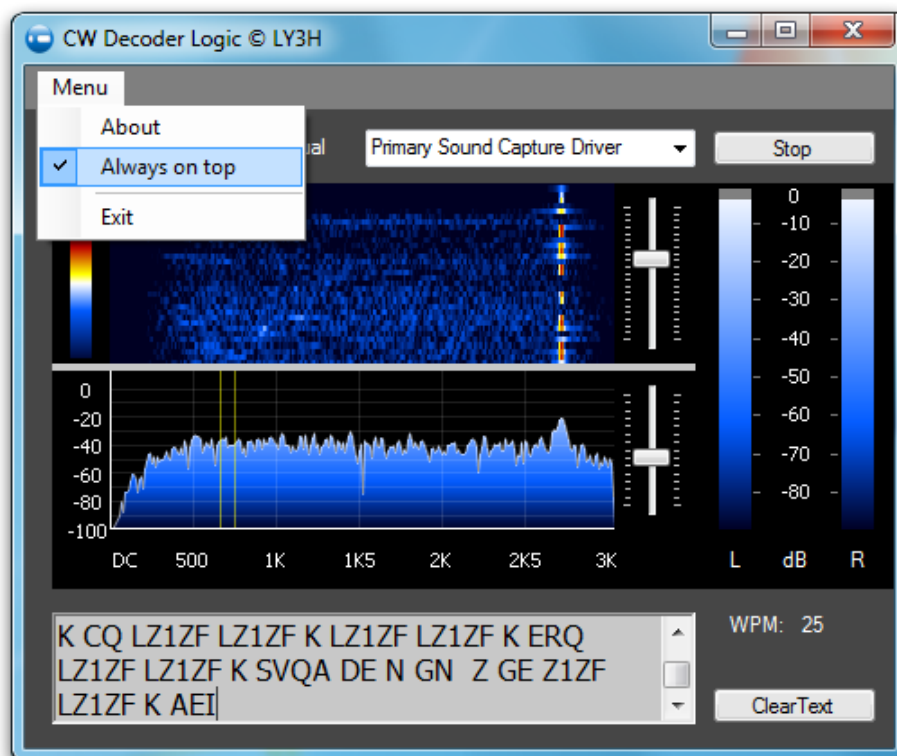
Cw Decoder Logic v1.0.0.2

Changelog:

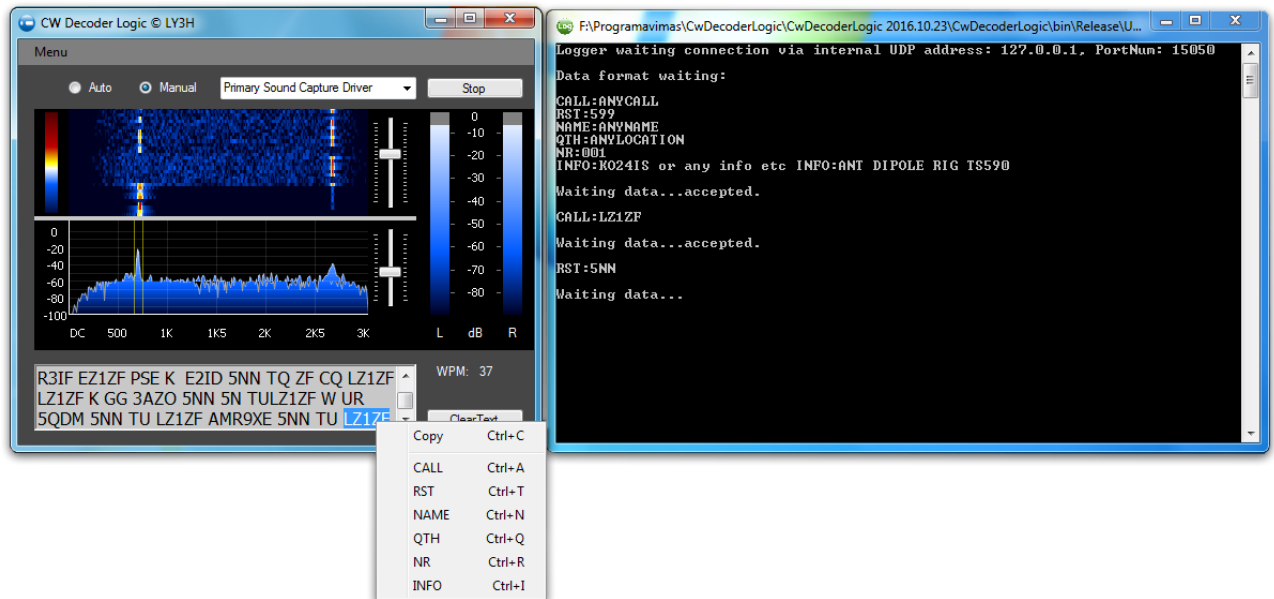
1. Fixed bug with colors and components positions on different Windows versions.
Also changing Windows themes now all should be set OK.
2. Fixed flicking on text.
3. Fixed program stability.
4. Added "Always on top" function.
5. Added resize function.
6. Added text selection and copy function.
7. Added possibility to send selected text to external Logger via UDP protocol.
8. Added logger source code example and program for logger developers.
Demo program Start / Programs / LY3H / UDPLogger.exe
Please try it.
9. Replacement of the installation path. Now in Start / Programs / ~~HAM~~ LY3H /

More about everything

"Always on top" function is in Menu



Text selection, copy or send to logger function.



To select a word.

If word is separated with spaces, just double click on it or select manually with left mouse click and drag for selection. If word(s) is selected, right mouse click pop ups the menu. Attention: selected text and menu is active for 3.5 sec. After that time it disappears.

„Copy“ function copies raw text, with spaces, to Clipboard. „Paste“ you can to notepad or logger.

Next items in menu are for logger commands. Selected text formatting without spaces, exception INFO and sending by format:

CALL:ANYCALL

RST:599 (Attention if RST text 5NN or 57N this text is sent without correction to 599 or 579)

NAME:ANYNAME

QTH:LOCATION

NR:001 (Attention if text TT1 this text sending without correction to 001)

INFO:LOCATOR OR ANY TEXT; ANT DIPOLE ETC. (without spaces removing)

All info sending via internal **UDP address: 127.0.0.1, port: 15050**

Cw Decoder Logic – Sender source example:

```
int    portNum = 15050;
string IP      = "127.0.0.1";
string word    = "CALL:LY3H";

byte[] byteSend = Encoding.ASCII.GetBytes(word);

UdpClient udpClient = new UdpClient();
udpClient.Send(byteSend, byteSend.Length, IP, portNum);
```

UDPLogger – Listener source code:

```
using System;
using System.Text;
using System.Net.Sockets;
using System.Net;

public class TCPLogger
{
    private const int portNum = 15050;

    public static int Main(String[] args)
    {
        bool done = false;

        UdpClient listener = new UdpClient(portNum);
        IPEndPoint IPEP = new IPEndPoint(IPAddress.Any, portNum);

        Console.WriteLine("Logger waiting connection via internal UDP address: 127.0.0.1, PortNum: 15050 \n\n");
        Console.WriteLine("Data format waiting:\n\n");
        Console.WriteLine("CALL:ANYCALL\n");
        Console.WriteLine("RST:599\n");
        Console.WriteLine("NAME:ANYNAME\n");
        Console.WriteLine("QTH:ANYLOCATION\n");
        Console.WriteLine("NR:001 \n");
        Console.WriteLine("INFO:K024IS or any info etc INFO:ANT DIPOLE RIG TS590\n\n");

        while (!done)
        {
            Console.WriteLine("Waiting data...");

            byte[] getBuff = listener.Receive(ref IPEP);

            Console.WriteLine("accepted.\n");

            Console.WriteLine(Encoding.ASCII.GetString(getBuff, 0, getBuff.Length) + "\n");
        }

        listener.Close();

        return 0;
    }
}
```