How to use the

ELZET 80

PER-NET900NT

NET/900(H)-module evaluation board

AC in   DC in
~ ~   + -

5V out
+ -
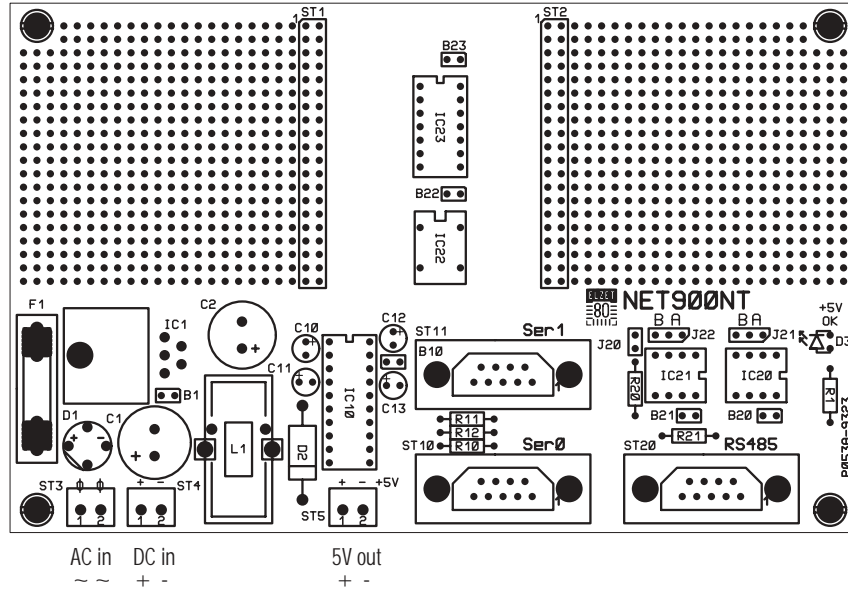
NET900NT provides NET/900 or NET/900H with 5V DC power using a DC/DC converter and minimal line drivers for the serial lines. NET900NT serves to test I/O-functions by adding drivers and connectors to the module on the breadboarding areas to both sides of the module.

Power supply

The DC/DC converter on NET900NT provides up to 2A of current at the 5V output. It requires AC or DC input.

AC input connected to ST3 must be in the range of 8 to 24V. DC input must be connected to ST4 with Pin 1 positive (left terminal) and Pin 2 ground (right terminal). 5V is indicated by the green LED.

The 5V power output is not only connected to the NET/900 module but to the top edge of the breadboarding area, too. For external devices, 5V can be accessed on screw terminal ST5 with Pin 1 positive (left terminal) and Pin 2 ground (right terminal). The NET/900 module and the NET900NT drivers require less than 200mA of current, thus up to 1.8A remain for your application. The AC (!) input is fused with 2A. If the supplying transformer does not deliver that much current, please replace by a fuse of applicable current.
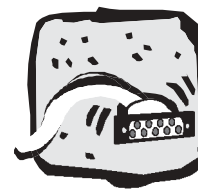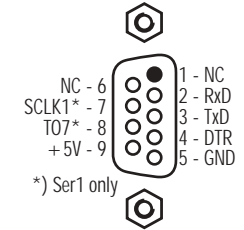
RS232

RS232 driver for SER0

The TLCS900-internal SER0 channel is accessible with RS232 levels on the 9pin-Sub-D-plug ST10. There are no programmable handshake lines for SER0. RTS and DTR outputs, however, are tied to active state by pullups thus generating correct signals if power is applied to NET900NT. Pin 9 carries 5V supply to power external devices (like LCDTERM) through the serial line cable.

RS232 driver for SER1

The TLCS900-internal SER1 channel is accessible with RS232 levels on the 9pin-Sub-D-plug ST11. With an AT-like pinout RTS is generated by the SCLK1-Pin while the CTS input goes to P37 of the TLCS900.
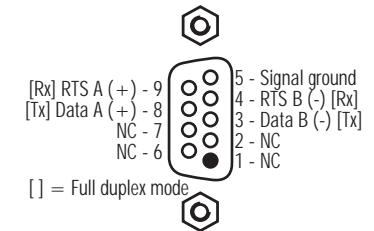
RS485

ELZET 80 supports the BITBUS fieldbus using the external serial controller "IUSC" on the NET/900 module. BITBUS uses RS485 as physical media, so NET900NT provides RS485 drivers for this purpose. BITBUS is a half-duplex protocol, sending and receiving over the same pair of wires. There is an additional driver for RTS, a signal that indicates (usually with active low) that the module is transmitting. This signal is only required for external repeaters used if the net has more than 28 nodes. J20 inverts the polarity of RTS, a jumper plugged to J20 results in a "Low" output that is required by ELZET80's repeater "BITRPT". Other repeaters might require J20 to be open.

The driver arrangement also allows for point-to-point communication or full duplex networks like DIN-MESSBUS that have separate wire pairs for receive and transmit. In this case the RTS line pair is used as receive input. The

ELZET80 standard software does not generally support these pins as assignments on EVA900 are different. DTR output (pin 4) will be tied to active state as soon as NET900NT is powered up. Pin 9 carries 5V supply to power external devices (like LCDTERM) through the serial line cable.


```
NC - 6        1 - NC
SCLK1* - 7    2 - RxD
TO7* - 8      3 - TxD
+5V - 9       4 - DTR
              5 - GND
*) Ser1 only
```

transmitter can still be switched inactive by the RTS signal (P7/TxCP of the IUSC) allowing for network applications.

Jumpers J21 and J22 are used to switch between half and full duplex: Connect the pins marked "A" to select half-duplex (BITBUS) or "B" to select full duplex operation. J20 has no effect in full-duplex mode.


```
[Rx] RTS A (+) - 9    5 - Signal ground
[Tx] Data A (+) - 8   4 - RTS B (-) [Rx]
NC - 7                3 - Data B (-) [Tx]
NC - 6                2 - NC
                      1 - NC
[ ] = Full duplex mode
```

mCAT cannot support process i/o on NET900NT as it cannot know what the customers use of the i/o pins will be. The only support provided is for the RS232 lines with the SerDrv high-level driver enabling telegram reception in the background by specifying start and end characters or message length. There is no handshake support. For the RS485 a BITBUS driver with full RAC/GBS functionality is available. Alternatively, a simple asynchronous driver supports reading and writing characters on a full or half duplex line.