



STAM1 PCI Card



stam1_pci_en 08/08

This card is a full featured receiver of the data sent by security systems or dialers via telephone line. It allows to set up central monitoring station (CMS) for supervision of security systems. It can be inserted into PCI slot of PC mainboard, however it makes use only of +12V supply and RESET signal from the bus. It can also operate outside a PC, providing supply voltage within the range of +12V..+15V. RESET signal is not necessary for standalone operation.

The card is equipped with RJ type telephone jack, to which telephone line should be connected. Minijack socket can be used for connecting high-impedance speaker (or headphones), for telephone line supervision purposes. For communication with PC, RS-232 port with DB9 connector is used.

To increase the number of CMS input ports, multiple cards can be used. In this case, DB9 connector equipped card should be used together with some (up to 15) expansion cards. All these cards are interconnected, have individual addresses set by the use of jumpers, and RS-232 equipped card is connected to free RS-232 port of a PC.

Card's firmware is stored in FLASH memory and can be upgraded by the user.

Data transmission scheme

The data between the cards and PC is sent at 4800bps speed, in 8N1 format (8 bit characters no parity with one stop bit). For data flow, only TxD and RxD signals are used; DTR signal is used for the cards to arbitrate access to the PC. Cards are sending both data on incoming events (telephone line state, handshake phase, incoming data) and response for PC's queries. PC is sending data reception confirmation or commands. Some of the data send by the cards requires acknowledge – lack of acknowledge will cause resend attempts. Less important data (additional information on connection state) not requiring acknowledge are not repeated – however it's good to acknowledge them freeing up communications channel for other cards.

Data transmission scheme is shown on the following pictures.

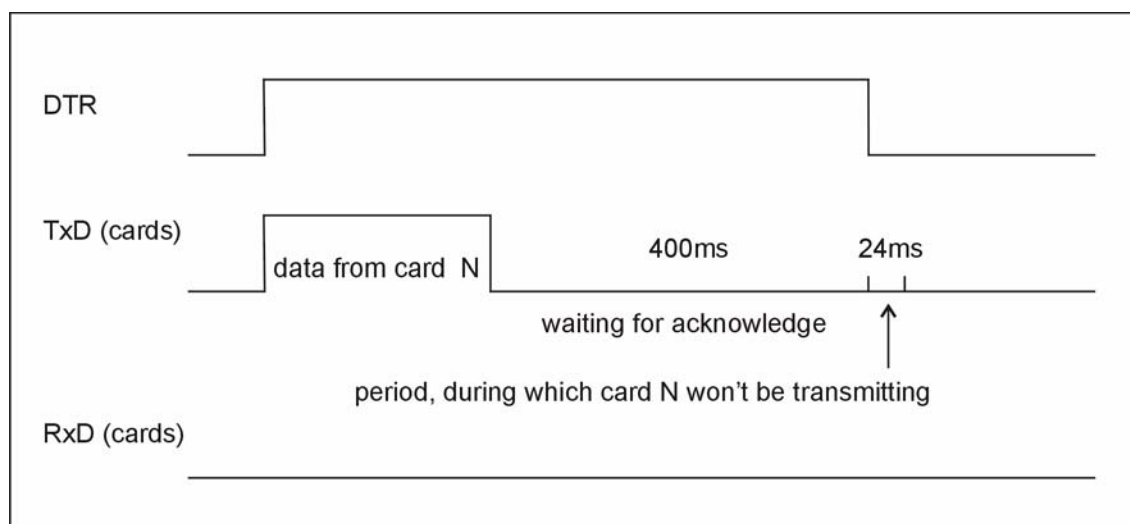


Fig.1: Data transmission not requiring acknowledge.

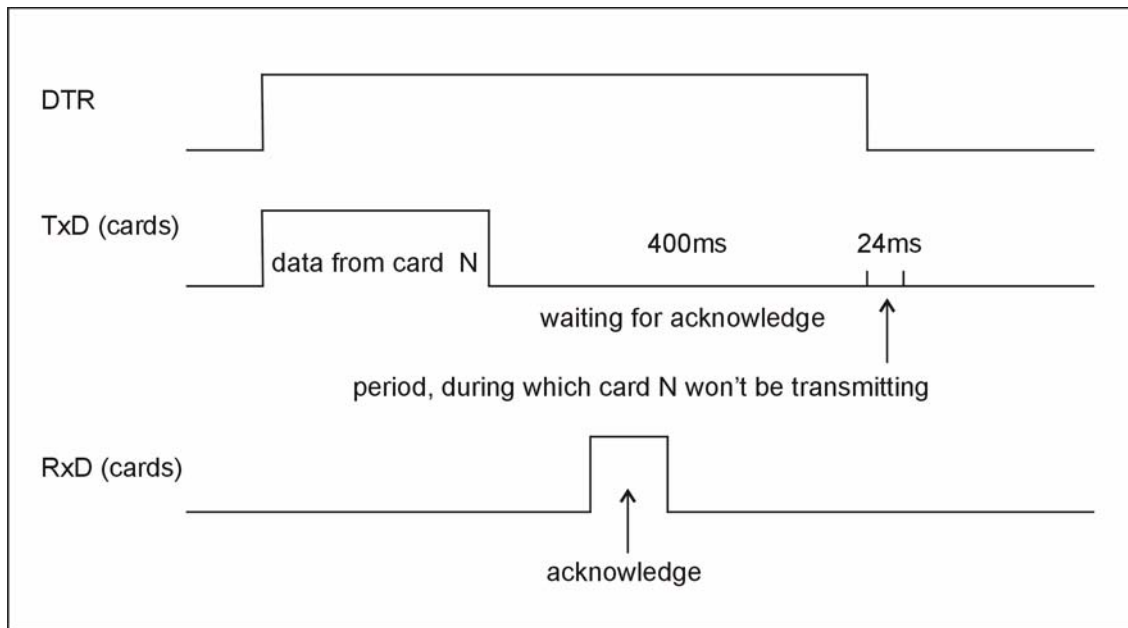


Fig.2: Data transmission with PC's acknowledge.

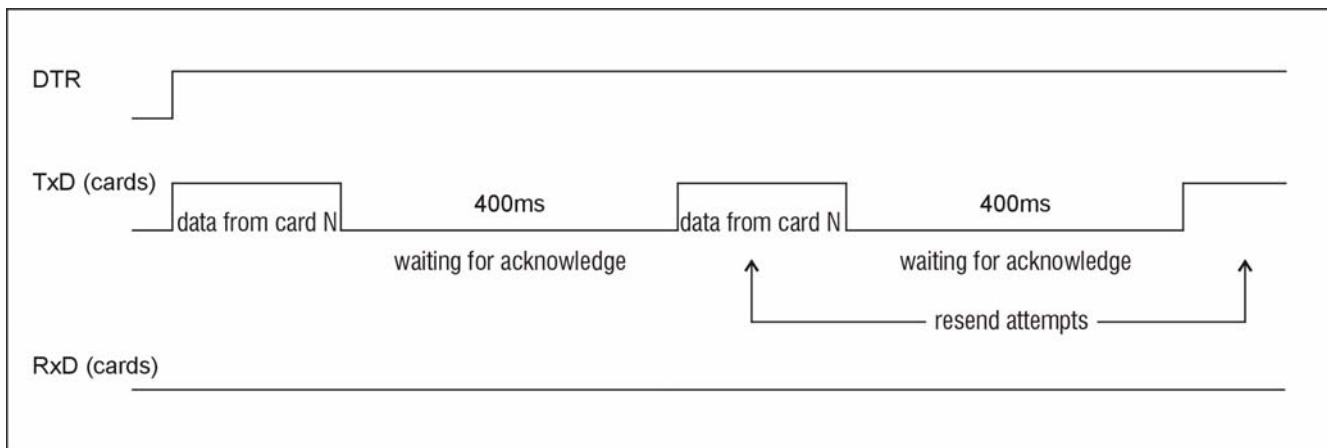


Fig.3: Data transmission (requiring acknowledge) without acknowledge.

Data sent by the cards

All the cards in the system are working asynchronously. In case of occurrence of an event which have to be transmitted to the PC, before data transfer the transmission arbitration procedure have to be applied to avoid TxD conflicts. This procedure allows to use DTR signal for the card with lowest address set. If more than one card at the time will have some data to transmit, the one with higher address will have to wait until all the cards with lower address will send their data.

Data format is as follows:

- first byte - card address (00h...0Fh)
- two following bytes - two ASCII characters (lower-case letters or lower-case letter and a digit)
- following bytes (if any) - event data (ASCII, digits or upper-case letters)

Cards can send following information (! mark indicates information requiring acknowledge):

- nrs** - card's CPU reset (!)
After a power-down or PC restart cards are sending information on restart. If the card was receiving data, restart will cause going on-hook. It may cause re-send attempt by the alarm panel which was currently communicating.
- nal** - telephone line failure detected (!)
Every 10 seconds, cards are checking telephone line voltage. If the voltage drops below 24V, the card indicates failure.
- nrl** - telephone line restore (!)
Cards are receiving data regardless they have previously reported telephone line failure. However, if telephone line failure was detected, after the telephone line will go back to normal failure flag is cancelled.
- nac** - difference between card's internal clock and PC's clock (!)
Each card is equipped with software clock, which is used for timestamping the events stored in EEPROM received from alarm system. This clock is set by PC with special command. Normally, PC should periodically send current time information (i.e. every couple of minutes). It's enough to send this command to card with address 0 as the other cards will also receive it. If the time setting sent by the PC will be different from card's clock, the card will send information on this fact. This is used to prevent event log tampering by changing PC's time. Also, after PC's failure or quitting the software, events will be properly timestamped.
- nml** - card went on hook (!)
STMON software shows the state of each card. If the card starts to receive, in the state window each phase of the reception is shown. To inform the PC about reception end, cards are sending information about going on-hook.
- nrg** - ring signal detected by the card
- nmr** - ring signal has ended
The card is answering the call after detecting programmed number of ring signals. Using these messages, it's possible to inform properly on signals detected by the card.
- nw1** - card is waiting for first data set
After going off-hook, card is sending handshakes after which is waiting for data.
- nh1** - card is sending handshake (or kissoff) at 1400Hz
Message sent for updating card's status field. This message (after w1 or w2) means that card is sending a kissoff.
- nw2** - card is awaiting data repetition by alarm panel
For pulse formats without parity check, data is verified by sending the same information twice. This message is sent after receiving first data block, when the card is awaiting for second (verification) data block.
- nh2** - card is sending handshake (or kissoff) at 2300Hz
Message sent for updating card's status field. This message (after w1 or w2) means that card is sending a kissoff.
- nd0+16bytes** – card is sending event data sent in one of supported pulse formats (!)
After successful reception of incoming event data, card stores it in EEPROM and tries to send it to the PC. If a PC acknowledges data reception, card is saving this in EEPROM and this event won't be sent to the PC again. If the cards will be working off-line (not connected to PC), all incoming events will be stored in cards' buffers (each card have got it's buffer capable of storing 250 entries in basic (pulse) formats,

or 125 entries for Ademco ContactID format). All the previously unsent events will be sent to the PC upon successful reception (and acknowledge) of incoming events. Detailed format of data sent is as follows:

n	d	0	D	D	M	M	h	h	m	m	i	i	i	i	c	c	0	0
			day		month		hour		minute		system's identifier				event code			

Examples:

1. 4/2 format event, received by card 1, 10 December at 18:36, Identifier=BA92, code 35

☺	d	0	1	0	1	2	1	8	3	6	B	A	9	2	3	5	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

2. 4/1 (or 3/2) format event, card 1, received by card 1, 3 December at 18:36, Identifier=BA92, code 3 (or BA923)

☺	d	0	0	3	1	2	1	8	3	6	B	A	9	2	3	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

nd1+16bytes - card is sending event data sent in Ademco Express format (!)

Ademco Express is fast 4/2 format using DTMF for sending data, supporting parity check. Data format for this frame is as follows:

n	d	1	D	D	M	M	h	h	m	m	i	i	i	i	2	7	c	c
			day		month		hour		minute		system's identifier				format tag	event code		

Example:

Event received by card 1, on 10 December, at 18:36: Identifier=BA92, code=35

☺	d	1	1	0	1	2	1	8	3	6	B	A	9	2	2	7	3	5
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

nd2+22bytes - card is sending event data sent in Ademco ContactID format (!)

Data is sent using DTMF. Data format for this frame is as follows:

n	d	2	D	D	M	M	h	h	m	m	i	i	i	i	Q	c	c	c	G	G	z	z	z	P
			day		month		hour		minute		system's identifier				event qualifier	event code			group	zone or user number			parity	

nv+character+3 characters – response for query on programmed number of rings before answer

Card is receiving incoming call after recognition of programmed number of ring signals. Using STMON, in Configuration, there's programmed number of ring shown near card's address. The number shown is sent with this message. Code sent is equal to ASCII code of digit zero plus number within the range between 01h..0Fh according to programmed number of rings ("1" will be sent if there's 1 ring programmed, "5" if there are 5 rings programmed and "." if there's 10 rings programmed etc.). Last 3 ASCII characters indicate card's firmware version.

nok - response for query on card's presence

nid+4 bytes – information on identifier of incoming alarm system's data

After receiving first 4 characters card is sending to the PC information on system's identifier. It allows to check if the subscriber is sending ContactID events, and in this case send back kissoff information for the card.

Commands sent by the PC

PC's commands for the cards may be sent at any time. Only data acknowledgment should be sent when the card is awaiting it.

Cards are not acknowledging data sent by the PC. If the command requires sending data back, this data acts as an acknowledgment for the command.

Special commands (two-byte):

n! - acknowledgment of data sent,

- n?** - query for data retransmission,
- n+** - switching to long kiss-off for current transmission (ContactID only).

Remaining commands have the following format:

- 1st byte - target card's number (00h...0Fh)
- three following bytes - three upper case letters (ASCII, B...T).

nTST - test for card's presence,

nRNG - query for programmed number of rings before answering incoming telephone call,

nSRB to **nSRP** – programming number of rings before answer,
(*SRB=answer after 1st ring, SRC=answer after 2nd, SRD=after 3rd etc.*)

nODB - force the answering call.

After receiving this command, card is going off-hook, sending a handshake tone and tries to receive incoming data just as after detecting programmed number of rings. It allows to check if telephone line is operating correctly (there should be a dialtone audible from the speaker connected to the card). It also allows to switch to receive mode if there's a large programmed number of rings before answer (card is used by the installer for failure monitoring, and telephone line is used with normal telephone).

nBRK - force communication break

If the system is communicating for a long time and blocking telephone line for other systems, sending this command will break current transmission and let the card go on-hook.

nBPS - beep for one second

Command allows to generate additional beeps with card's beeper. Command sent more often allows to generate continuous tone. Beeper can be used if PC speaker is not working.

nBPC - turn off card's beeper

Command allows for generating shorter beeps with card's beeper.

nCLK+4bytes – program card's clock

Note: command should be sent to card 0 (even if it doesn't exist). All the cards will receive it simultaneously. Data should be sent as follows:

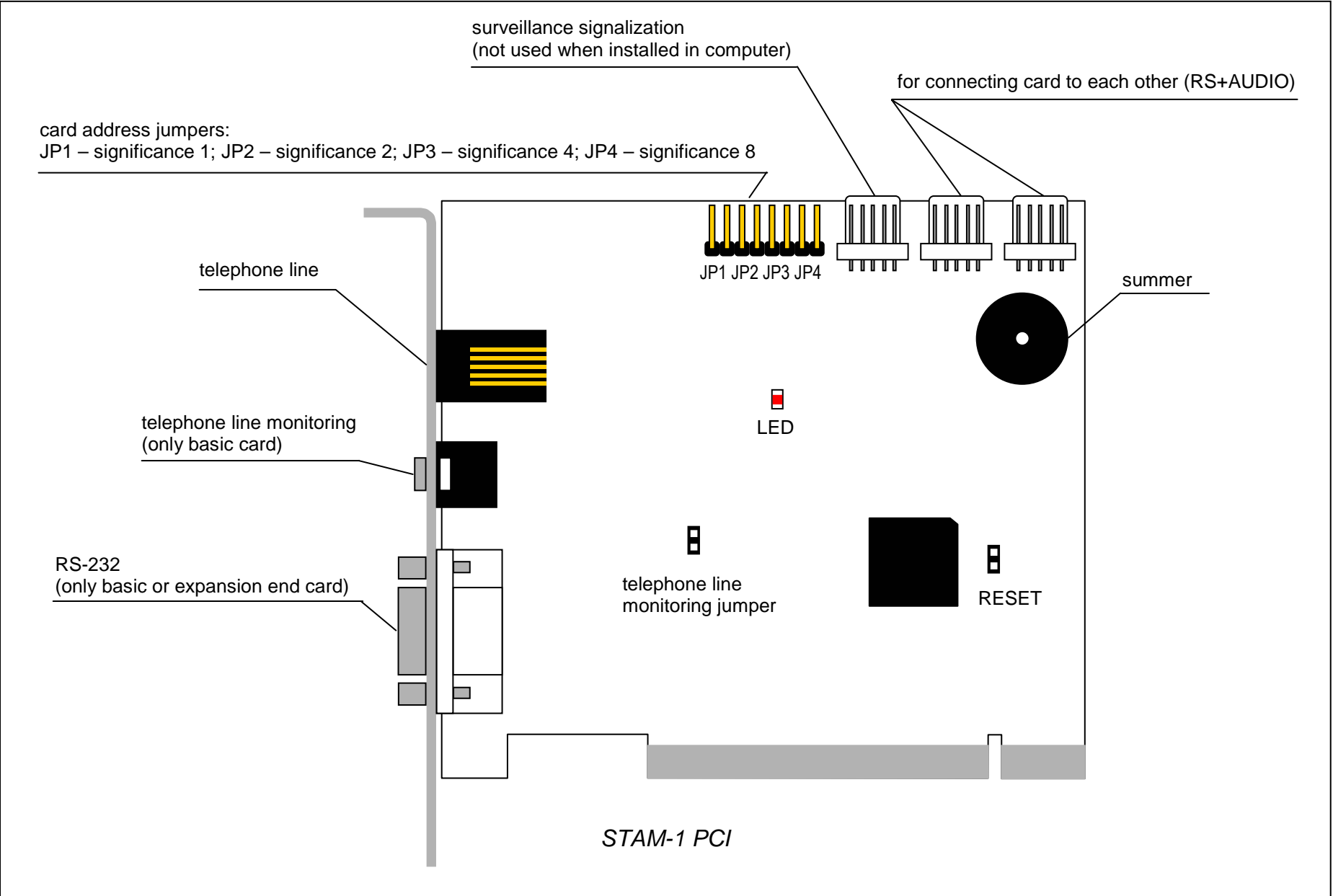
n	C	L	K						
---	---	---	---	--	--	--	--	--	--

BCD: hour minute day month

Example:

time programming: 10th of December, 18:36

00h	C	L	K	1	8	3	6	1	0	1	2
-----	---	---	---	---	---	---	---	---	---	---	---



SATEL sp. z o.o.
ul. Schuberta 79
80-172 Gdańsk
POLAND
tel. + 48 58 320 94 00
info@satel.pl
www.satel.pl