

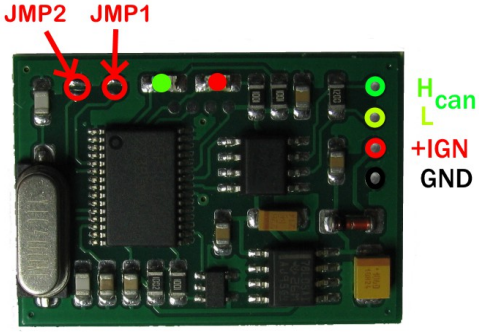
ZOTYE

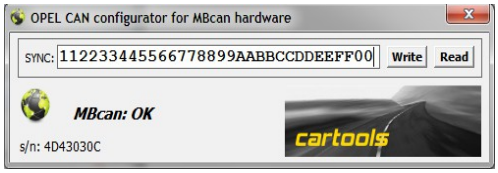
immo emulator, CAN, 500 kB

Zotye T300, T500, T600, Z500, Z560 etc: 2015 up

Usage:

Designed for ZOTYE cars, 2015 up, authorization via CAN. Plug&Play for MT62 ecu. Or must store MAC and SYNC codes manually using any CAN logger or **MBcan** (OPEL CAN cfg software used for MBcan).





LEDs on emulator board – GRN and **RED**:

- both shortly ON at emulator power up,
- **RED** on: authorization FAIL, blinks if ERROR.
- **GRN** on: auth OK, blinks if CFG frame received.

Self – teaching (MT62 ECU only!):

- attach emulator to CAN, IGN ("Kl.15"), GND wires. You can use OBD connector (permanent +BAT power supply from OBD pin16 while teaching occurs). PT-CAN is wired directly to OBD pins6/14 too.
- Both **JMP** on emulator board must be OPEN to allow SYNC update (configuration mode).
- Switch IGN ON, ECU must send immo request. If response from emulator is not accepted (what will 100% occur at first try) **RED** on emulator board will blink for 2 seconds.
- Wait 10 seconds, **RED** and **GRN** on emulator board must go on shortly (self teaching starts). If there is no any LED activity it is possibly caused by:
  - no IMMO request from ECU, emulator is silent and nothing happens.
  - ECU is already authorized (MAC / SYNC in emulator is already ok, previously **GRN** at IGN ON).
- After teaching procedure ECU is restarted and makes another authorization try. Observe LED -
  - if **GRN** is on for 2 seconds – everything is most likely OK,
  - if **RED** stays ON or blinks – something went wrong, must make another try or store data manually!
- If success, you may try to start car.
- Finally place solder joint(s) to short any or both JMP! Emulator is ready for installation now.

Manual configuration:

- Both JMP must be OPEN for configuration.
- Power emulator from 12v power source, attach MBcan / CAN logger hardware, launch software.
- **Using MBcan:** enter and store 4 bytes MAC, 8 bytes SYNC followed by 4x 00's in row. **GRN** must blink for 2 seconds, data is accepted and stored.
- **Using logger:** to store data must send 3 CAN frames, ID = 0x7FE, DLC = 8, 11-bit identifier. 1st DATA byte is a sequence number (00, 01, 02), remaining 7 bytes are MAC & SYNC data. Example:

```
0x7FE 8 00 11 22 33 44 55 66 77
0x7FE 8 01 88 99 AA BB CC 00 00
0x7FE 8 02 00 00 00 00 00 00 00
```

  means you are storing MAC **11223344** and SYNC **5566778899AABBCC** here.
- Response from emulator looks similar, must see same data with ID 0x7FF -

```
0x7FF 8 00 11 22 33 44 55 66 77
0x7FF 8 01 88 99 AA BB CC 00 00
0x7FF 8 02 00 00 00 00 00 00 00
```

  If data match, SYNC is stored successfully.
- You can request actual SYNC from emulator (NOTE: if it is stored manually before!), must send -

```
0x7FE 8 FF 00 00 00 00 00 00 00
```

  Emulator will respond as above, 3 frames with SYNC.
- Finally place solder joint(s) to short any or both JMP to disable self teaching AND / OR further SYNC updates by accident.

(c) RobinDAB '2024

cartools