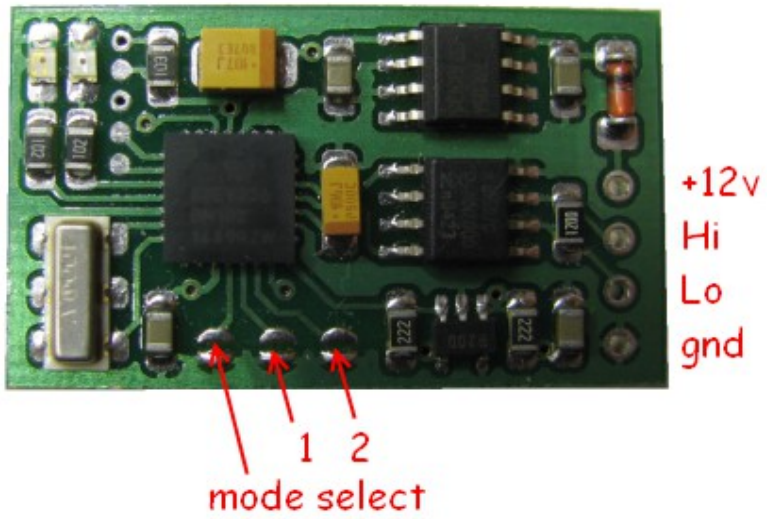

MB FBS3 emulator

for CAN HS 500kB

Designed to replace original immobox on **FBS3** systems (based on hashes) for drive authorization. Basic principles are very similar to FBS3 IR KEY operation: must store initial hash, track number and distance into emulator.

How to configure:

Solder joint for MODE selection must be removed (open) to enter configuration mode. Must use any suitable CAN bus logger. Communication speed is 500 kb.

Abbreviations used:

-> transfer to emulator
<- response from emulator
YLED means yellow led, RLED means red led, both located on emulator board (upper left corner).

Step 1: must transfer initial hash
example value = 14C2DAE130F81AC8

```
-> 7F0 (8) 14 C2 DA E1 30 F8 1A C8
<- 7FF (8) 14 C2 DA E1 30 F8 1A C8
```

YLED is blinking if hash accepted / received.

Step 2: must transfer track number and distance
example: track = **3**, distance = **2F1F7F** (full key track here, although it is possible to configure emulator with any valid distance value from 000001 up to 2F1F7F)

```
-> 7F1 (8) 03 2F 1F 7F 00 00 00 00
```

1st part of response (immediately):

```
<- 700 (4) 03 2F 1F 7F
```

If everything is okay and initial hash was received previously in current power-on cycle, YLED goes off, RLED is blinking. Emulator is preparing buffers now. Usually it takes about 20 seconds to complete task. When finished, emulator responds with calculated final hash for distance 2F1F7F (always for full track length, no matter what distance was actually requested):

2nd part of response (buffers already built, emulator is ready)

```
<- 7FF (8) D8 A5 57 F3 F8 FD 87 36
```

YLED goes permanently ON, emulator is configured and ready now.

Normal operational mode:

Solder joint **shorted**, emulator connected to ECU via CAN.

After power-up YLED will blink, emulator is ready to communicate with ECU. After successfull authorization with ECU YLED goes permanently ON. Emulator is preparing buffers for next power up now: RLED goes on, when prepared goes OFF. It can take up to 1/2 of second in worst case for recalculation.

If after authorization RLED and YLED both are **flashing simultaneously**, authorization is already complete, but remaining distance (life) is going low (less than 4096, at distance 00xxxx).

Error state:

If immediately after power-up RLED and YLED both are **flashing alternately**, emulator isn't ready to authorize ECU. Some possible causes:
- out of life - distance is 000000.
- emulator isn't configured (blank)

Additional commands for enthusiasts:

(configuration mode only! Solder joint MODE removed / open)

Erase ALL:

```
-> 7F1 (8) 55 00 00 00 00 00 00 00
```

RLED goes on until erase is complete. Success is confirmed with message:

```
<- 7FF (6) 45 52 41 53 45 44
```
(ascii: ERASED)

Current HASH request, decrease life by 1:

```
-> 7F1 (8) AA 00 00 00 00 00 00 00
```

response, 2 frames:

```
<- 700 (4) KK DD DD DD
```
, where KK = track number, DDDDDD = current distance

```
<- 7FF (8) 8xDATA
```
, where DATA = current hash

Request data blocks (EEPROM):

```
-> 7F1 (8) FF xx 00 00 00 00 00 00
```
, where **xx** = data block number, 00...FF

response:

```
<- 7xx (8) 8xDATA
```