

# AVR-CAN development board

## Users Manual

Rev.A, January 2010  
Copyright(c) 2011, OLIMEX Ltd, All rights reserved

## INTRODUCTION:

AVR-CAN is inexpensive way to learn AT90CAN128 microcontroller from Atmel. It's in very compact form 60x55mm and the presence of both CAN and RS232 allows the customer to make a bridge between UART and CAN networks. All AVR pins are available on extension connectors and the board can be easily a snap-on prototype or other mother board where additional circuits are built.

## BOARD FEATURES:

- MCU: AT90CAN128 - 128Kb Flash, 4Kb EEPROM, 4Kb SRAM, 52 GPIO, 32 working registers, a CAN controller, Real Time Clock, four Timer/Counters with PWM, 2 UARTs, Two-wire Serial Interface, an 8-channel 10-bit ADC, Watchdog Timer, SPI serial port, JTAG interface, five selectable power saving modes
- 5x2 pin JTAG connector for programming and debugging with AVR-JTAG-L or AVR-JTAG-USB
- RS232 DB9 female connector and driver
- CAN controller and interface
- User button
- Status LED
- 16 MHz oscillator circuit
- 32768 Hz oscillator circuit
- Reset button and circuit
- On board voltage regulator (+5V)
- Power supply LED
- Power supply connector
- Extension pin headers for every uC pin
- FR-4, 1.5 mm (0.062"), soldermask, white silkscreen component print
- Dimension 60x50 mm (2.36x1.96")

## ELECTROSTATIC WARNING:

The AVR-CAN board is shipped in protective anti-static packaging. The board must not be subject to high electrostatic potentials. General practice for working with static sensitive devices should be applied when working with this board.

## BOARD USE REQUIREMENTS:

- Cables:** RS232 cable in case you want to use the RS232 and/or CAN interface. You might need other cables depending on the programming/debugging tool you use.
- Hardware:** Programmer/Debugger - **AVR-JTAG-L** or **AVR-JTAG-USB** or other compatible programming/debugging tool.
- Software:** **AVR Studio 4.13** (or later) + **WinAVR** (latest version) or IAR AVR. AVR Studio and WinAVR are free to download and use. Take a note that AVR Studio 4.13 has a bug and doesn't read the fuses correctly. On our request to Atmel support they had confirmed the bug and

had suggested to fix the bug by “AVR Studio 4.13 SP2 JTAGICE Fix” available from [http://www.atmel.no/beta\\_ware/](http://www.atmel.no/beta_ware/)

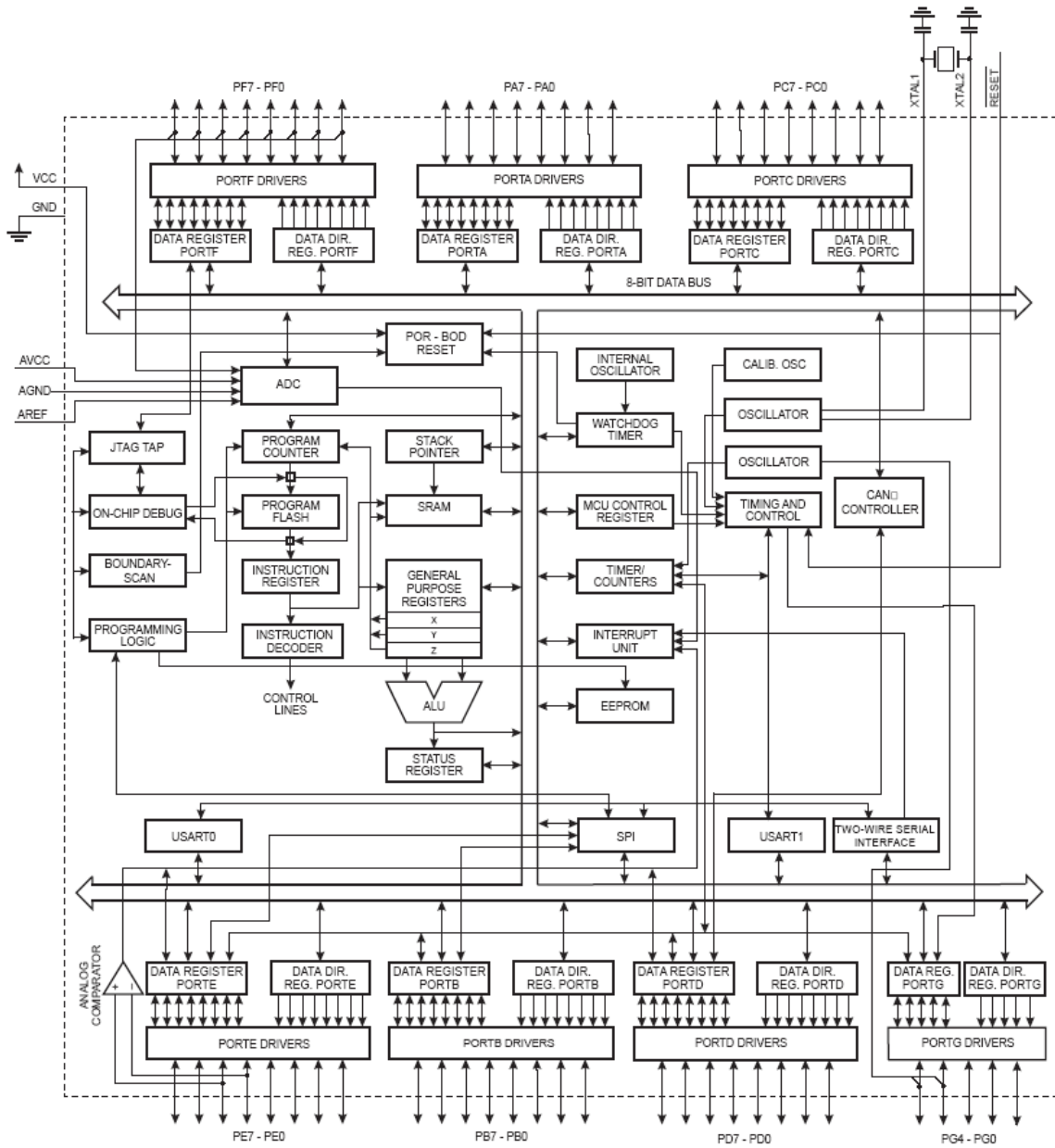
## **PROCESSOR FEATURES:**

AVR-CAN board use MCU AT90CAN128 from Atmel with these features:

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 133 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers + Peripheral Control Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Non volatile Program and Data Memories
  - 128K Bytes of In-System Reprogrammable Flash
    - Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    - Selectable Boot Size: 1K Bytes, 2K Bytes, 4K Bytes or 8K Bytes
    - In-System Programming by On-Chip Boot Program (CAN, UART, ...)
    - True Read-While-Write Operation
  - 4K Bytes EEPROM (Endurance: 100,000 Write/Erase Cycles)
  - 4K Bytes Internal SRAM (AT90CAN32/64/128)
  - Up to 64K Bytes Optional External Memory Space
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Programming Flash (Hardware ISP), EEPROM, Lock & Fuse Bits
  - Extensive On-chip Debug Support
- CAN Controller 2.0A & 2.0B - ISO 16845 Certified
  - 15 Full Message Objects with Separate Identifier Tags and Masks
  - Transmit, Receive, Automatic Reply and Frame Buffer Receive Modes
  - 1Mbits/s Maximum Transfer Rate at 8 MHz
  - Time stamping, TTC & Listening Mode (Spying or Autobaud)
- Peripheral Features
  - Programmable Watchdog Timer with On-chip Oscillator
  - 8-bit Synchronous Timer/Counter-0
    - 10-bit Prescaler
    - External Event Counter
    - Output Compare or 8-bit PWM Output
  - 8-bit Asynchronous Timer/Counter-2
    - 10-bit Prescaler
    - External Event Counter
    - Output Compare or 8-Bit PWM Output
    - 32Khz Oscillator for RTC Operation
  - Dual 16-bit Synchronous Timer/Counters-1 & 3
    - 10-bit Prescaler
    - Input Capture with Noise Canceler
    - External Event Counter
    - 3-Output Compare or 16-Bit PWM Output

- Output Compare Modulation
- 8-channel, 10-bit SAR ADC
  - 8 Single-ended Channels
  - 7 Differential Channels
  - 2 Differential Channels With Programmable Gain at 1x, 10x, or 200x
- On-chip Analog Comparator
- Byte-oriented Two-wire Serial Interface
- Dual Programmable Serial USART
- Master/Slave SPI Serial Interface
  - Programming Flash (Hardware ISP)
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - 8 External Interrupt Sources
  - 5 Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down & Standby
  - Software Selectable Clock Frequency
  - Global Pull-up Disable
- I/O and Packages
  - 53 Programmable I/O Lines
  - 64-lead TQFP and 64-lead QFN
- Operating Voltages: 2.7 - 5.5V
- Operating temperature: Industrial (-40°C to +85°C)
- Maximum Frequency: 8 MHz at 2.7V, 16 MHz at 4.5V

# BLOCK DIAGRAM:

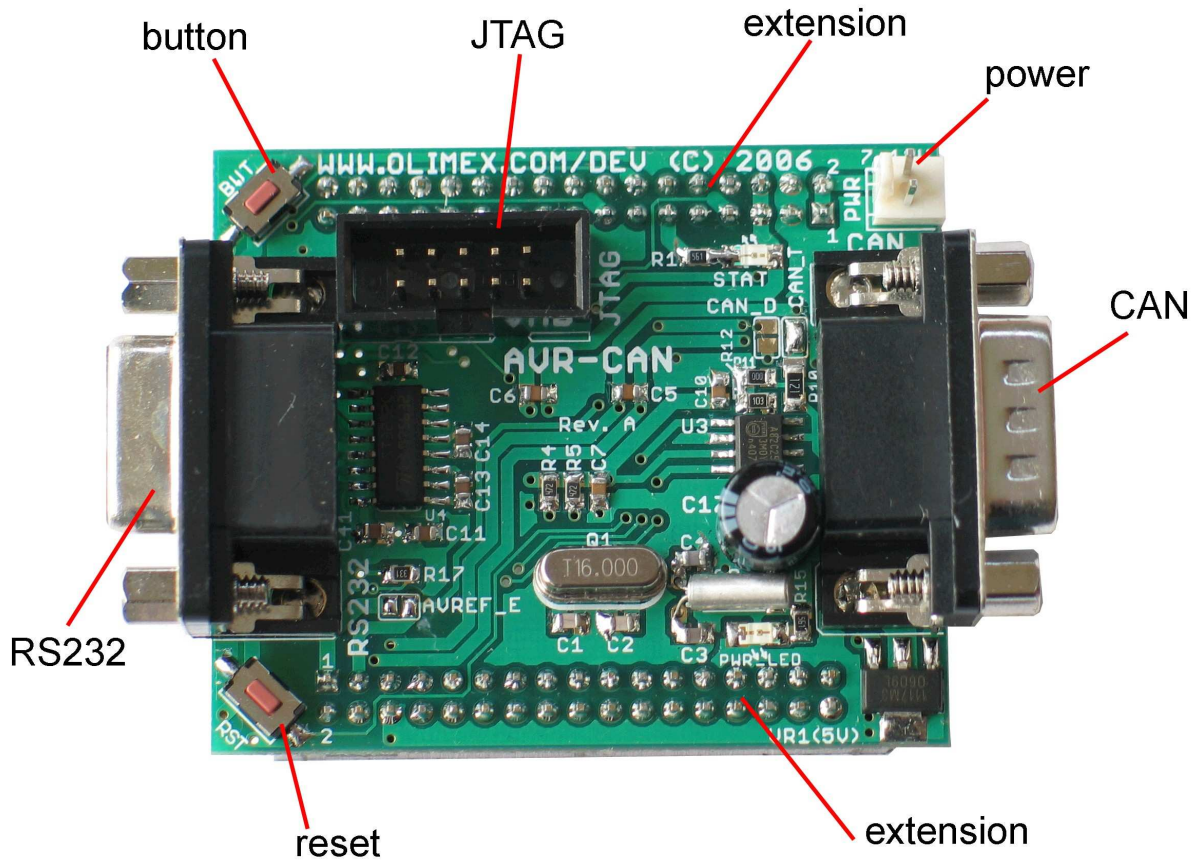


# MEMORY MAP:





## BOARD LAYOUT:



## POWER SUPPLY CIRCUIT:

Power should be applied to CON2PV2-2.5MM connector. AVR-CAN could be powered by applying up to 12VDC (7 to 12 VDC) to EXT1-1 and EXT1-2 and GND to EXT1-3 and EXT1-4 or if you supply 7 to 12 VDC to CON2PV2-2.5MM pin 2 (VIN) and GND to CON2PV2-2.5MM pin 1.

The consumption of AVR-CAN is about 40-50 mA.

## RESET CIRCUIT:

AVR-CAN reset circuit is made with R6 (10k) pull-up and button RST. On the board there is a place for the voltage supervisory device MCP120-T, but we sell it without MCP120-T.



## CLOCK CIRCUIT:

Quartz crystal 16 MHz is connected to **AT90CAN128** pin 24 (XTAL1) and pin 23 (XTAL2).

Quartz crystal 32.768 kHz is connected to **AT90CAN128** pin 19 (TOSC1/PG4) and pin 18 (TOSC2/PG3) and supplies the internal Asynchronous Timer/Counter (Timer/Counter2 in asynchronous operation).

## JUMPER DESCRIPTION:

### CAN\_D



**CAN Disable.** If this jumper is closed, the CAN is disabled. Default state is open.

### CAN\_T



This jumper assures correct work of the CAN. At each end of the bus it should be closed. This means that if you have only two devices with CAN, the jumpers of both devices should be closed. If you have more than two devices, only the two end-devices should be closed. Default state is closed.

### AVREF\_E



This jumper when closed supplies VDDA reference to the AVREF. If this jumper is open the desired reference voltage should be applied at EXT2-3. Default state is open.

## INPUT/OUTPUT:

**Status LED (red)** with name **STAT** connected to **AT90CAN128** pin 6 (OC3B/INT4/PE4).

**Power-on LED (red)** with name **PWR\_LED** - this LED shows that +5V is applied to the board.

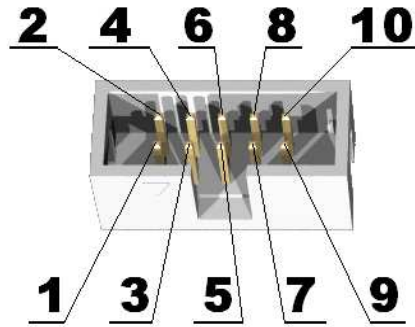
**User button** with name **BUT** connected to **AT90CAN128** pin 7 (OC3C/INT5/PE5).

**Reset button** with name **RST** connected to **AT90CAN128** pin 20 (#RESET).

## EXTERNAL CONNECTORS DESCRIPTION:

### JTAG:

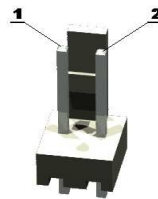
Pin #	Signal Name
1	TCK
2	GND
3	TDO
4	+5V
5	TMS
6	RESET
7	+5V
8	NC
9	TDI
10	GND



**TDI** Input **Test Data In.** This is the serial data input for the shift register.  
**TDO** Output **Test Data Out.** This is the serial data output for the shift register. Data is shifted out of the device on the negative edge of the TCK signal.  
**TMS** Input **Test Mode Select.** The TMS pin selects the next state in the TAP state machine.  
**TCK** Input **Test Clock.** This allows shifting of the data in, on the TMS and TDI pins. It is a positive edge triggered clock with the TMS and TCK signals that define the internal state of the device.  
JTAG is used to to program and debug the MCU.

### PWR:

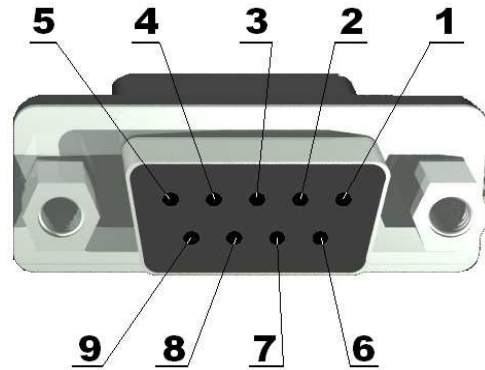
Pin #	Signal Name
1	GND
2	VIN



VIN (Voltage In) - you should supply +7 to + 12 VDC.

### RS232:

Pin #	Signal Name
1	NC
2	TXD0
3	RXD0
4	NC (not connected through R14 to pin 6)
5	GND
6	NC (not connected through R14 to pin 4)
7	NC (not connected through R13 to pin 8)
8	NC (not connected through R13 to pin 7)
9	NC

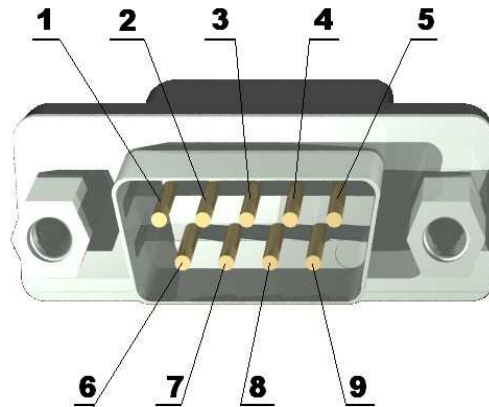


**TXD0** Output **Transmit Data**. This is the asynchronous serial data output (RS232) for the shift register on the UART controller.

**RXD0** Input **Receive Data**. This is the asynchronous serial data input (RS232) for the shift register on the UART controller.

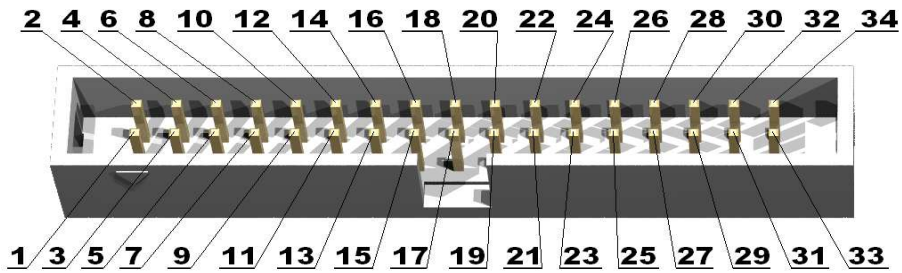
## CAN:

Pin #	Signal Name
1	NC
2	CANL
3	GND
4	NC
5	NC
6	GND
7	CANH
8	NC
9	VIN



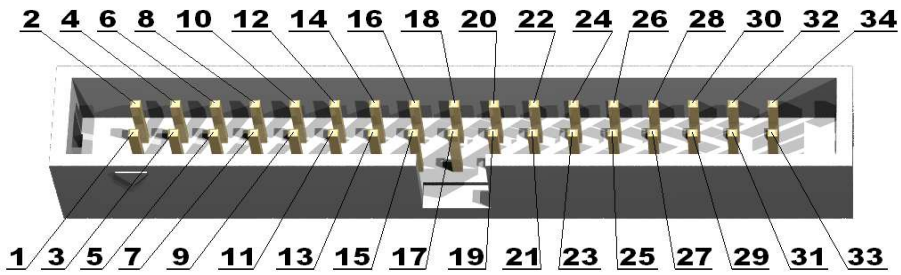
CANL and CANH are either differential input, or differential output depending on the function of the MCP2551 CAN controller (receiving or transmitting data).

## EXT1:



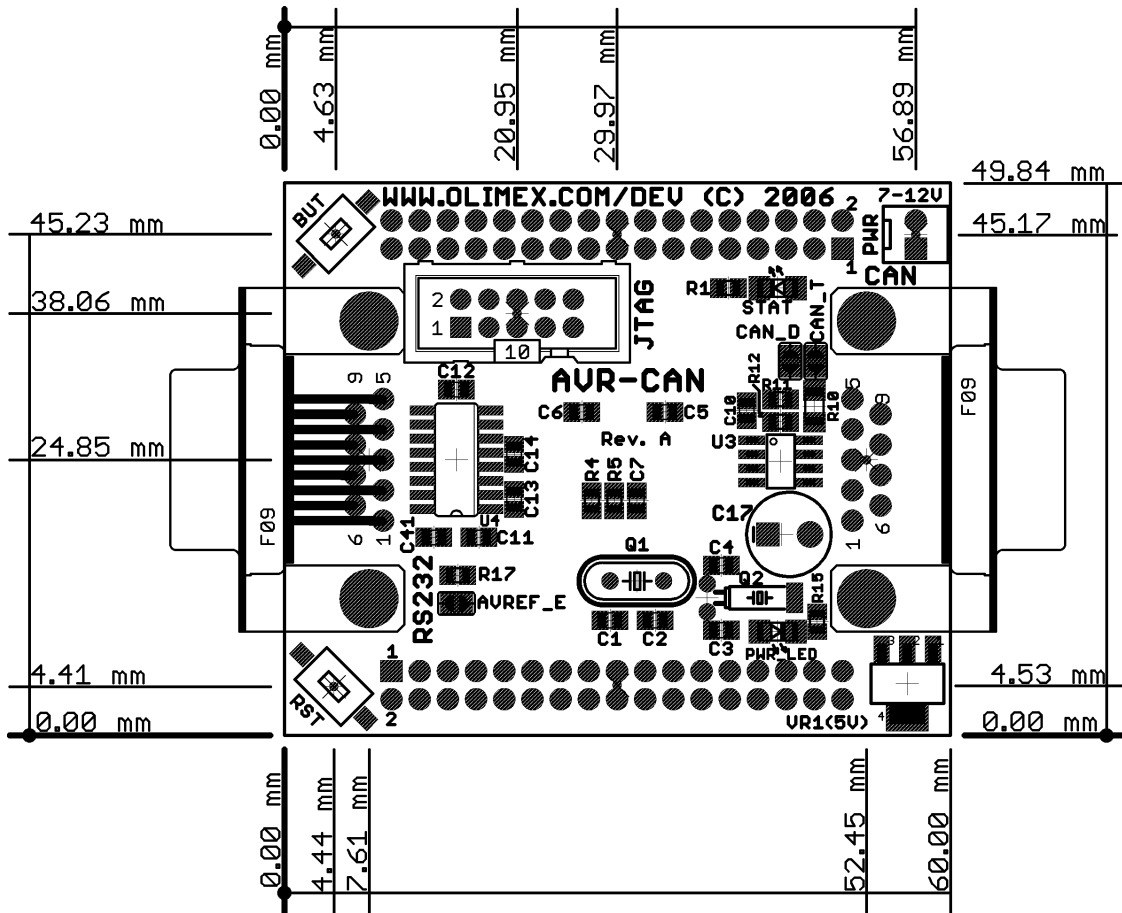
Pin #	Signal Name	Pin #	Signal Name
1	VIN	2	VIN
3	GND	4	GND
5	+5V	6	+5V
7	-	8	ICP3/INT7/PE7
9	T3/INT6/PE6	10	OC3C/INT5/PE5
11	OC3B/INT4/PE4	12	OC3A/AIN1/PE3
13	XCK0/AIN0/PE2	14	TXD0/PDO/PE1
15	RXD0/PDI/PE0	16	ADC0/PF0
17	ADC1/PF1	18	ADC2/PF2
19	ADC3/PF3	20	ADC4/TCK/PF4
21	ADC5/TMS/PF5	22	ADC6/TDO/PF6
23	ADC7/TDI/PF7	24	PA0/AD0
25	PA1/AD1	26	PA2/AD2
27	PA3/AD3	28	PA4/AD4
29	PA5/AD5	30	PA6/AD6
31	PA7/AD7	32	PG2/ALE
33	PG1/#RD	34	PG0/#WR

## EXT2:



Pin #	Signal Name	Pin #	Signal Name
1	VDDA	2	VDDA
3	AVREF	4	AGND
5	AGND	6	+5V
7	GND	8	GND
9	TXD1_OUT	10	RXD1_IN
11	PC7/A15/CLK0	12	PC6/A14
13	PC5/A13	14	PC4/A12
15	PC3/A11	16	PC2/A10
17	PC1/A9	18	PC0/A8
19	T0/PD7	20	RXCAN/T1/PD6
21	TXCAN/XCK1/P5	22	ICP1/PD4
23	TXD1/INT3/PD3	24	RXD1/INT2/PD2
25	SDA/INT1/PD1	26	SCL/INT0/PD0
27	PB7/OC0A/OC1C	28	PB6/OC1B
29	PB5/OC1A	30	PB4/OC2A
31	PB3/MISO	32	PB2/MOSI
33	PB1/SCK	34	PB0/#SS

# MECHANICAL DIMENSIONS:



## AVAILABLE DEMO SOFTWARE:

You could download AVR-CAN software form [www.olimex.com/dev](http://www.olimex.com/dev) :

- AVR-CAN Blinking LED demo code (C source and HEX)
- AVR-CAN Button read demo code (C source and HEX)
- AVR-CAN RS232 demo code (C source and HEX)

## ORDER CODE:

AVR-CAN – assembled and tested (no kit, no soldering required)

How to order?

You can order to us directly or by any of our distributors.

Check our web [www.olimex.com/dev](http://www.olimex.com/dev) for more info.



All boards produced by Olimex are RoHS compliant

## Revision history:

Board - REV.A	- created	January 2010
Users Manual – Rev. A	- created	February 2011
– added latest revision of the scheme		
– changed supply voltage		
– add page numbers		
– added more detailed mechanical dimensions		



**Disclaimer:**

© 2011 Olimex Ltd. All rights reserved. Olimex®, logo and combinations thereof, are registered trademarks of Olimex Ltd. Other terms and product names may be trademarks of others.

The information in this document is provided in connection with Olimex products. No license, express or implied or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Olimex products.

Neither the whole nor any part of the information contained in or the product described in this document may be adapted or reproduced in any material from except with the prior written permission of the copyright holder.

The product described in this document is subject to continuous development and improvements. All particulars of the product and its use contained in this document are given by OLIMEX in good faith. However all warranties implied or expressed including but not limited to implied warranties of merchantability or fitness for purpose are excluded.

This document is intended only to assist the reader in the use of the product. OLIMEX Ltd. shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information or any incorrect use of the product.