.

Project name: **Splitter/Watchdog**

Document name: **Description**

Customer: NewVoice

Ver: 0.1
Date: 8/02/2005

Contents.

1. Background

Current documents gives a detailed description of operations of Splitter/Watchdog device, developed by New Voice.

UPDATE NOTE (27.09.2004):

firmware **VER2COM (** THE SLAVE'S EEPROM NEEDS TO BE REPROGRAMMED ALSO ! **)** is able to choose COM port's speed by command "txyz" where x,y and z are index(ASCII char) of speed on Master, IP and Slave COM port respectively. The following speeds are avaible:

Index - speed
0 - 300
1 - 600
2 - 1200
3 - 1800
4 - 2400
5 - 4800
6 - 7200
7 - 9600
8 - 14400
9 - 19200
A - 38400
B - 56000

Speeds of index '0', 'A', 'B' are avaible only for IP COM port, because with TUSB32 speed are limited by crystal frequency. The speeds are stored in EEPROM.
If other index than above is specified in command, then the default index '7' will be used.
The number of stop bits and parity are possible to change only on IP-COM port. If you need to use this feature, please specify. On built-in UART of TUSB32 the parity checking needs to be implemented by software. Thus, code space must be taken into consideration.

UPDATE NOTE (17.05.2004):

you can configure (i.e. use ‘c’, ’a’ and ’d’-commands) WD through tel. line.
During, tel. inbound (or, if you want, outbound) call, WD is acting according it’s ‘d’-parameters line, and, When character ‘g’ is reached (‘g’-mode), WD will receive all pairs of DTMF tones as HEX codes, and deal with them as if they were entered through COM1 or COM2. This ‘g’ - mode is over when WD will receive corresponding command (or wrong command).
For example: take a look at red string – it is responsible for answering to inbound call:

> “d40j00jw1-321g-123ew1_41ArA-1111ee” – mean wait a second after hang-up, play 321, enter in configurable mode, play 123 after, hang-off.
> Now, to enter 13/10”c00010”13/10 through tel. line, you can use modem and enter via HyperTerminal following: “ATDT409@82D0DA633D3D3D313DDAD0” where “409” – WD’s tel. number, and “82” – checksum of all following bytes (!)
> (remember: 1..9,0,*,#,A,B,C,D corresponds to 1..9,A,B,C,D,E,F,0 in HEX)
> To turn on buzzer (13/10“a1”13/10) use “ATDT409@#DD0DA6131D0DA” , #D (C0) – checksum.

As you noticed, first byte, received from tel. line is considered as checksum of all following bytes.
WD hang-ups when some error is detected (too long entered line, syntax, checksum) , otherwise – if entered line was accepted – WD continues acting according to ‘d’- parameters.
In first example, the last played DTMF tones “123” can be as indication of correct ending of ‘g’-mode. If the were hang-up before playing these “123” tones, then entered line was not accepted.

Some new debug-messages:
“TCFG” – entering in parameter-line receiving mode (‘g’)
“tk” – end of tel. call.

UPDATE NOTE (19.05.2004) :

Now, WD is checking its IP_COM port for “Rxxx…13/10” string i.e. for IPcontacts status line. On first detection (after power-on, i.e. WD do not write it to EEPROM) and later, if it comes changed, WD initiates alarm outbound call.
To make WD to transmit IPcontacts by DTMF just put ‘t’ or ‘T’ (short or long tones) in ‘d’ parameters line.

For example, here we put it into alarm sequence (call in case of alarm or if IPcontacts were changed) :
“d40j00jwg-123eew1_41ArAT-1111e” : WD hook-offs,waits 1 sec, calls 410, waits 10secs for any DTMF, transmits IPcontacts and “1111”.

Additional debug message: ‘*’ – if there was detection of changed ‘R’ line from IPcom

2. Description of firmware operation and commands.

2.1. Command c

Command '***c'*** is used to specify specific timeout separately for master, slave and IPBox., to control debug messages output, and to set Relay's logic.
Master's timeout
If data is not received from master PC for time specified as master's timeout, then splittrer's alarm event is launched. Splitter starts calling out with parameters, specified in 't' command.
Slave's timeout
If data is not received from slave PC for time specified as slave timeout, then splittrer's alarm event is launched. Splitter starts calling out with parameters, specified in 't' command.
The same with IPbox's timeout.

Use "c***xxxyz***" (***x,y,z*** – ASCII digits) command to configure:

Timeouts for Splitter's alarm use 'c' command (first and second '***x***' will indicate timeout for com1 and com2 in minutes. Third '***x***' corresponds to IPbox timeout in seconds minus 60, and divided by 5 *(Example: com1 – 1 min, com2 – 10 min, IPbox – 181 sec "**xxx**" = "1AP")* . '0' – timeout does not checked.

Debug messages output ('***y***': '0' – no, '1' – only master, '2' – only slave , '3' – both)
if command sent by Master or Slave will switch debug_message_output configuration (on or off) of the other COM port, the changes will take effect only after power reset. i.e. you can switch debug_message_output only by the same COM that was used for sending current command
Debug Messages will appear only if last character from IPb was #13 or #10. If in 2 seconds from last received character from IPb any of these two (#13 or #10) characters will not be received, Splitter will act like they were received (i.e. send debug messages).

Relay's logic ('***z***': '0' – positive , '1' - negative)

Please note that in last two cases DTMF sequence
"`1 2 3 4 5 6 7 8 9 0 * # A B C D`" will correspond to ASCII digits
"`1 2 3 4 5 6 7 8 9 A B C D E F 0`" needed to be used in the parameter line.

You could use "ASCII digit" as
`'1'...'9','A','B','C','D','E','F','G','I'....'z'` to set corresponding values of
` 1 ... 9 ,10 ,11 ,12 ,13 ,14 ,15 ,16 ,17 ....66`

2.3. Command d

Use “d*uvwxyz[….]*e*[….]*e*[….]*e” to set WD’s functioning.

‘*u*’ – seconds of delay before hang-off : after the delay is expired , WD checks for ring-in tone presence. (‘0’ – no answer)

‘*v*’ – number of line check times between Call-ins. (‘0’ – no line checking)

‘*w*’ – minutes between line checks divided by five

*For example if you want one linecheck in call-in period of 20 min., you should set ‘**v**’ to '1' and ‘**w**’ to '2' (2*5 = 10)*

‘*x*’ and ‘y’ – beeper’s and relay’s seconds divided by five respectively '0' - for no action, and 'j' (see comments below) - for keypreesed deactivation.

‘*z*’ - Calling Out Period in minutes divided by 5. 255 ('j') will mean no CallingOut

First, second and third ***[…]*** – optional (all three).

First – acting sequence for WD’s Call_IN(answering on incoming call).
Second – Calling Out sequence which will be activated on CallOut timeout .
Third – Calling Out sequence which will be activated on either Splitter’s or Watchdog’s alarm.

To transmit DTMF sequence use '-' or '_' followed by appropriate character (as with Splitter's command line).
To receive known DTMF tones you must use 'r' followed by one character representing timeout in seconds in which you are expecting all following (after ”rx”) tones .
NOTE: in second and third ***[…]*** ,in case you would expect any DTMF, you must use only “rx” – i.e. you should specify only timeout (‘x’ secs) in which. you expect any DTMF.
To wait you must use 'w' followed by ASCII digit of desired seconds minus 1 (so to wait one second, use '0')
To check for CallProgress tone – use 'c' followed by one character representing timeout in seconds in which you expect CP tone.
If “Line Busy tone ” will be detected during Calling Out, the device will hang-up and outbound call will be retried.
If received tone does not match expected during incoming call, WD hangs-up. Call-in is considered succesful if all received tones corresponds to expected (specified).

2.4 Command a

“a1” – turns beeper on.
“a2” – turns beeper off.
“a3” – turns relay on.
“a4” – turns relay off.
“a5” – turns Alarm CallingOut sequence on .
“a6” – makes WD’s CallingOut timeout minutes to ‘0’ – i.e. turns WD’s CallingOut sequence on

LastErrorStates:

```
1 - Tel.Line Down
 2 - Number Busy
 3 - NoAnswer
 4 - Answer was Cancelled
```

```
    5 - DTMF Receiving Timeout
    7 - If, after reading TimeOuts from EEPROM, byte representing TimeOut for
Master/Slave was greater than #122 (ASCIIs 'z') or byte representing TO for
IPb was greater than #94 (ASCIIs '^'), and all three TOs were set to 0 (i.e.
we will not check them)
    8 - Master Computer is not reading from USB (PC are loosing new messages
from IP)
    9 - Bad (unrecognized) parameters in WD ParamLine
    A - receiving DTMF tone did not correspond to WD ParamLine's one
    B - Time Out was encountered for WD's Dial_In procedures
    C - Bad (unrecognized) parameters in SplittersParamLine
    D - Time Out Detected for IPb
    E - Time Out Detected for Master
    F - Time Out Detected for Slave
```

Info commands:

"d#10#13" or

"d#13#10" - returns 64 byte area of EEPROM containing 'd' parameter line

"c#10#13" or

"c#13#10" - returns “***xxxxx***LES***yyyy***J8***[….]***VER***[…]***” where “***xxxxx***” stands for entered ‘c’- parameter, “***yyyy***” – four last error states, ***[….]*** after “J8” – closed jumper list on J8 (‘1’,’2’ and/or ‘3’, if any) and ***[…]*** after “VER” – version’s string.

Debug messages:

When configuring:

’k’ – accepted.

‘D’ – some kind disorder detected, entered line not accepted.

‘M’ – too long, not accepted.

When working:

"TO3" – data timeout from IPbox.

"TO2" – data timeout from COM1.

"TO1" – data timeout from COM2

"MSTRRST" – master reset pressed

"ALRMRST" – slave reset pressed

“CHLINE” – tel. line down. Check the line.

“TODTMF” – DTMF tone not detected.

“TOCP” – CallProgress tone not detected.

“LChk” – Start of tel. line check.

“SLUSB” – Slave USB connection init.

"cr" – Ring In signal was not detected after WD’s Ring In timeout.

"lineD" – tel. line has a busy tone.

"LINEk" – tel. line has correct status

"CPk" -

When calling out:

'w' – wait

'c' – check for CallProgress

'k' – successful calling end.

When answering to inbound call:

"OK" – received DTMF tone the same as expected

"bad" - received DTMF tone does not match to expected

3. Device PCB' layout and connections

The board is equipped with two TUSB3210 microcontrollers (MCUs). Both MCUs have embedded serial and USB interfaces to establish upstream connection with two hosts (PCs).

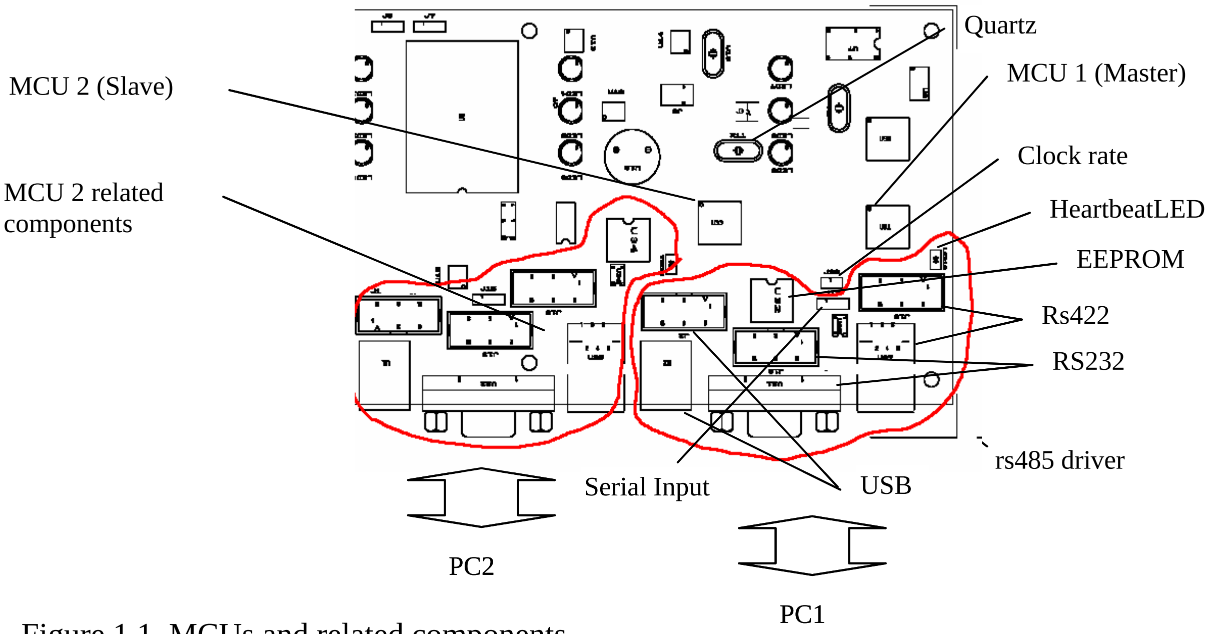
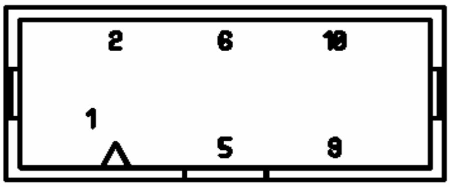


Figure 1.1. MCUs and related components

Each controller requires an EEPROM containing firmware for its operation. Heartbeat led informs that MCU is powered and is ready to execute a program. It can be used in conjunction with clock rate selecting jumper for diagnosis. This jumper should always remain open in the operating mode and will not be mounted in the future. The clock generation components consisting of the jumper and quartz highlighted in Figure 1.1 are shared between Master and Slave.



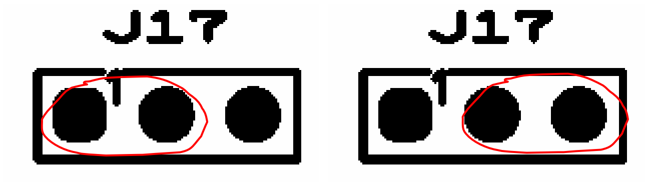
Fugure 1.2. A Ribbon Header.

Interface with two PCs is identical. Is consists of two full-duplex serial ports (rs232 and rs422) and USB connection. Each external connector is duplicated by 10-pin ribbon header to be used in the configuration with host computer located on a neighbor's PCB in the same case. This internal connector is always located behind the external socket (when looking from the edge of board outline). Pinout for both internal and external connectors is identical to that on IP-Box v3:

Rs232 connectors are of DCE type (slaves)			
signal	DB9(DCE)	Ribbon	
Gnd	5	1,2,3,4,5,6	
Tx	2	7,8	
Rx	3	9,1	

rs422		
signal	RJ45	Ribbon
Tx+	2	7,8
Tx-	3	9,1
Rx+	4	1,2
Rx-	5	3,4

Gnd	1,6	5,6



(a) Rs232 port (b) Rs422 port

Figure 1.2. Serial input selection for Master MCU. **J15** does the same for Slave.

When any of MCUs sends serial data the data is written out through both rs232 and rs422 interfaces. The serial port selection setting is related only with input. Since UART cannot process data coming from both rs232 and rs422 serial ports simultaneously the option to select beween port is illustrated in Fugure 1.2.
In addition to serial and USB interfaces that master MCU provides for connection with PC1, it controls all the remaining system and should always present. Slave MCU relays on the identical set of components to provide interface with PC2; its presence is required only when working in Splitter mode.

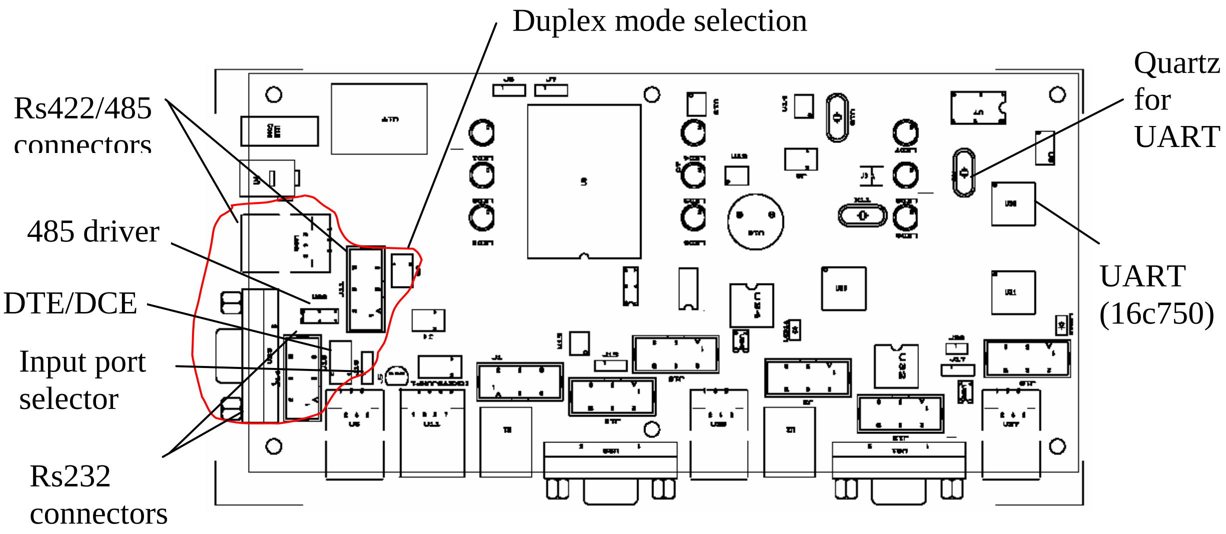


Figure 2.1. A Downstream Connection based on external UART

3.1. Analogue telephone interface

System can dial, respond on the calls and transfer DTMF tones simultaneously in both directions. Different countries have different requirements to telephone interface. Line interface (DAA) is based on Zarlink’s MH88437 device [2] to make it suitable for most countries worldwide. While most of parameters were hardwired in accordance to CTR-21 two parameters were left programmable. These are

- current limiting and
- line impedance matching

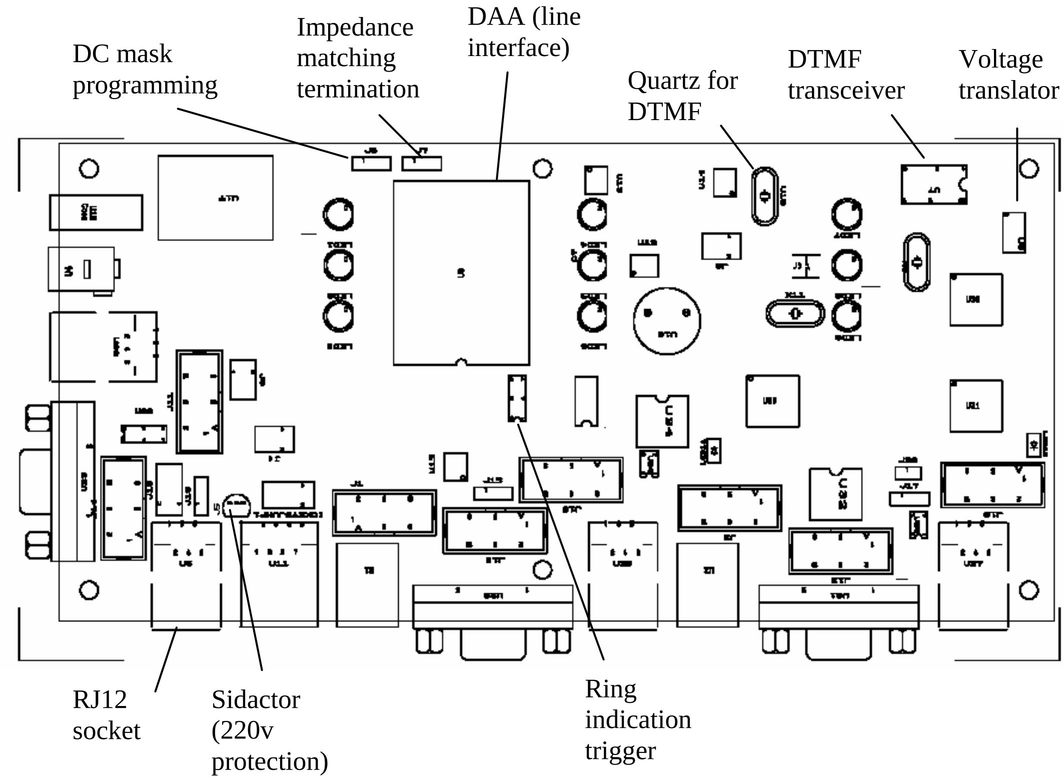
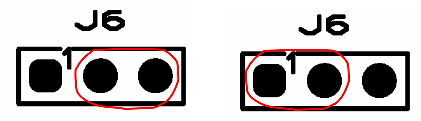
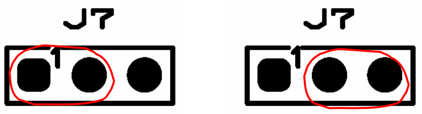


Figure 3.1. Components of telephone interface.



(a) CL\ is 0. (b) CL\ is 1.

Figure 3.2 DC mask programming

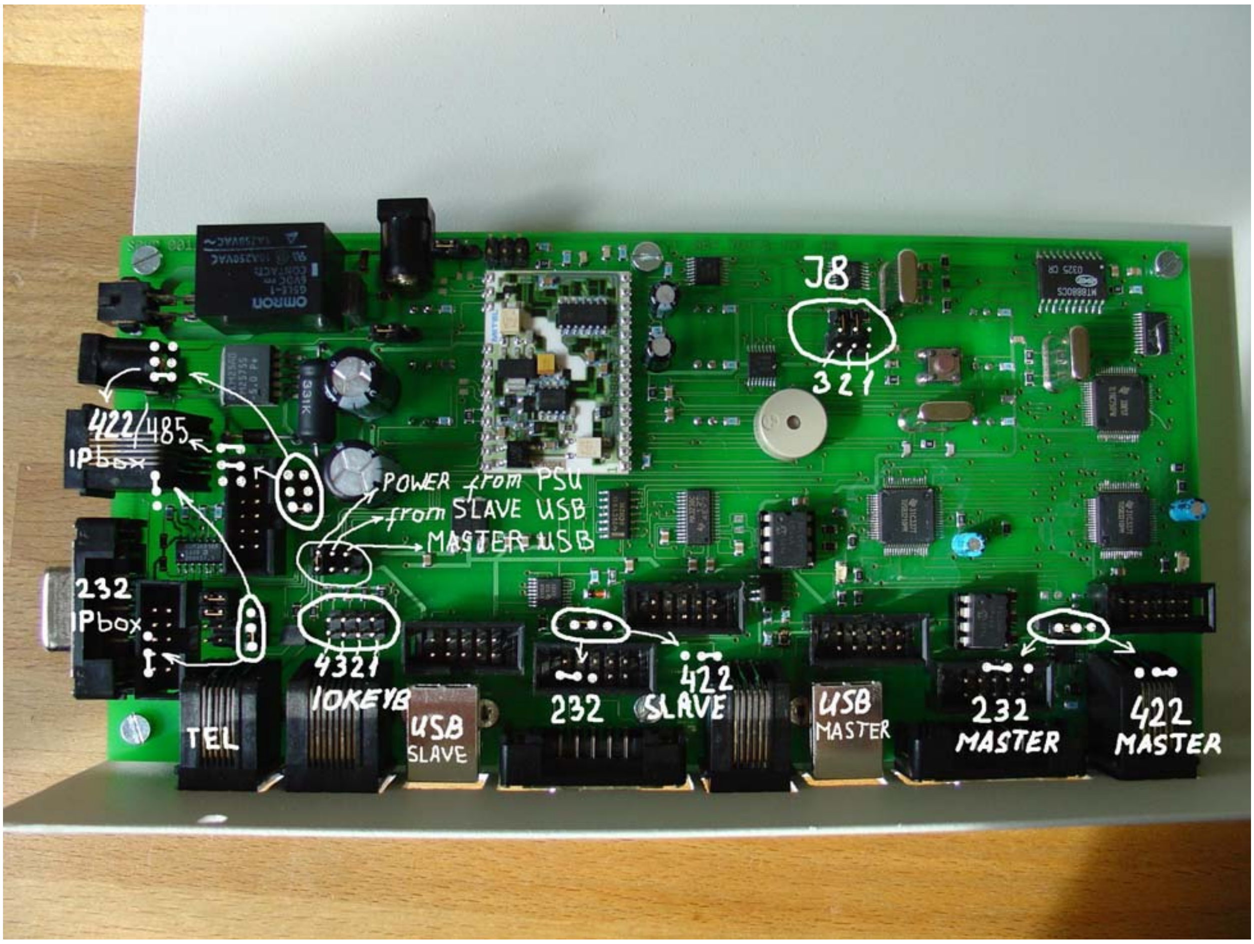


(a) SW1 Position is 1 (b) SW1 Position is 2

Figure 3.3 Line impedance programming

Table 3.1 in conjunction with figures 3.2 and 3.3 specifies jumper settings for some countries. Our experience shows that system operates with telephone reliably independently of the jumper settings. Nevertheless, this may be needed for certification. In addition to line interface hybrid function (2-wire to 4-wire conversion) this DAA chip also used to detect line termination (line may be busy by a parallel phone) and indicates ringing. Ring indication signal is periodic and stored into a nor-elements based trigger with a purpose not to be lost. Once master MCU handles the ring it resets the trigger ready to accept new calls. DTMF chip (MT8880) is intended to send/receive DTMF tones. In addition it has tone recognition output that can be used to analyze call progress. Unfortunately, when DTMF chip is in call progress mode its DTMF function is blocked. This means that when peer accepts a call it cannot send DTMF tone in response. Other algorithms of connection establishing should be used (a silence for a specific time, for example).

3.2. Firmware controlled jumpers



IOKEYB jumpers:
4 - Splitter's CallOut starts if jumper 1 on J8 is on.
3 - WD's CallOut will start.
2 - Master reset (Its differs from Alarm reset only by turning off CallReceived WD's led)
1 – Alarm Reset.

J8 jumpers:
‘1’ – enables Media Converter’s LEDindication mode.
‘2’ – enables WD’s CallOut in case of WD’s alarm
‘3’ – if ‘1’ is not set, then ‘3’ enables WD’s LEDindication interface.