Volvo Motronic M4.4

A detailed description for Motronic 4.4 ECU's used in Volvo 850 TS(R) and S70 TS(R)
Table of content

Hardware........................................................................................................................................... 4
  Overview of the board......................................................................................................................... 4
  Main CPU: SAB83C517A-5N18......................................................................................................... 5
  Flash eprom: AN28F010..................................................................................................................... 6
  ECU Pinout .......................................................................................................................................... 7
  M4.4 specific implementations ......................................................................................................... 9

Software............................................................................................................................................... 10
  Software conversion factors .............................................................................................................. 10
  Software information data ................................................................................................................ 10
  Reading the code .............................................................................................................................. 11

Maps and variables.............................................................................................................................. 12
  Getting the map lookup table index .............................................................................................. 12
  Reading the map lookup table ......................................................................................................... 12
  How Motronic calculates theoretical engine load .......................................................................... 12

Communication with the ECU .......................................................................................................... 14
  Connection diagram ....................................................................................................................... 14
  Normal mode communication ......................................................................................................... 14
  Wakeup procedure for normal mode ............................................................................................. 14
  Normal mode: KWP71 ................................................................................................................... 14

Bootmode communication .................................................................................................................. 15
  Erase command ............................................................................................................................ 15
  Wait for erase completed .............................................................................................................. 15
  Send data for lower flash bank ..................................................................................................... 16
  Switch to upper flash bank .......................................................................................................... 16
  Send data for upper flash bank ..................................................................................................... 16
  Acknowledge message (flash completed) ..................................................................................... 16
  Checksum verification message .................................................................................................... 16
  Checksum answer message .......................................................................................................... 16
Introduction

This document describes the Motronic M4.4 ECU in detail. It will first describe the hardware and proceed with a even more detailed description of the software that is running in the ECU so that we can learn how to tweak and tune the ECU to match the hardware – altered or not – that is on the car better.

*Special thanks for getting all this together go out to rkam, T5_Germany, Steve Hayes and others on ecurproject.com, trionictuning.com and volvospeed.com.*
Hardware

Overview of the board
The ECU contains a tri-layer printed circuit board (PCB) which holds a lot of SMD components. The main components are – logically: Main CPU, Flash program storage, SRAM memory (working memory) and a lot of input/output (I/O).
Main CPU: SAB83C517A-5N18

- Upto 18Mhz operation
- 32K x 8 ROM
- 256 x 8 on-chip RAM
- 2K x 8 on-chip RAM (XRAM)
- Four 16 bit timers/counters
- 10 bit A/D converter with 12 multiplexed inputs
- Two full duplex serial interfaces
- Nine ports: 56 i/O lines, 12 input lines
- Mask programmable ROM (Internal ROM protection)

Datasheet documents


http://trionic.mobixs.eu/Motronic/M4.4/80c517um.pdf

**Flash eprom: AN28F010**

This is an automotive specified 128Kb flash chip with a temperature range from -40 degrees up to 125 degrees Celsius.

- Programming voltage: 12V
- Chip erasure time: 1 second
- Byte program time: 10 uS
- Access time: 120 nS

Datasheet:

# ECU Pinout

<table>
<thead>
<tr>
<th>Pin number</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td></td>
<td>Signal (+) front knock sensor (KS) (measured to terminal #A17)</td>
</tr>
<tr>
<td>A2</td>
<td></td>
<td>Power ground mass air flow (MAF) sensor</td>
</tr>
<tr>
<td>A3</td>
<td></td>
<td>Signal mass air flow (MAF) sensor (measured to terminal #A5)</td>
</tr>
<tr>
<td>A4</td>
<td></td>
<td>Signal ground mass air flow (MAF) sensor</td>
</tr>
<tr>
<td>A5</td>
<td></td>
<td>Signal ground engine speed (RPM) sensor</td>
</tr>
<tr>
<td>A6</td>
<td></td>
<td>Control signal engine cooling fan (FC) low-speed</td>
</tr>
<tr>
<td>A7</td>
<td></td>
<td>Control signal injector 5</td>
</tr>
<tr>
<td>A8</td>
<td></td>
<td>Control signal injector 1</td>
</tr>
<tr>
<td>A9</td>
<td></td>
<td>Opening signal the idle air control (IAC) valve</td>
</tr>
<tr>
<td>A10</td>
<td></td>
<td>15 supply (power supply from the ignition switch) (+12V)</td>
</tr>
<tr>
<td>A11</td>
<td></td>
<td>Power ground</td>
</tr>
<tr>
<td>A12</td>
<td></td>
<td>Control signal front heated oxygen sensor (HO2S), preheating</td>
</tr>
<tr>
<td>A13</td>
<td></td>
<td>Y Power supply the throttle position (TP) sensor</td>
</tr>
<tr>
<td>A14</td>
<td></td>
<td>G/W Signal the throttle position (TP) sensor</td>
</tr>
<tr>
<td>A15</td>
<td></td>
<td>BR/B Knock sensor common ground</td>
</tr>
<tr>
<td>A16</td>
<td></td>
<td>Ground (sensor)</td>
</tr>
<tr>
<td>A17</td>
<td></td>
<td>A19 Signal (–) rear heated oxygen sensor (HO2S)</td>
</tr>
<tr>
<td>A18</td>
<td></td>
<td>Signal engine speed (RPM) sensor (measured to #A6)</td>
</tr>
<tr>
<td>A19</td>
<td></td>
<td>BL/Y Signal camshaft position (CMP) sensor</td>
</tr>
<tr>
<td>A20</td>
<td></td>
<td>A21 Control signal engine cooling fan (FC) high-speed</td>
</tr>
<tr>
<td>A21</td>
<td></td>
<td>Control signal injector 4</td>
</tr>
<tr>
<td>A22</td>
<td></td>
<td>Control signal injector 3</td>
</tr>
<tr>
<td>A23</td>
<td></td>
<td>Closing signal the idle air control (IAC) valve</td>
</tr>
<tr>
<td>A24</td>
<td></td>
<td>30– supply (power supply from the battery) (+12V)</td>
</tr>
<tr>
<td>A25</td>
<td></td>
<td>Power supply (from the main relay) (+12V)</td>
</tr>
<tr>
<td>A26</td>
<td></td>
<td>Power ground control module</td>
</tr>
<tr>
<td>A27</td>
<td></td>
<td>Control signal rear heated oxygen sensor (HO2S), preheating</td>
</tr>
<tr>
<td>A28</td>
<td></td>
<td>Y/GY Signal the engine coolant temperature (ECT) sensor</td>
</tr>
<tr>
<td>A29</td>
<td></td>
<td>A31 Signal (+) rear knock sensor (KS) (measured to terminal #A17)</td>
</tr>
<tr>
<td>A30</td>
<td></td>
<td>A32 Signal (+) front heated oxygen sensor (HO2S)</td>
</tr>
<tr>
<td>A31</td>
<td></td>
<td>A33 Signal (–) front heated oxygen sensor (HO2S)</td>
</tr>
<tr>
<td>A32</td>
<td></td>
<td>A34 Signal (+) rear heated oxygen sensor (HO2S)</td>
</tr>
<tr>
<td>A33</td>
<td></td>
<td>A35 G/Y Outer temperature sensor</td>
</tr>
<tr>
<td>A34</td>
<td></td>
<td>Y/R Power supply camshaft position (CMP) sensor</td>
</tr>
<tr>
<td>A35</td>
<td></td>
<td>A37 Pulsed secondary air injection system (PAIR) pump valve, control signal</td>
</tr>
<tr>
<td>A36</td>
<td></td>
<td>A38 Control signal injector 2</td>
</tr>
<tr>
<td>A37</td>
<td></td>
<td>A39 Control signal Canister purge (CP) valve</td>
</tr>
<tr>
<td>A38</td>
<td></td>
<td>A40 Control signal main relay</td>
</tr>
<tr>
<td>A39</td>
<td></td>
<td>A41 Signal ground control module (measured to the battery negative terminal)</td>
</tr>
<tr>
<td>A40</td>
<td></td>
<td>A42</td>
</tr>
<tr>
<td>Pin number</td>
<td>Color</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>B1</td>
<td></td>
<td>Power supply Accelerometer (vehicle speed)</td>
</tr>
<tr>
<td>B2</td>
<td></td>
<td>Signal torque limiting (from automatic gearbox)</td>
</tr>
<tr>
<td>B3</td>
<td></td>
<td>Signal torque limiting (from automatic gearbox)</td>
</tr>
<tr>
<td>B4</td>
<td></td>
<td>Signal torque limiting acknowledgement (to automatic gearbox)</td>
</tr>
<tr>
<td>B5</td>
<td></td>
<td>To diagnostics socket (also to Aut-transmission control module)</td>
</tr>
<tr>
<td>B6</td>
<td></td>
<td>Signal air conditioning (A/C) compressor status</td>
</tr>
<tr>
<td>B7</td>
<td>GY</td>
<td>Control signal malfunction indicator lamp (MIL) (to the combined instrument panel)</td>
</tr>
<tr>
<td>B8</td>
<td></td>
<td>Enable internal ECU ROM flashing when +12V is applied (results in +5V on EA pin on CPU)</td>
</tr>
<tr>
<td>B9</td>
<td></td>
<td>Signal A/C pressure sensor</td>
</tr>
<tr>
<td>B10</td>
<td></td>
<td>Control signal ignition discharge module (IDM)</td>
</tr>
<tr>
<td>B11</td>
<td></td>
<td>Signal load Tq (to automatic gearbox)</td>
</tr>
<tr>
<td>B12</td>
<td></td>
<td>Power supply fuel tank pressure sensor</td>
</tr>
<tr>
<td>B13</td>
<td></td>
<td>Signal speed (from combined instrument panel)</td>
</tr>
<tr>
<td>B14</td>
<td></td>
<td>Control signal fuel pump</td>
</tr>
<tr>
<td>B15</td>
<td></td>
<td>Signal throttle position (TP) sensor (to automatic gearbox)</td>
</tr>
<tr>
<td>B16</td>
<td></td>
<td>Signal tachometer (to combined instrument panel)</td>
</tr>
<tr>
<td>B17</td>
<td></td>
<td>Atmospheric pressure sensor</td>
</tr>
<tr>
<td>B18</td>
<td></td>
<td>Signal engine coolant temperature (ECT) (to ECC and combined instrument panel)</td>
</tr>
<tr>
<td>B19</td>
<td></td>
<td>Signal constant idle speed compensation P/N position (from automatic gearbox)</td>
</tr>
<tr>
<td>B20</td>
<td>BL/GY</td>
<td>Signal air conditioning (A/C) relay status</td>
</tr>
<tr>
<td>B21</td>
<td></td>
<td>Signal malfunction indicator lamp (MIL) request (from automatic gearbox)</td>
</tr>
<tr>
<td>B22</td>
<td></td>
<td>Control signal fuel pump</td>
</tr>
<tr>
<td>B23</td>
<td></td>
<td>Signal ground sensor (measured to the battery negative terminal)</td>
</tr>
<tr>
<td>B24</td>
<td></td>
<td>Power supply A/C pressure sensor</td>
</tr>
<tr>
<td>B25</td>
<td></td>
<td>Fuel tank pressure sensor (certain markets only)</td>
</tr>
<tr>
<td>B26</td>
<td></td>
<td>Signal Accelerometer (vehicle speed)</td>
</tr>
<tr>
<td>B30</td>
<td></td>
<td>Diagnostic lead K–link</td>
</tr>
<tr>
<td>B31</td>
<td></td>
<td>Control signal pulsed secondary air injection system (PAIR) pump relay</td>
</tr>
<tr>
<td>B32</td>
<td></td>
<td>Signal fuel consumption (to the trip computer)</td>
</tr>
<tr>
<td>B33</td>
<td></td>
<td>Control signal air conditioning (A/C) relay (allows A/C to start)</td>
</tr>
<tr>
<td>B34</td>
<td></td>
<td>Control signal, turbocharger (TC) control valve</td>
</tr>
<tr>
<td>B35</td>
<td></td>
<td>Turbocharger (TC) boost pressure limiting signal (from automatic gearbox)</td>
</tr>
</tbody>
</table>
M4.4 specific implementations

Since M4.4 has twice the flash size compared to M4.3 so it has 128Kb of ROM to work with and the CPU only has 16 address lines, Bosch needed to figure out a way to address the upper flash bank (ranging 0x10000-0x1FFFF) that needs the 17th address line. This is done by using a general purpose IO pin for switching the banks. P5.7 (port 5 highest bit) was chosen for this task. P5.7 high means the upper flash bank is selected, P5.7 low means the lower flash bank is selected. The latter being default.

P8.2 is the 10th analogue input on the CPU and this one is used in the bootloader code as well.
Software

Once we download the data from the ECU – e.g. with a flash programmer – we can load the binary file into MotronicSuite (ref: http://trionic.mobixs.eu/Motronic/Motronic.msi)

Details on how the software works can be found in appendix I.

Software conversion factors

The software contains a lot of information for which we have to know the conversion factors to use to be able to translate them into units that we use on a daily basis. This is a shortlist of the most used factors.

- 36 Battery voltage * 0.0704 Volt
- 38 Engine Coolant Temperature (ECT) * 1 - 80 < - 50 degrees Celsius
- 3B Engine speed * 30 RPM
- 40 Internal load signal * 0.05 msec
- 4B Mass Air Flow (MAF) sensor signal * 0.0196 Volt
- 54 Ignition advance * (- 0.75) + 78 Degrees
- BC Turbo duty cycle * 0.391 (uncertain)

Complete document here: http://trionic.mobixs.eu/Motronic/M4.4/M44-scaling-RAM.pdf

Software information data

In the software, the identifiers are stored as well about HW revision, SW version, volvo partnumbers etc. This data is stored in ASCII in the binary file and looks something like this:

000016f0h: 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 ; 000000000000000
00001700h: 40 40 40 30 32 36 31 32 30 34 32 33 39 31 30 33 ; 0000261204239103
00001710h: 37 33 35 35 37 38 30 31 32 37 30 34 32 32 FF FF ; 73557801270422yy
00001720h: FF 2E 33 31 32 37 30 34 32 2D 2D 30 32 ; yy.331270422--002
00001730h: 7D 7D 7D 4B 00 00 00 00 00 00 00 00 ; )})e.............

1270422 = P01270422
0261204239 is the hardware ID
1037355780 is the software ID
331270422—002 is some partnumber ID

And there’s more readable information in the file as well:

0000ff00h: CC 74 02 02 02 02 02 02 02 02 02 02 02 02 02 ; Ìt.................
0000ff10h: 34 37 2F 31 2F 4D 34 2E 34 2F 31 39 2F 31 34 ; 47/1/M4.4/19/114
0000ff20h: 2E 33 33 2F 44 41 4D 4F 53 33 38 2F 33 38 30 32 ; .33/DAMOS38/3802
0000ff30h: 55 2F 41 55 33 31 32 2F 32 36 30 33 39 36 2F FF ; U/UA312/260396/y
0000ff40h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ; yyyyyyyyyyyyyyyy
**Reading the code**

To be able to understand the software better we’ll need to dive into the world of assembler language. This is a sort of intermediate between understandable human language and the operation codes used by the microprocessor. Once we can read the assembler language (assembly for short) we can track all the things the microprocessor is told to do when the program is running. This is very valuable information because we don’t have first hand information from either Bosch or Volvo that can tell us in details what the ECU does.

We convert the binary file into assembly language we need to disassemble the file. We can do that by running the disassembler in Motronic Suite, by running the disassembler manually or by using a separate program like IDAPro to do it for us. A separate disassembler can be found here in the website.

Disassembler D52 [http://trionic.mobixs.eu/Motronic/M4.3/d52.exe](http://trionic.mobixs.eu/Motronic/M4.3/d52.exe)

Once we disassemble the binary file we have an file containing the complete assembly listing in which we can start to explore and understand the internal workings of M4.4.
Maps and variables

Determining the location and type of maps and variables in the M4.4 binaries is quite a hassle. In contrast to M4.3 the map addresses are present in the file but they are not linked directly to the axis (support point) in the file. To be able to detect the available maps we have to do some tricks and make a couple of assumptions in the algorithm used. This chapter will describe – in detail – what the Motronic Suite software does to fetch the maplist from the file.

Getting the map lookup table index

First we need to lookup the index in the file at which the lookup table is located. This lookup table consists of addresses in the file which we can use to determine axis and map information. Do find the correct index we look for a certain byte sequence in the file. M4.3 has a leading sequence of 4 bytes that always seem to be the same, 0x00 0x02 0x04 0x00 0x02 0x00. The picture below shows the data found in a certain M4.4 file.

Reading the map lookup table

If we find a 0x00 directly after this sequence, we need to add 7 to the address found, so reading of map locations in the above file will commence at address 0x000D51, 2 bytes at a time. So, the first address we find is 0x1CF3 and the second one is 0x1D06. We keep reading addresses and storing them in a list until we reach a 2 byte value that is smaller than 0x1000 (so, in fact we assume that there are no maps located in the memory section < 0x1000).

How Motronic calculates theoretical engine load

Motronic needs three things to calculate the internal Load signal (which can be found as axis for several maps):

1. A signal from the airmass meter, normalized to airflow in kg/hr: \( Q \)
2. The current engine speed (rpm): \( n \)
3. The programmed injector constant: \( K_i \)

\[
Q = f\left(\frac{U_p}{U_v}\right)
\]
in which $\frac{u_P}{u_v}$ is the ratio between MAF output and MAF reference voltages.

$$TL = \frac{Q}{n \times Ki}$$

$TL$ (LOAD) is not just a representation of cylinder filling, but the theoretical Injector Time Open ($Ti$) needed to reach stoich (Lambda= 1) with the current injector setup assuming that the motor has an efficiency of 100% (VE), which it has not of course. Hence there are fueling tables which are used as multiplicative corrections to $TL$ to reach the actual $Ti$.

With $TL$ quantified, Motronic now takes into account the correction factors for the engine and the current operating conditions by introducing multiplicative factors to correct the THEORETICAL injector time to the ACTUAL time for injection ($Ti$) needed at that operating condition point. Finally an additive factor ($Tv$) is added to compensate for the fluctuating injector opening time under lower than nominal voltages (battery correction map).

$$Ti = (TL \times [X, Y, Z ...]) + Tv$$

The final $Ti$ is the injector open time that is applied to the injectors.

Informational credits to: Jim Conforti
Communication with the ECU

Connection diagram

There are two methods of communication that can be used with a M4.4 ECU.
- Normal mode communication
- Boot mode communication

Normal mode communication

Normal mode is used for reading data from the ECU while it is in its operational state. Reading live data and the contents of the flash file are procedures that are carried out in normal mode. To activate normal mode communication we need to connect a K-line interface to the ECU on pin B36 and after the wake-up procedure communication can commence at 12700 baud.

Wakeup procedure for normal mode

To be able to communicate in normal mode, the ECU needs to be aware of the fact that there is a diagnostics device connected to pin B5. To let the ECU know we need to send a 0x10 byte to the port at 5 baud (!). After a correct wakeup byte on the B5 pin we will receive a response from the ECU at 12700 baud. This response will be 0x55 0xAB 0x02 in which the 0x55 is the acknowledge and the 0xAB and 0x02 are the keywords used to communicate with a M4.3 ECU. After reception of this sequence we need to send an acknowledge message to the ECU which is the inverted last keyword which will be 0xFD.

Normal mode: KWP71

After the wakeup procedure, communication with the ECU takes place in the KWP71 protocol. This protocol is standarized and therefore it will not be discussed within this document.
Bootmode communication

Boot mode communication is only useful for flashing the ECU with a new firmware version. This is a special mode which is indicated to the ECU by pulling pin B8 to +12V before the ECU starts (boots). So, you will need to have the ECU powered down and apply +12V to B8 before the ECU is powered on. The ECU will now run a special boot loader program which resides in internal ROM and allows us to reprogram the ECU. After starting in bootmode communication can commence on pin B36 at 9600 baud with a K-line interface.

The M4.4 bootloader is quite a bit bigger than the M4.3 bootloader. There is additional code for (probably) hardware testing purposes or KWP communication in the bootloader as well. It also checks more stuff than the M4.3 bootloader and it has additional code for switching banks since M4.3 does not have that.

Bootmode is forced by applying +12V on pin B8 and commencing communication on pin B36 at 9600 baud. Pin B5 is not used in M4.4.

The programming sequence for M4.4 is as follows:

- Send erase command
- Wait for erase completed
- Send data to flash from 0x000000-0x00FFFF
- Send command to switch to upper flashbank
- Send data to flash from 0x010000-0x01FFFF
- Wait for acknowledge message
- Send checksum verification message (optional)
- Wait for checksum OK or FAIL message (optional)

Erase command

3AFF5443221100000000000020000000000000000000000000000000000000000000000000

FF as length results in FRAME_TYPE 0x55 (does not get programmed)

Wait for erase completed

02305930390363 START_ERASE
02315930390362 START_ERASE
02315930390362 START_ERASE
0230593035036F FINISHED_ERASE
0231593035036E FINISHED_ERASE
0231593035036E FINISHED_ERASE
Send data for lower flash bank

3A20000000022EC1020A3B020202020A8802020202020AA70202E
...
3A20FFE000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

3A20AHAL00XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXCS
AH = Address high
AL = Address low
XX = Data to program

Switch to upper flash bank

3A20000002000000000000000000000000000000000000000000DE

Send data for upper flash bank

3A20000000022EC1020A3B020202020A8802020202020AA70202E
...
3A20FFE000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

3A20AHAL00XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXCS
AH = Address high
AL = Address low
XX = Data to program

Acknowledge message (flash completed)

023059303603XX     FLASH COMPLETED
023159303603XX     FLASH COMPLETED
023159303603XX     FLASH COMPLETED

Checksum verification message

3AFE55432211000000000300000000XXXX00000000000000000000000000CC

XXXX = checksum calculated by host
CC = message checksum

Checksum answer message

023059304503XX     CHECKSUM OK
023159304503XX     CHECKSUM OK
023159304503XX     CHECKSUM OK
OR
023059304403XX     CHECKSUM FAILED
023159304403XX     CHECKSUM FAILED
023159304403XX     CHECKSUM FAILED