

# RTU-RL8xxM

## Modbus Slave RS-485 Relay Modules.

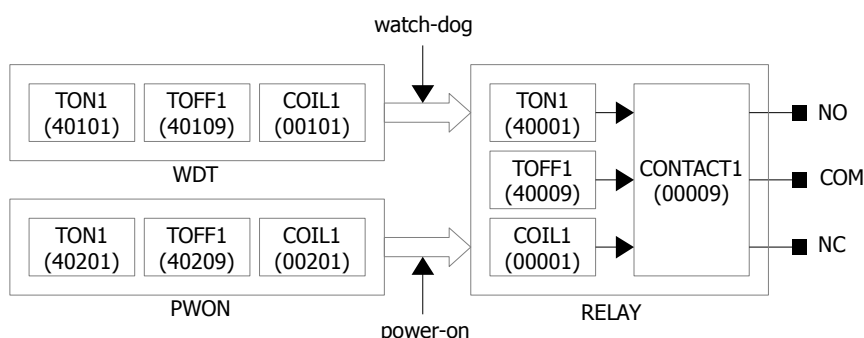
### Introduction

RTU-RL8xxM is a slave Modbus two-wire RS-485 serial communication. It connects to RS-485 network using Modbus RTU and ASCII protocols. It has 8 relays which can be configured for working in 3 different modes, on/off mode, timer mode and pulse with modulation mode. All registers are performed through serial communication. User can define state of relays when the board is powered-on or reset. Also relays can be defined state when communication fails.

### Registers

The RTU-RL8xxM has 3 groups of registers dealing with relay. They are RELAY, WDT and PWON registers. And it has another one register. It is communication register. The communication register is used to configure Modbus network communication.

Each relay has COIL, CONTACT, TON and TOFF registers. These registers can be specified values for working in 3 different modes. ON/OFF Mode, Timer Mode and Pulse With Modulation (PWM) Mode. User can read and write the COIL, TON and TOFF registers but cannot write to the CONTACT register, just only read. The CONTACT register is an action between COM and NC contacts. The CONTACT register will be 1 when the relay turns on but it will be 0 when the relay turns off. The following is an diagram of relay1. The relay2 to relay8 have the same registers as the relay1. When the board is powered-on or reset, PWON registers load into RELAY registers. And the WDT registers load into RELAY registers when the communication-watchdog timer expires.



### ON/OFF Mode

This mode allows user to turn on/off. If the user wants to turn on relay. The user has to write 0 into TOFF, 1 into COIL registers and value of TON register must more than 0.

But if the user wants to turn off the relay. The value in the COIL must be 0. If the value in the COIL register is not 0, the user has to write 0 into the TON register.

When the relay turns on, value of the CONTACT register will be 1.

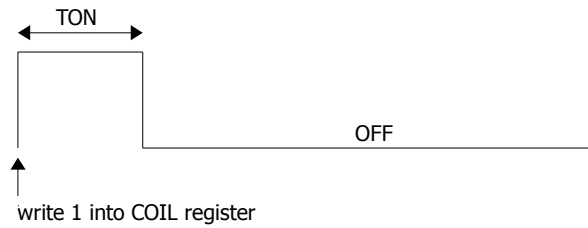
Write Values			Read Value
COIL	TON	TOFF	CONTACT
1	1 - 65535	0	1
1	0	don't care (0 - 65535)	0
0	don't care (0 - 65535)	don't care (0 - 65535)	0

### Timer Mode

This mode allows user to turn on relay for a desired time. By writing 65535 into TOFF register and set a desired time into TON register. The value in TON register must more than 0. After 1 is written into COIL register the relay contact will turns on. The relay contact turns off when TON timer expires.

While the relay turns on, value of the CONTACT register will be 1.

Write Values			Read Value
COIL	TON	TOFF	CONTACT
1	1 - 65535	65535	"1" for a specified time of TON and then go to "0"

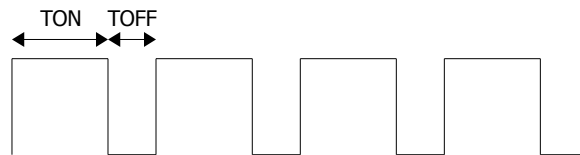


**Pulse With Modulation (PWM) Mode.**

This mode allows user to set each relay to generate pulse with modulation signal. The user has to write a desired time-on period into TON register and write desired time-off period into TOFF register. The relay generates pulse with modulation signal when the value in the COIL register is 1.

The user can read the CONTACT register to know state of relay. When the relay turns on, value in the CONTACT register is 1, otherwise the value is 0.

Input Value			Output Value
COIL	TON	TOFF	CONTACT
1	1 - 65535	1 - 65534	0, 1 (PWM)



**Register Summary**

REGISTERS	ADDRESS	DESCRIPTION	VALUE		UNIT	FUNCTION		
			MIN.	MAX.		WRITE	READ	R/W
COIL1	00001	coil of relay1	0	1		FC05, FC15	FC01	R/W
COIL2	00002	coil of relay2	0	1		FC05, FC15	FC01	R/W
COIL3	00003	coil of relay3	0	1		FC05, FC15	FC01	R/W
COIL4	00004	coil of relay4	0	1		FC05, FC15	FC01	R/W
COIL5	00005	coil of relay5	0	1		FC05, FC15	FC01	R/W
COIL6	00006	coil of relay6	0	1		FC05, FC15	FC01	R/W
COIL7	00007	coil of relay7	0	1		FC05, FC15	FC01	R/W
COIL8	00008	coil of relay8	0	1		FC05, FC15	FC01	R/W
CONTACT1	00009	contact of relay1	0	1		FC05, FC15	FC01	R
CONTACT2	00010	contact of relay2	0	1		FC05, FC15	FC01	R
CONTACT3	00011	contact of relay3	0	1		FC05, FC15	FC01	R
CONTACT4	00012	contact of relay4	0	1		FC05, FC15	FC01	R
CONTACT5	00013	contact of relay5	0	1		FC05, FC15	FC01	R
CONTACT6	00014	contact of relay6	0	1		FC05, FC15	FC01	R
CONTACT7	00015	contact of relay7	0	1		FC05, FC15	FC01	R
CONTACT8	00016	contact of relay8	0	1		FC05, FC15	FC01	R
TON1	40001	time-on period of relay1	0	65535	100 ms	FC06, FC16	FC03	R/W
TON2	40002	time-on period of relay2	0	65535	100 ms	FC06, FC16	FC03	R/W

REGISTERS	ADDRESS	DESCRIPTION	VALUE		UNIT	FUNCTION		
			MIN.	MAX.		WRITE	READ	R/W
TON3	40003	time-on period of relay3	0	65535	100 ms	FC06, FC16	FC03	R/W
TON4	40004	time-on period of relay4	0	65535	100 ms	FC06, FC16	FC03	R/W
TON5	40005	time-on period of relay5	0	65535	100 ms	FC06, FC16	FC03	R/W
TON6	40006	time-on period of relay6	0	65535	100 ms	FC06, FC16	FC03	R/W
TON7	40007	time-on period of relay7	0	65535	100 ms	FC06, FC16	FC03	R/W
TON8	40008	time-on period of relay8	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF1	40009	time-off period of relay1	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF2	40010	time-off period of relay2	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF3	40011	time-off period of relay3	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF4	40012	time-off period of relay4	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF5	40013	time-off period of relay5	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF6	40014	time-off period of relay6	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF7	40015	time-off period of relay7	0	65535	100 ms	FC06, FC16	FC03	R/W
TOFF8	40016	time-off period of relay8	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT COIL1	00101	comm-watchdog coil of relay1	0	1		FC05, FC15	FC01	R/W
WDT COIL2	00102	comm-watchdog coil of relay2	0	1		FC05, FC15	FC01	R/W
WDT COIL3	00103	comm-watchdog coil of relay3	0	1		FC05, FC15	FC01	R/W
WDT COIL4	00104	comm-watchdog coil of relay4	0	1		FC05, FC15	FC01	R/W
WDT COIL5	00105	comm-watchdog coil of relay5	0	1		FC05, FC15	FC01	R/W
WDT COIL6	00106	comm-watchdog coil of relay6	0	1		FC05, FC15	FC01	R/W
WDT COIL7	00107	comm-watchdog coil of relay7	0	1		FC05, FC15	FC01	R/W
WDT COIL8	00108	comm-watchdog coil of relay8	0	1		FC05, FC15	FC01	R/W
WDT TON1	40101	time-on of relay1 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON2	40102	time-on of relay2 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON3	40103	time-on of relay3 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON4	40104	time-on of relay4 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON5	40105	time-on of relay5 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON6	40106	time-on of relay6 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON7	40107	time-on of relay7 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TON8	40108	time-on of relay8 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF1	40109	time-off of relay1 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF2	40110	time-off of relay2 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF3	40111	time-off of relay3 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF4	40112	time-off of relay4 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF5	40113	time-off of relay5 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF6	40114	time-off of relay6 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF7	40115	time-off of relay7 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
WDT TOFF8	40116	time-off of relay8 when communication fails	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON COIL1	00201	coil of relay1 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL2	00202	coil of relay2 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL3	00203	coil of relay3 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL4	00204	coil of relay4 when power-on	0	1		FC05, FC15	FC01	R/W

REGISTERS	ADDRESS	DESCRIPTION	VALUE		UNIT	FUNCTION		
			MIN.	MAX.		WRITE	READ	R/W
PWON COIL5	00205	coil of relay5 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL6	00206	coil of relay6 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL7	00207	coil of relay7 when power-on	0	1		FC05, FC15	FC01	R/W
PWON COIL8	00208	coil of relay8 when power-on	0	1		FC05, FC15	FC01	R/W
PWON TON1	40201	time-on of relay1 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON2	40202	time-on of relay2 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON3	40203	time-on of relay3 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON4	40204	time-on of relay4 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON5	40205	time-on of relay5 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON6	40206	time-on of relay6 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON7	40207	time-on of relay7 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TON8	40208	time-on of relay8 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF1	40209	time-off of relay1 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF2	40210	time-off of relay2 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF3	40211	time-off of relay3 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF4	40212	time-off of relay4 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF5	40213	time-off of relay5 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF6	40214	time-off of relay6 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF7	40215	time-off of relay7 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
PWON TOFF8	40216	time-off of relay8 when power-on	0	65535	100 ms	FC06, FC16	FC03	R/W
NODE	40901	node address	0	247		FC06, FC16	FC03	R/W
BAUD	40902	baud rate	see details			FC06, FC16	FC03	R/W
BITS	40903	data bits	7, 8			FC06, FC16	FC03	R/W
PARITY	40904	parity bits	0, 2, 3			FC06, FC16	FC03	R/W
STOP	40905	stop bits	1, 2			FC06, FC16	FC03	R/W
MB MODE	40906	modbus mode, RTU/ASCII	1, 2			FC06, FC16	FC03	R/W
COMM WDT	40907	timer for communication watchdog	0	65535	1 sec	FC06, FC16	FC03	R/W
RPY DLY	40908	timer for replay delay	0	65535	1 ms	FC06, FC16	FC03	R/W
VERSION	30901	software version	0	65535				R

**Register Details**

**COIL**

These register can be read and written. The values of these registers are not states of relay contacts.

Relay Action	Value In Register	Relay Mode
deactivate	0	ON/OFF
activate	1	
	1	TIMER
	1	PWM

**CONTACT**

These registers are read only. They are the current states of relay contacts, NO and COM.

Action Of Contacts (NO,COM)	Value In Register
opened	0
closed	1

**TON**

These registers can be read and written. They are used to specify time-on period of each relay.

Time-On Period (10 ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
100ms – 6,553,500ms	1 - 65535	TIMER
100ms – 6,553,500ms	1 - 65535	PWM

**TOFF**

These registers can be read and written. They are used to specify time-off period of each relay.

Time-Off Period (10 ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
	65535	TIMER
100ms – 6,553,400ms	1 - 65534	PWM

**WDT COIL**

These registers can be read and written. The values of the registers are stored in EEPROM when the register are written. When the communication-watchdog timer expires the values are loaded into COIL registers.

Relay Action	Value In Register	Relay Mode
deactivate	0	ON/OFF
activate	1	
	1	TIMER
	1	PWM

**WDT TON**

These registers can be read and written. Values of the registers are stored in EEPROM when the registers are written. When the communication-watchdog timer expires the values are loaded into **TON** registers.

Time-On Period (10 ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
100ms – 6,553,500ms	1 - 65535	TIMER
100ms – 6,553,500ms	1 - 65535	PWM

**WDT TOFF**

These registers can be read and written. Values of the registers are stored in EEPROM when the registers are written. When communication-watchdog timer expires the values are loaded into TOFF registers.

Time-Off Period (10ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
	65535	TIMER
100ms – 6,553,400ms	1 - 65534	PWM

**PWON COIL**

These registers can be read and written. The values of the registers are stored in EEPROM when the registers are written. When the board is powered-on or reset the values are loaded into COIL registers.

Relay Action	Value In Register	Relay Mode
deactivate	0	ON/OFF

Relay Action	Value In Register	Relay Mode
activate	1	
	1	TIMER
	1	PWM

**PWON TON**

These registers can be read and written. Values of the registers are stored in EEPROM when the registers are written. When the board is powered-on or reset the values are loaded into **TON** registers.

Time-On Period (10ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
100ms – 6,553,500ms	1 - 65535	TIMER
100ms – 6,553,500ms	1 - 65535	PWM

**PWON TOFF**

These registers can be read and written. Values of the registers are stored in EEPROM when the registers are written. When the board is powered-on or reset the values are loaded into **TOFF** registers.

Time-On Period (10ms)	Value In Register	Relay Mode
	don't care (0 - 65535)	ON/OFF
100ms – 6,553,500ms	65535	TIMER
100ms – 6,553,400ms	1 - 65534	PWM

**NODE**

This register is used to specify node address. It can be read and written. Values of the register are between 0 and 247. Address 0 is used for a broadcast address. When Modbus functions write node address to the register. The node address is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

**BAUD**

This register is used to specify baud rate of communication. It can be read and written. When Modbus functions write a baud rate to the register. The baud rate is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

Baud Rate	Value In Register
1200	12
2400	24
4800	48
9600	96
14400	144
19200	192
28800	288
38400	384
57600	576
76800	768
115200	1152

**BITS**

This register is used to specify data bit of communication. It can be read and written. When Modbus functions write a data bit to the register. The data bit is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

Data Bits	Value In Register
7 bits	7

Data Bits	Value In Register
8 bits	8

**PARITY**

This register is used to specify parity bit of communication. It can be read and written. When Modbus functions write a parity bit to the register. The parity bit is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

Parity Bits	Value In Register
NONE	0
EVEN	2
ODD	3

**STOP**

This register is used to specify stop bits of communication. It can be read and written. When Modbus functions write a stop bit to the register. The stop bit is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

Stop Bits	Value In Register
1 bit	1
2 bits	2

**MB MODE**

MB MODE stands for Modbus mode. This register is used to specify mode of Modbus. It can be read and written. When Modbus functions write mode to the register. The mode is written into EEPROM. It is not written into the register. The register will be updated when the board is powered-on or reset.

Modbus Mode	Value In Register
RTU	1
ASCII	2

**COMM WDT**

COMM WDT stands for communication-watchdog. This register is used to specify value of timer of communication-watchdog. The communication-watchdog is reset every time the slave board receives a command from the master. Whenever the communication between the master and the slave fails and then the communication-watchdog timer expires. The whole registers of WDT are loaded into RELAY registers. The communication-watchdog can be disabled by writing 0 into this register. The communication-watchdog is not a CPU hardware watchdog. When Modbus function write a timer value to the register. The register updates new timer value and EEPROM stores the new timer value.

Watchdog Timer (seconds)	Value In Register
1 - 65535 seconds	1 - 65535
disable	0

**RPY DLY**

RPY DLY stands for slave reply delay. It is used to specify reply delay of the slave. Normally when the master sends a command to the slave. The slave immediately sends back an answer to the master when value in the register is 0. If the value in the register is more than 0, the slave delays for a specified time and then sends an answer back to the master. This feature is used for wireless networking.

Delay Time (ms)	Value in Register
0 - 65,535ms	0 - 65535

**VERSION**

This register shows firmware version of the board. If the firmware version is 1.00, the value 100 will be read. The register is read only.

**Configurations**

RTU-RL8xxM board has to be configured before connecting them in an RS-485 network. It is recommended to check communication parameters are correctly. Though whole boards are configured at the factory.

RTU-RL8xxM has 2 modes, the first is RUN mode and the second is INIT mode. The RUN mode uses Modbus protocol for communicating between the master and the slave. The INT mode is used for setting communication parameters. In the INT mode the board communicate a Terminal program on a computer, such as Hyperterminal or Tera Term. The following is details of how to use INIT mode. All commands in the INIT mode store values in EEPROM. The values will be loaded into communication registers when the board is powered-on or reset.

**Command format**

>[Letter][Value]<enter>

\*\*\*Many commands can be written in one line. Except **R** and **V** commands.

Command Examples	Description
>A32<enter>	Node address is 32
>B96<enter>	Baud rate is 9600
>A32 B96 D8 PN S1<enter>	Node address is 32, 9600 baud rate, 8 stop bits, NONE parity bit and 1 stop bit
>R<enter>	Show all parameters on screen
>?<enter>	Show helps on screen

**Commands**

Command	Letter	Description
Address	A	This command writes slave node address into EEPROM.
Baud	B	This command writes baud rate value into EEPROM.
Data	D	This command writes data bit value into EEPROM.
Parity	P	This command writes parity bit value into EEPROM.
Stop	S	This command writes stop bit value into EEPROM.
Modbus	M	This command writes Modbus mode, RTU/ASCII into EEPROM.
Watchdog	W	This command writes communication-watchdog timer value into EEPROM.
Delay	E	This command writes reply delay value into EEPROM.
Read	R	This command reads all parameters and shows on screen.
Version	V	This command reads current firmware version and shows on screen.
Help	?	This command shows helps on screen.

**Command Details**

**Address – A**

This command stores node address in EEPROM. The value of the command between 0 and 247. Address 0 is used for broadcast address.

*Example:* >A32<enter>

*Comment* Set node address , 32 .

**Baud – B**

This command stores baud rate value in EEPROM.

*Example:* >B1152<enter>

*Comment* Set baud rate, 115200 bps .

Value	Baud Rate
12	1200
24	2400
48	4800
96	9600
144	14400
192	19200
288	28800



Value	Baud Rate
384	38400
576	57600
768	76800
1152	115200

**Data – D**

This command stores data bit value in EEPROM.

Example: >D8<enter>

Comment Set data bit, 8 bits.

Value	Data Bit
7	7 data bits
8	8 data bits

**Parity – P**

This command stores parity bit value in EEPROM.

Example: >PN<enter>

Comment Set parity bit, NONE parity bit.

Value	Parity Bit
N	NONE parity bit
E	EVEN parity bit
O	ODD parity bit

**Stop – S**

This command stores stop bit value in EEPROM.

Example: >S1<enter>

Comment Set stop bit, 1 stop bit.

Value	Stop Bit
1	1 stop bit
2	2 stop bits

**Modbus – M**

This command stores Modbus mode value in EEPROM.

Example: >MR<enter>

Comment Set Modbus mode, RTU mode.

Value	Modbus Mode
R	RTU
A	ASCII

**Watchdog – W**

This command stores communication-watchdog timer value in EEPROM.

Example: >W5<enter>

Comment Set communication-watchdog timer, 5 seconds.

Value	Time (second)
0	disable
1 - 65535	1 – 65,535 seconds

**Delay – E**

This command stores reply delay time value in EEPROM.

Example: >E10<enter>

Comment Set reply delay, 10 ms.

Value	Time (ms)
0	disable
1 - 65535	1 – 65,535 ms

**Read – R**

This command shows communication parameters on screen.

Example: >R<enter>

Comment: Show all communication parameters on screen.

**Version – V**

This command shows current version of firmware on screen.

Example: >V<enter>

Comment: Show firmware version on screen.

**Help - ?**

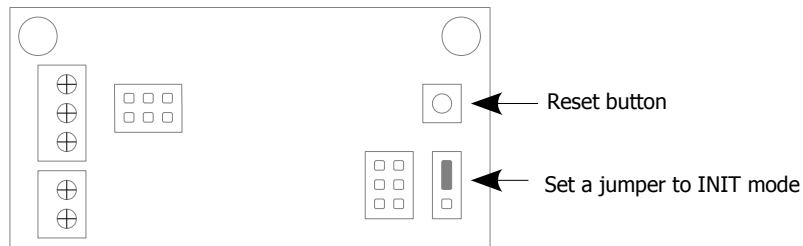
This command shows helps of these commands on screen.

Example: >?<enter>

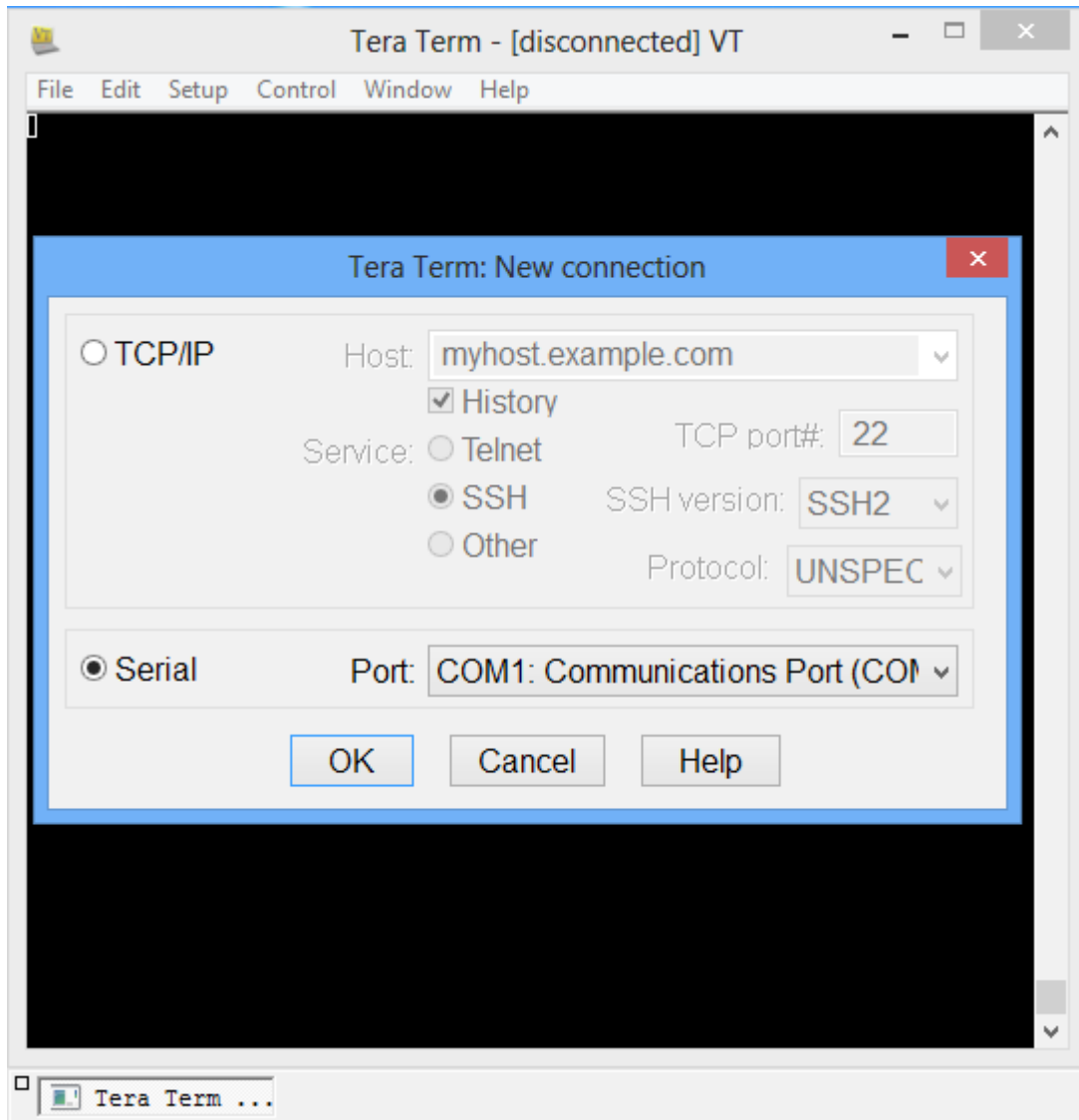
Comment: Show helps on screen.

**Tera Term For Configurations**

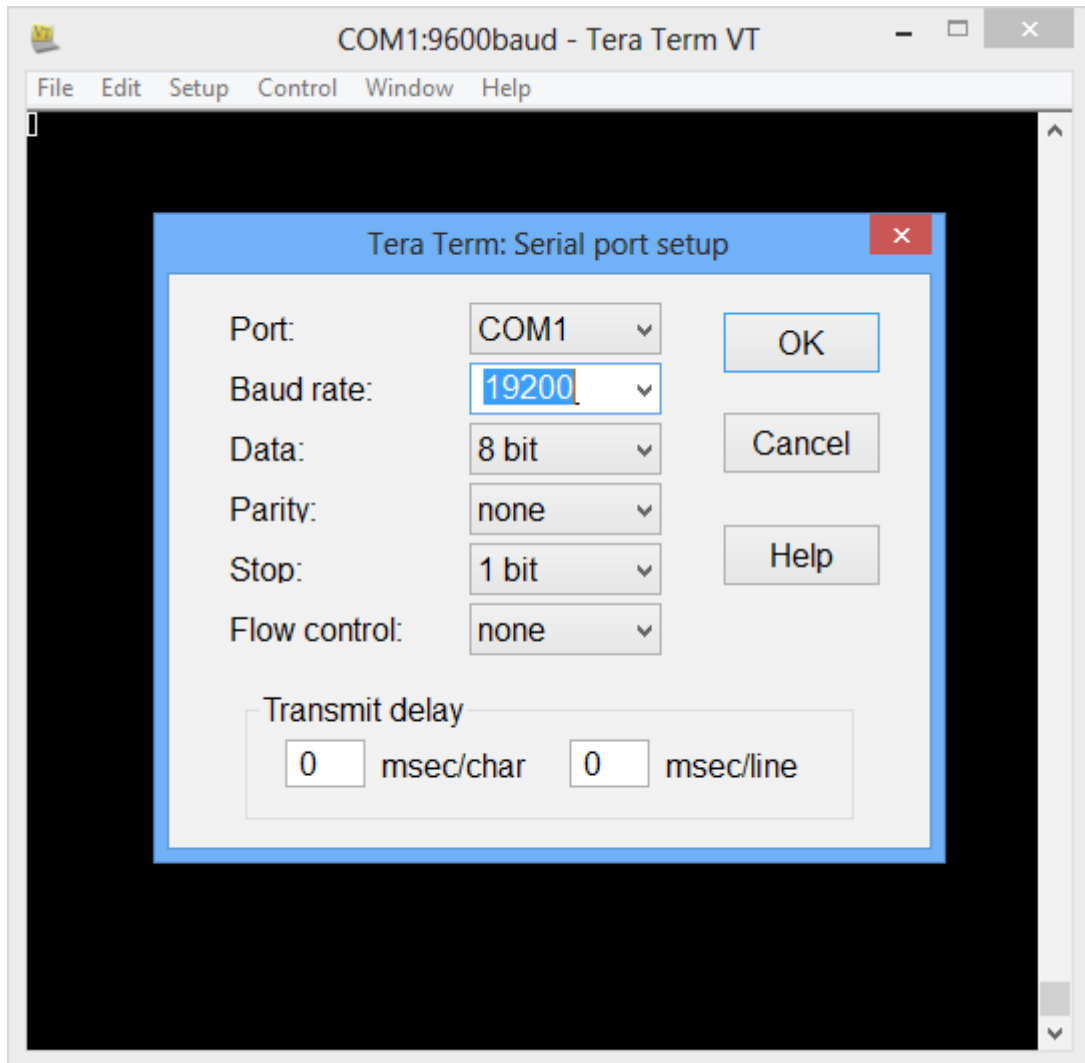
1. Connect the board to RS-485 converter(with auto direction control).
2. Connect power supply.
3. Set a jumper to INIT.



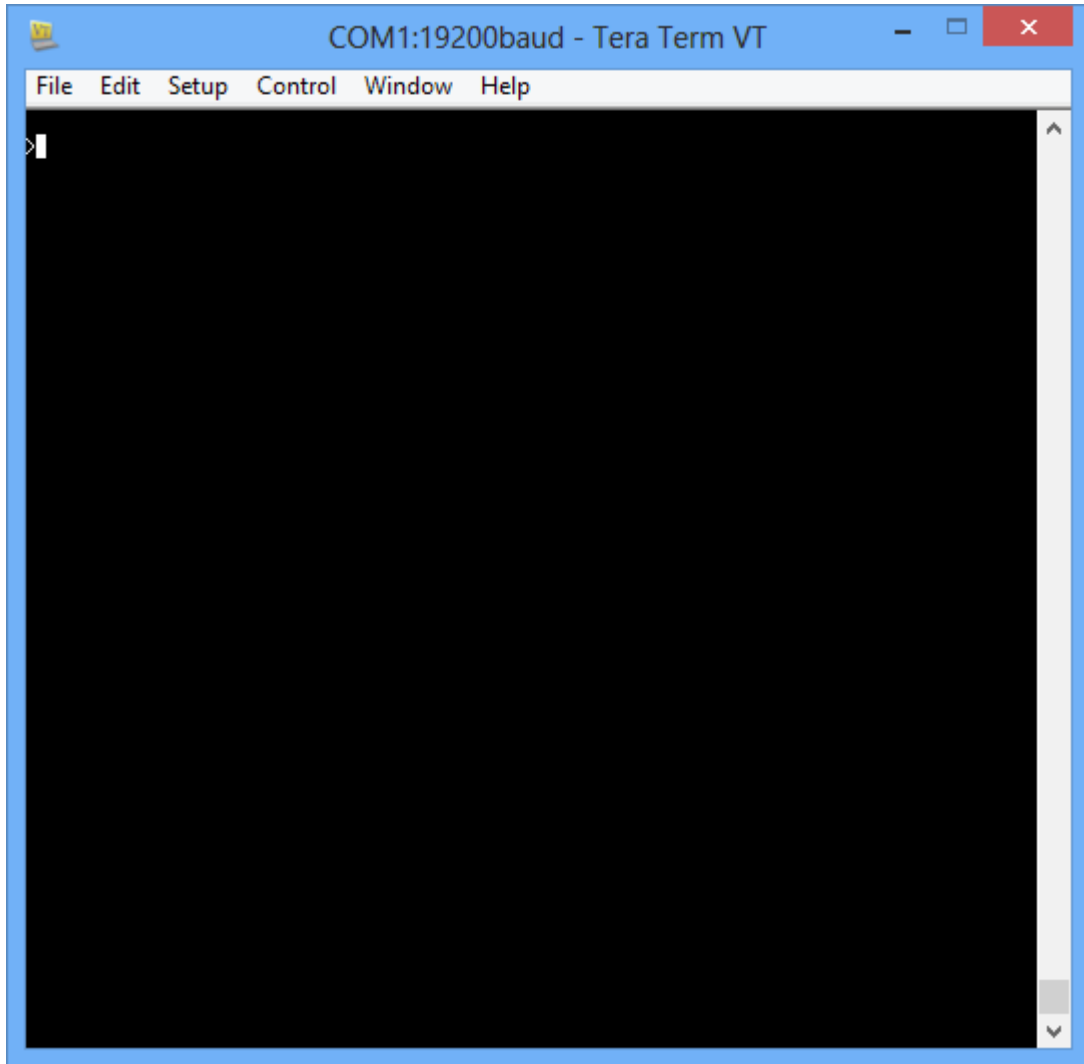
4. Run **Tera Term** program (or other serial terminal program).



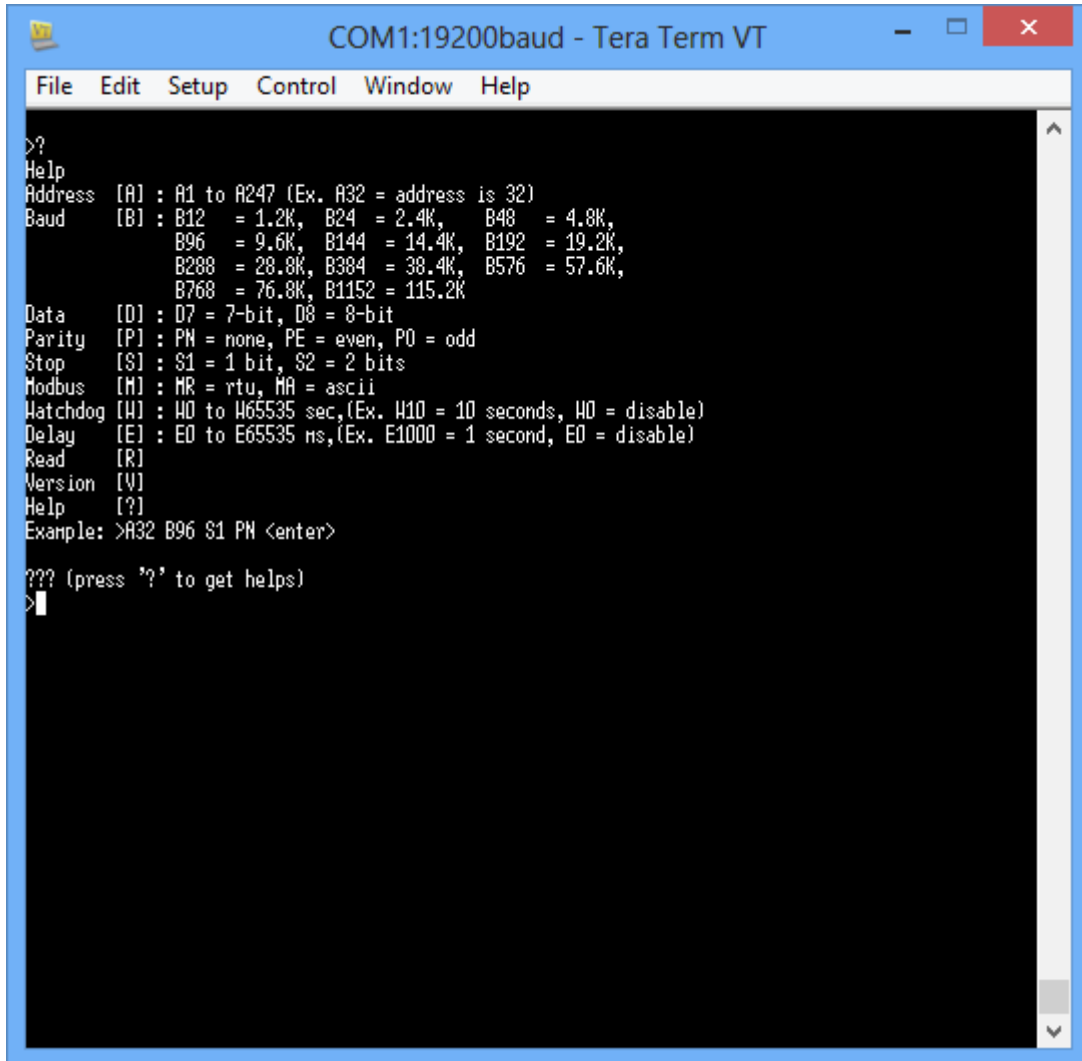
5. Select **Serial** and then click **OK** button.
6. Setup serial port from **Setup** menu.
7. Set Comport, Baud = 19200, Data = 8, Parity = None, Stop = 1, Flow control = NONE and then click **OK** button.



8. Press **RESET** button on the board the LED will blink and then press **ENTER** key on the keyboard within 5 seconds. The > will be shown.



9. Type **?** and press **ENTER** on the keyboard, helps will be shown.



```
COM1:19200baud - Tera Term VT
File Edit Setup Control Window Help
>?
Help
Address [A] : A1 to A247 (Ex. A32 = address is 32)
Baud [B] : B12 = 1.2K, B24 = 2.4K, B48 = 4.8K,
          B96 = 9.6K, B144 = 14.4K, B192 = 19.2K,
          B288 = 28.8K, B384 = 38.4K, B576 = 57.6K,
          B768 = 76.8K, B1152 = 115.2K
Data [D] : D7 = 7-bit, D8 = 8-bit
Parity [P] : PM = none, PE = even, PO = odd
Stop [S] : S1 = 1 bit, S2 = 2 bits
Modbus [M] : MR = rtu, MA = ascii
Watchdog [W] : W0 to W65535 sec,(Ex. W10 = 10 seconds, W0 = disable)
Delay [E] : E0 to E65535 ms,(Ex. E1000 = 1 second, E0 = disable)
Read [R]
Version [V]
Help [?]
Example: >A32 B96 S1 PM <enter>

??? (press '?' to get helps)
>|
```

10. Type **R** and press **ENTER** on the keyboard. The current setting parameters will be shown.

```

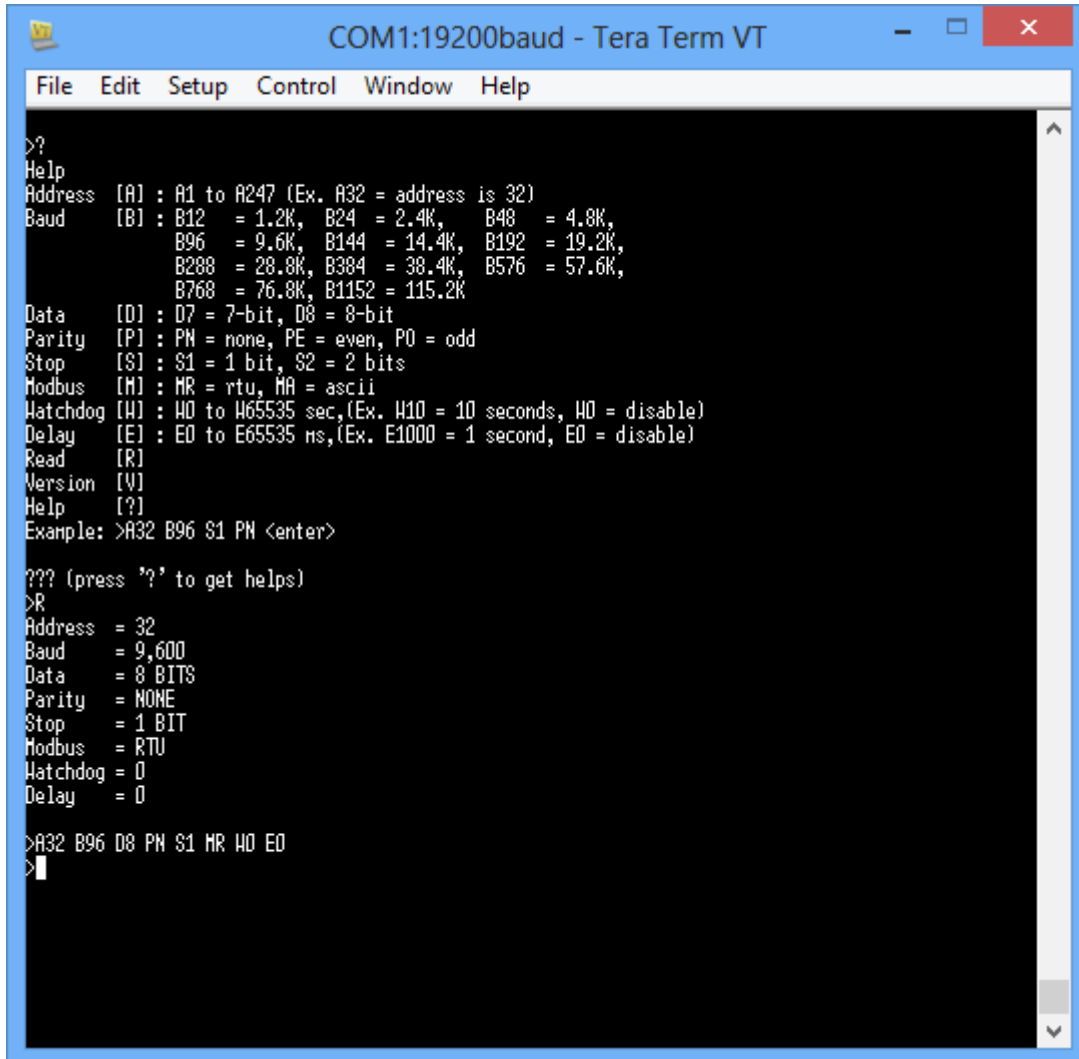
COM1:19200baud - Tera Term VT
File Edit Setup Control Window Help
>?
Help
Address [A] : A1 to A247 (Ex. A32 = address is 32)
Baud [B] : B12 = 1.2K, B24 = 2.4K, B48 = 4.8K,
          B96 = 9.6K, B144 = 14.4K, B192 = 19.2K,
          B288 = 28.8K, B384 = 38.4K, B576 = 57.6K,
          B768 = 76.8K, B1152 = 115.2K
Data [D] : D7 = 7-bit, D8 = 8-bit
Parity [P] : PM = none, PE = even, PO = odd
Stop [S] : S1 = 1 bit, S2 = 2 bits
Modbus [M] : MR = rtu, MA = ascii
Watchdog [W] : W0 to W65535 sec,(Ex. W10 = 10 seconds, W0 = disable)
Delay [E] : E0 to E65535 ms,(Ex. E1000 = 1 second, E0 = disable)
Read [R]
Version [V]
Help [?]
Example: >A32 B96 S1 PM <enter>

??? (press '?' to get helps)
>R
Address = 1
Baud = 115,200
Data = 8 BITS
Parity = NONE
Stop = 1 BIT
Modbus = RTU
Watchdog = 0
Delay = 0

>

```

11. Type commands to set parameters. The below picture is an example.



```

COM1:19200baud - Tera Term VT
File Edit Setup Control Window Help
>?
Help
Address [A] : A1 to A247 (Ex. A32 = address is 32)
Baud [B] : B12 = 1.2K, B24 = 2.4K, B48 = 4.8K,
          B96 = 9.6K, B144 = 14.4K, B192 = 19.2K,
          B288 = 28.8K, B384 = 38.4K, B576 = 57.6K,
          B768 = 76.8K, B1152 = 115.2K
Data [D] : D7 = 7-bit, D8 = 8-bit
Parity [P] : PM = none, PE = even, PO = odd
Stop [S] : S1 = 1 bit, S2 = 2 bits
Modbus [M] : MR = rtu, MA = ascii
Watchdog [W] : W0 to W65535 sec,(Ex. W10 = 10 seconds, W0 = disable)
Delay [E] : E0 to E65535 ms,(Ex. E1000 = 1 second, E0 = disable)
Read [R]
Version [V]
Help [?]
Example: >A32 B96 S1 PM <enter>

??? (press '?' to get helps)
>R
Address = 32
Baud = 9,600
Data = 8 BITS
Parity = NONE
Stop = 1 BIT
Modbus = RTU
Watchdog = 0
Delay = 0

>A32 B96 D8 PM S1 MR W0 E0
>

```

12. Type **R** and press **ENTER** to check parameters again.



```

COM1:19200baud - Tera Term VT
File Edit Setup Control Window Help
Address [A] : A1 to A247 (Ex. A32 = address is 32)
Baud [B] : B12 = 1.2K, B24 = 2.4K, B48 = 4.8K,
          B96 = 9.6K, B144 = 14.4K, B192 = 19.2K,
          B288 = 28.8K, B384 = 38.4K, B576 = 57.6K,
          B768 = 76.8K, B1152 = 115.2K
Data [D] : D7 = 7-bit, D8 = 8-bit
Parity [P] : PM = none, PE = even, PO = odd
Stop [S] : S1 = 1 bit, S2 = 2 bits
Modbus [M] : MR = rtu, MA = ascii
Watchdog [W] : W0 to W65535 sec,(Ex. W10 = 10 seconds, W0 = disable)
Delay [E] : E0 to E65535 ms,(Ex. E1000 = 1 second, E0 = disable)
Read [R]
Version [V]
Help [?]
Example: >A32 B96 S1 PM <enter>

??? (press '?' to get helps)
>R
Address = 32
Baud = 9,600
Data = 8 BITS
Parity = NONE
Stop = 1 BIT
Modbus = RTU
Watchdog = 0
Delay = 0

>A32 B96 D8 PM S1 MR W0 E0
>R
Address = 32
Baud = 9,600
Data = 8 BITS
Parity = NONE
Stop = 1 BIT
Modbus = RTU
Watchdog = 0
Delay = 0

>

```

13. Type **V** and press **ENTER** on the keyboard. The current version will be shown. 100 will be shown for version 1.00.

```

COM1:19200baud - Tera Term VT
File Edit Setup Control Window Help
      B288 = 28.8K, B384 = 38.4K, B576 = 57.6K,
      B768 = 76.8K, B1152 = 115.2K
Data   [D] : D7 = 7-bit, D8 = 8-bit
Parity [P] : PM = none, PE = even, PO = odd
Stop   [S] : S1 = 1 bit, S2 = 2 bits
Modbus [M] : MR = rtu, MA = ascii
Watchdog [W] : W0 to W65535 sec,(Ex. W10 = 10 seconds, W0 = disable)
Delay  [E] : E0 to E65535 ms,(Ex. E1000 = 1 second, E0 = disable)
Read   [R]
Version [V]
Help   [?]
Example: >A32 B96 S1 PM <enter>

??? (press '?' to get helps)
>R
Address = 32
Baud    = 9,600
Data    = 8 BITS
Parity  = NONE
Stop    = 1 BIT
Modbus  = RTU
Watchdog = 0
Delay   = 0

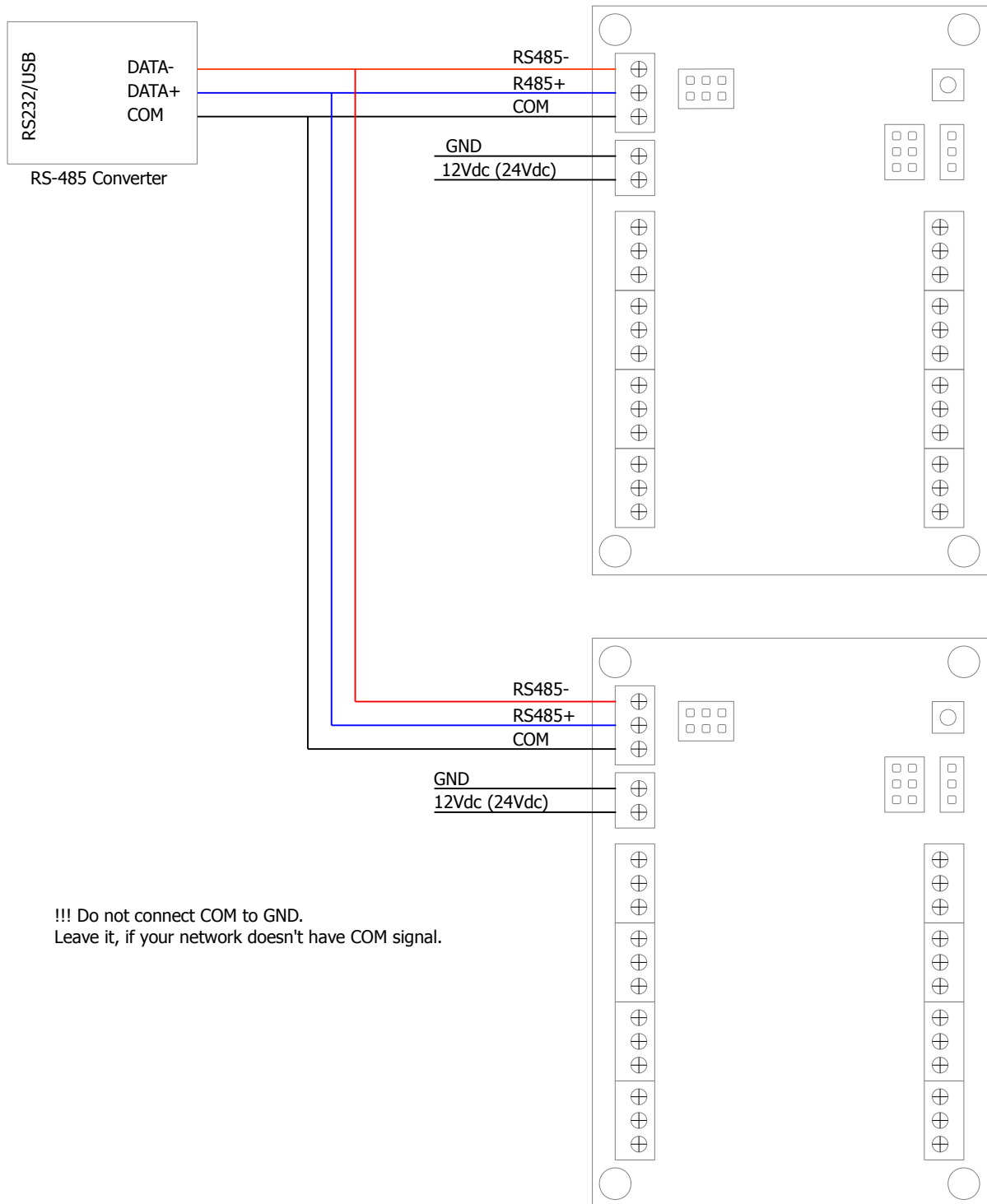
>A32 B96 D8 PM S1 MR W0 E0
>R
Address = 32
Baud    = 9,600
Data    = 8 BITS
Parity  = NONE
Stop    = 1 BIT
Modbus  = RTU
Watchdog = 0
Delay   = 0

>V
100
>

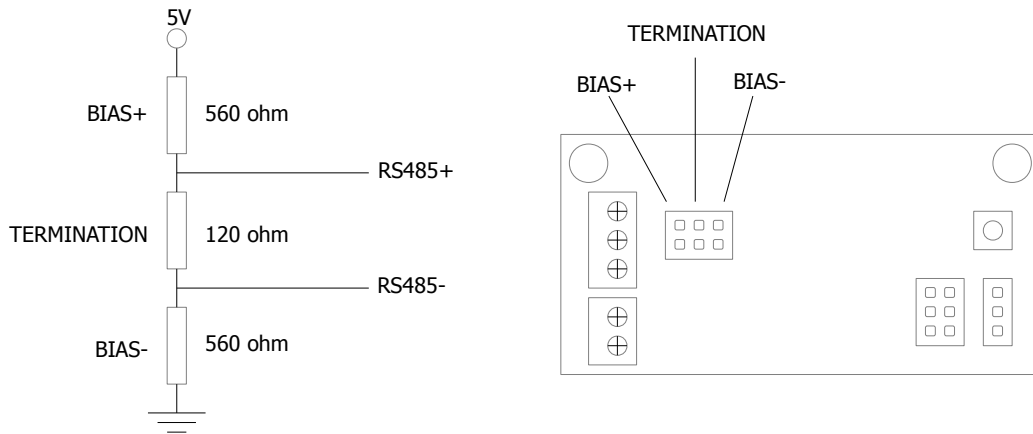
```

- Remove the jumper and press the **RESET** button to run the board in normal mode.

#### RS-485 Connection



**Bias Resistors**



**Bias Setting**

Enable BIAS+	Enable BIAS-	Enable Termination	Enable BIAS+, BIAS-	Enable All

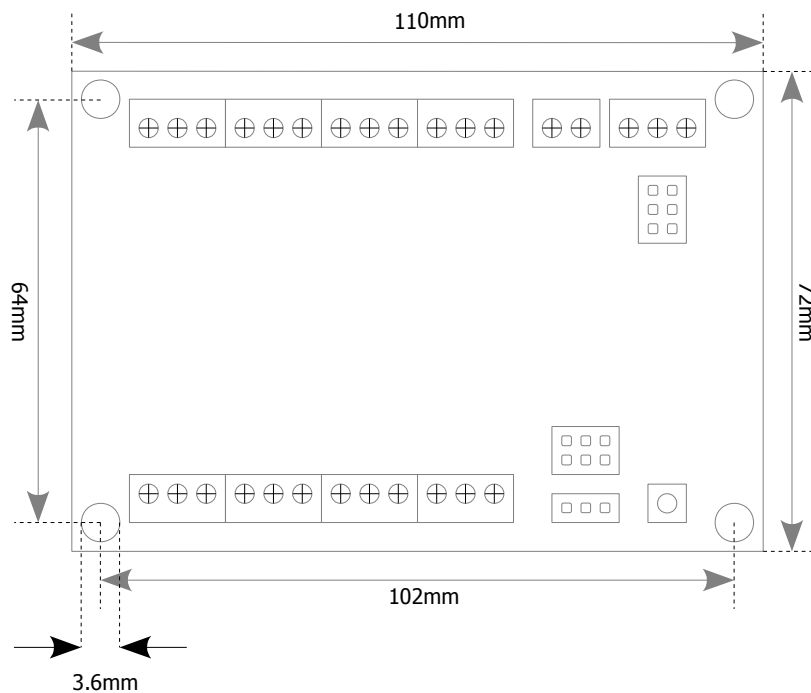
**Specifications**

	RTU-RL812M	RTU-RL824M
Power Supply	12Vdc	24Vdc
Relay Contact Rating	10A @120Vac/24Vdc 6A @250Vac	10A @120Vac/24Vdc 6A @250Vac
Max. Node	32	32

**System Requirements**

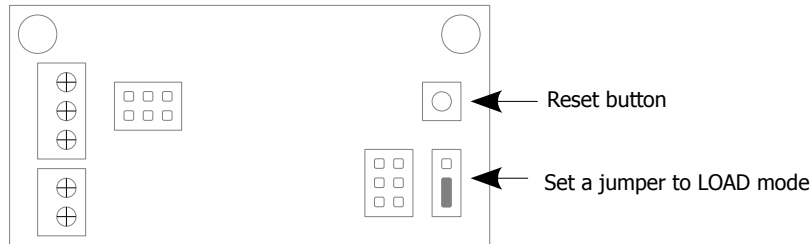
- Computer
- RS-485 Converter
- Power Supply (12Vdc, 24Vdc)

**Dimensions**

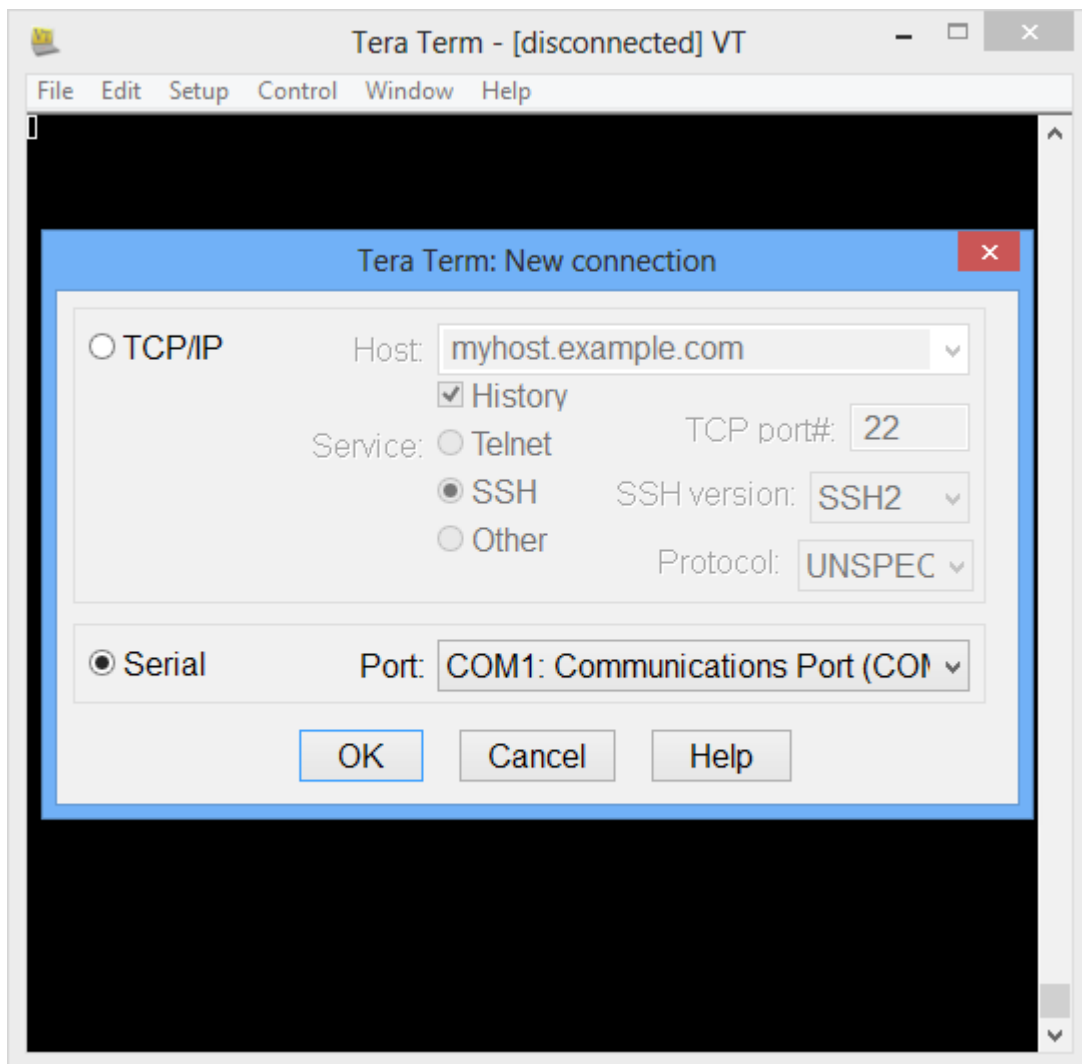


### Upgrading Firmware

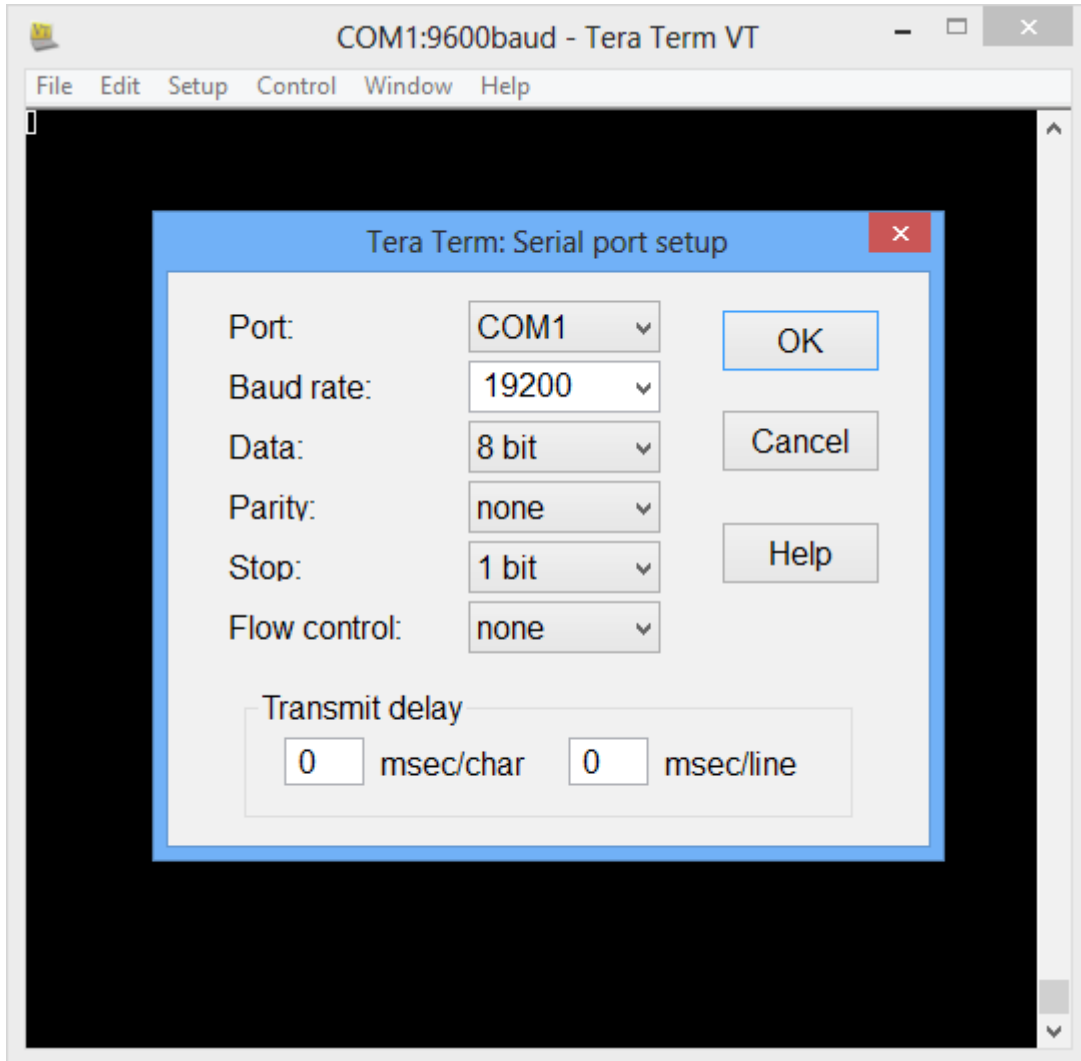
1. Connect the board to RS-485 converter(with auto direction control).
2. Connect power supply.
3. Set a jumper to **LOAD**.



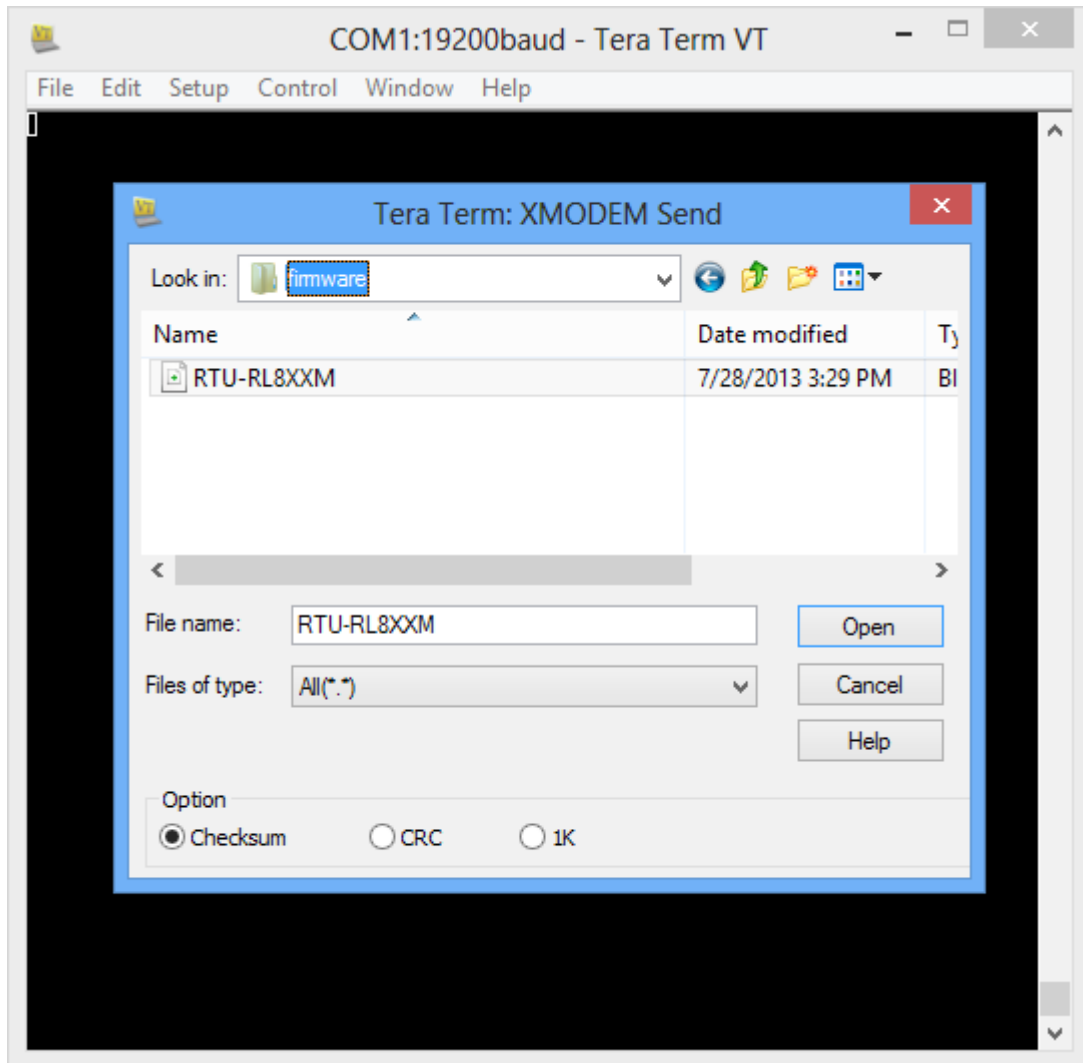
4. Run **Tera Term** program (or other serial terminal program which can send file with XMODEM).



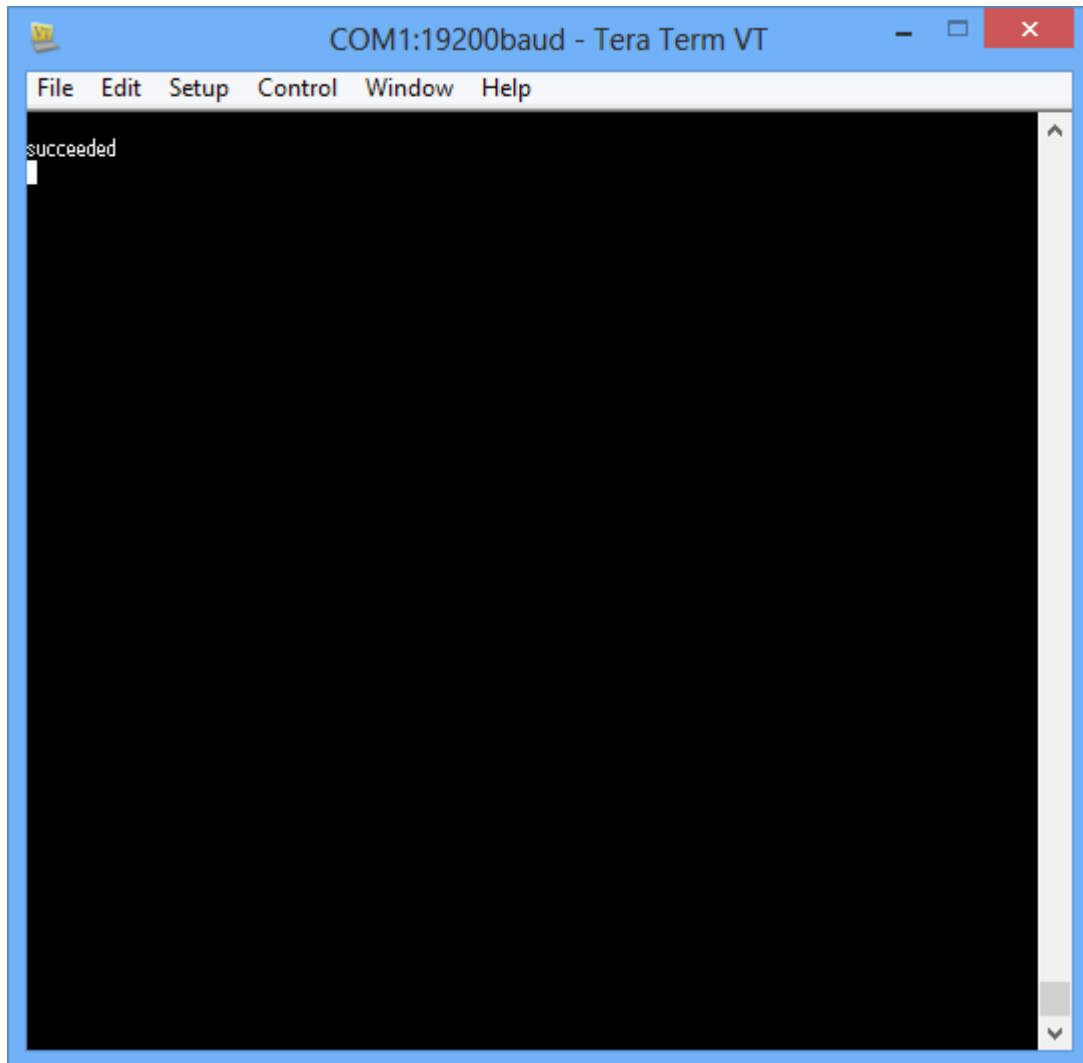
5. Select **Serial** and then click **OK** button.
6. Setup serial port from **Setup** menu.
7. Set Comport, Baud = 19200, Data = 8, Parity = None, Stop = 1, Flow control = NONE and then click **OK** button.



8. Go to menu bar and select **File->Transfer->XMODEM->Send...**



9. Open firmware file **xxx.bin** and then click **Open** button by selecting **Checksum** on the Option.
10. Press **RESET** button on the board the LED will blink and wait until the screen shows succeeded.



11. Remove the jumper and press the **RESET** button to run program in normal mode.