Dual-CAN

MANUAL

DualCAN
“add-on” module
für ec3xx module
from Würz elektronik

Ingenieurbüro Edwin Würz
Ackergarten 3
D-35789 Weilmünster
Tel.: ++49 6475 912944; Fax:++49 6475 912945
E-Mail: Wuerz.elektronik.@t-online.de
http://www.wuerz-elektronik.com

This manual describes the hardware of the DualCAN „add-on“ module.
This description has been prepared with utmost care. We decline any warranty for errors and their consequences.

Edition History

Edition Content/Changes Date
01 1. Edition Hardware Manual, deutsch erste Version 12.Mai 1999
01 1st Edition Hardware Manual, English, first version Jan. 10, 2000

Nomenclature

All signal names marked with „*“ are active low. Hexadecimal numbers are preceded by a ´$’ .
All squares on pin rows and jumpers designate pin 1.

This manual does not contain data sheets, standards, etc. Please refer to the supplied CD-ROM.

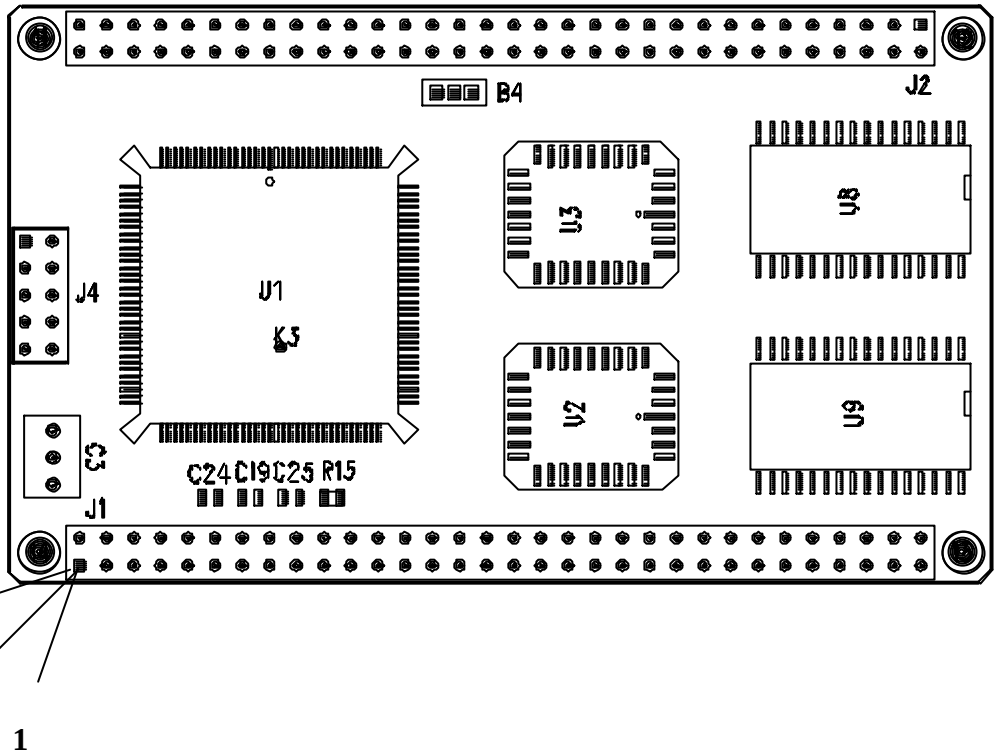
CAUTION: Electrostatic charges can destroy this module !

Contents

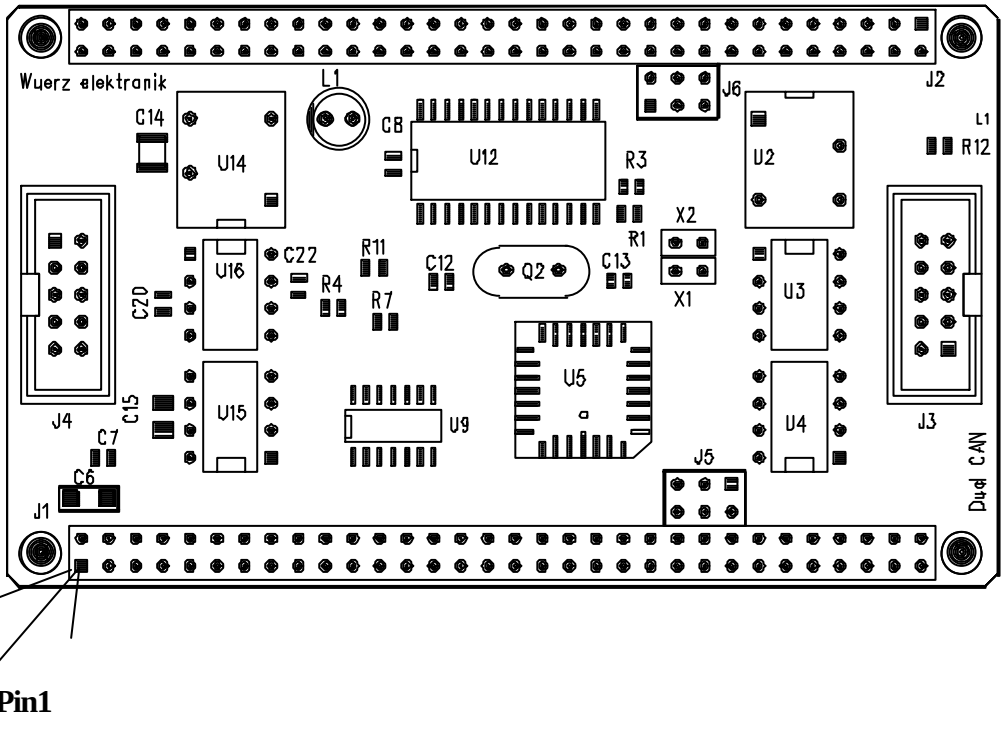
Installation of the module.

The module can be connected below or above the controller module ec3332, ec336 or ec376. The preferred position of the DualCAN Module is below the ec3xx module.
The connection of the 10 pin CAN interface to a motherboard can be made by pin rows.

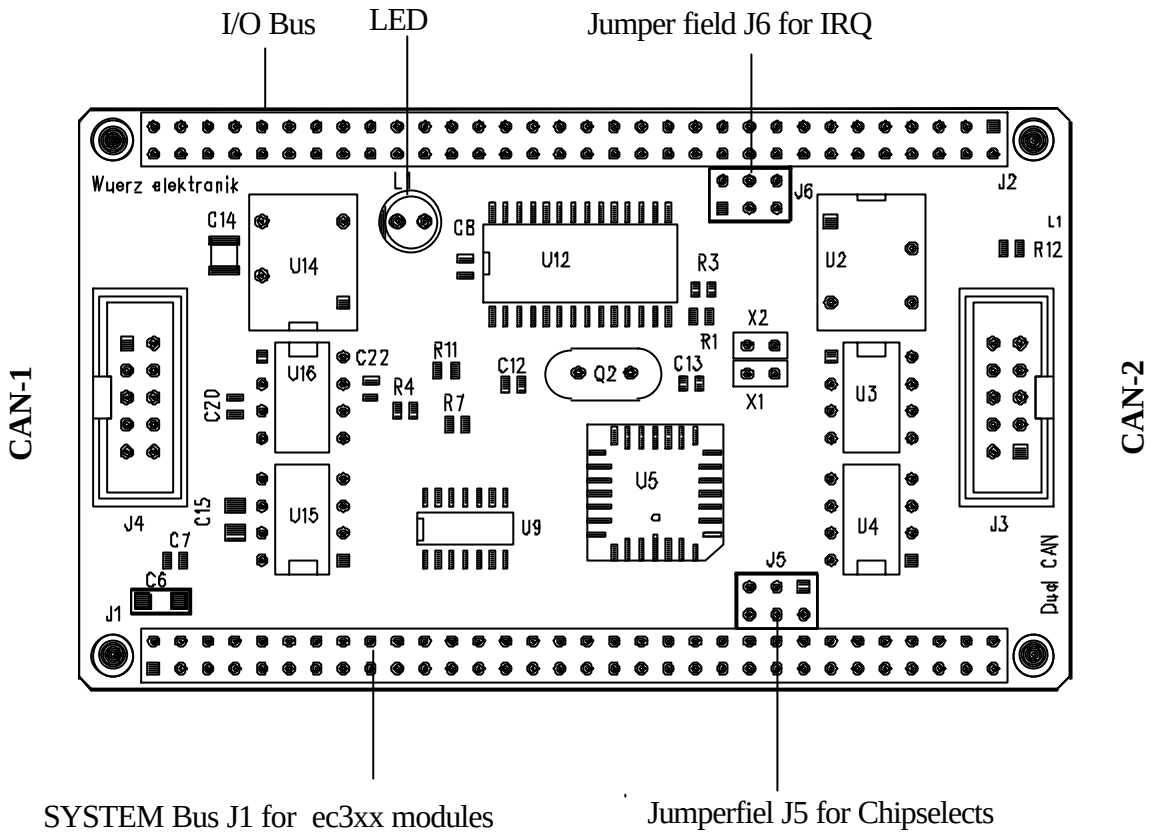
Example: Configuration of ec332 module + DualCAN module (also applies to modules ec376 and ec336)



PIN 1

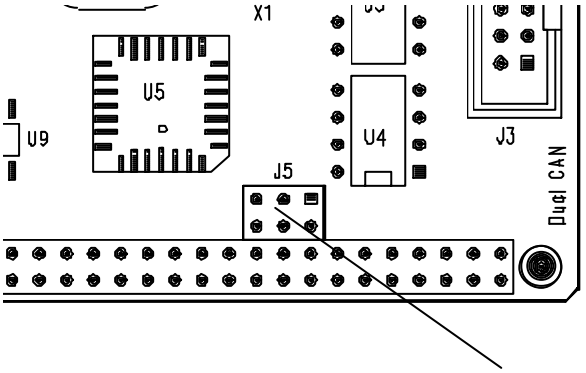


Pin1

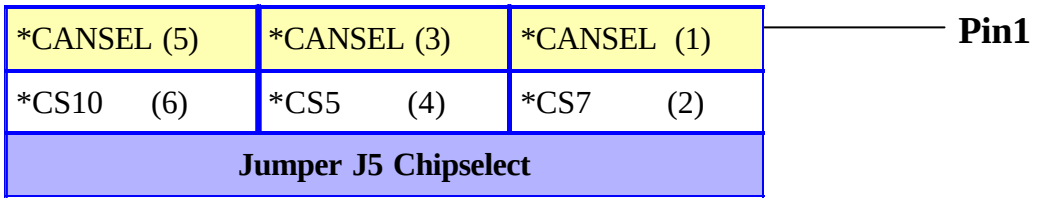


Addressing:

The module has two CAN controllers SJA1000 made by Philips and two separate, optically isolated CAN interfaces with CAN driver 82C250T.
A chip select of the ex3xx module serves for DualCAN module selection.
The chip select can be defined in a jumper field. Alternatives are *CS5, *CS7 oder *CS10.
The gal splits the corresponding chip select into two chip selects. The first controller would be at address $600000, the second one at address $600200, as an example.



CS 5 is selected by default.



The GAL contains a state machine for the control of the two CAN controllers SJA1000. Address and data lines of the 6833x processor are multiplexed. The gal also generates a DSACK. Both CAN controllers operate in INTEL mode.

*CS5 programming example, address $600.000

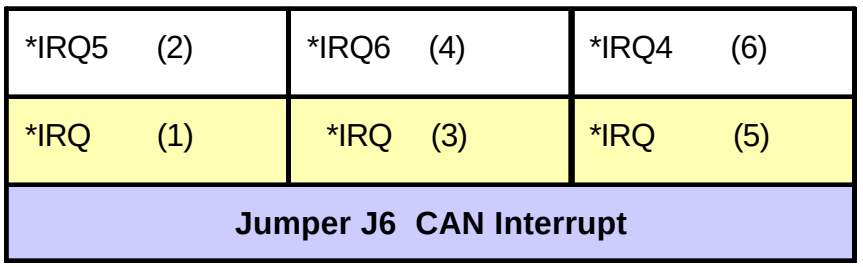
Set autovector 5 address $0078 dc.l DualCAN
Chipselect CS5 programming
move.w #$6000,csbar5 *set CS5 to $600000, 2kb space
move.w #$5bf0,csor5 * set CS5 upper read/write ext. DSACK0

Interrupt

Two interrupt processing methods exist:
I) Both CAN controllers are at the same interrupt level, but one of three interrupts can be selected.
or
II) Each CAN controller is at a fixed interrupt level.
Both methods will be described below:

Method 1: (configuration as supplied)

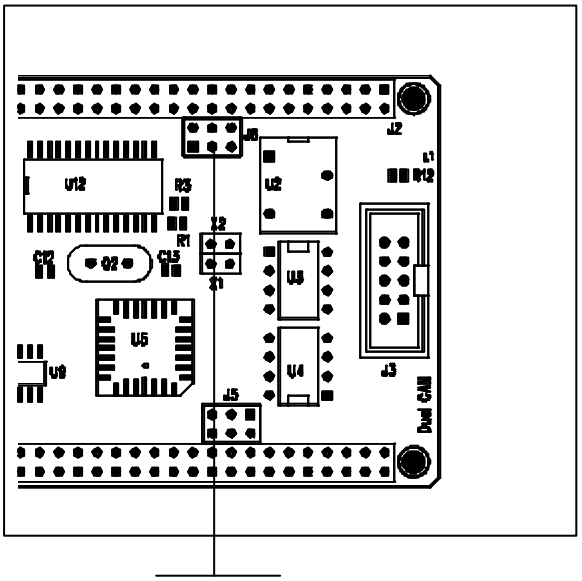
Both CAN controllers are at the same interrupt level :
Both interrupts *IRQ_CAN1 and *IRQ_CAN2 are connected by a AND gate (U9) and by jumper J6 to one of three interrupts of the MC68376.
The interrupts IRQ6, IRQ5 or IRQ4 can be used. ***The default setting is IRQ6 !***



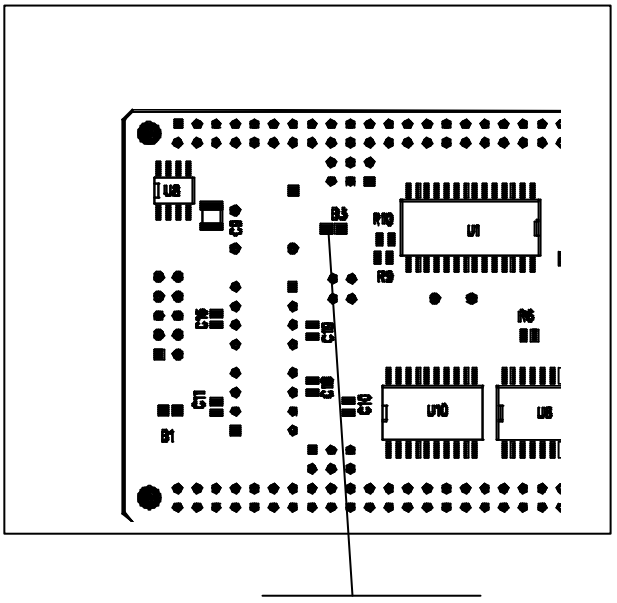
The CAN controller status registers must be polled when an interrupt occurs.
The corresponding interrupt remains and can be processed by the service routine.
The SJA1000 CAN controller cannot supply an interrupt vector, therefore an autovector must be generated. It is obtained by using the chip select of the ec3xx module.

Alternative:
Each one of both CAN controllers has its own interrupt level. In this case *IRQ_CAN1 is at *IRQ5 and *IRQ_CAN2 at *IRQ6 of the MC683xx controller.
Two additional chip selects of the MC683xx controller are needed in this configuration to generate a *IACK.
For this configuration remove jumper B3 (SMD resistor, zero Ohm) and insert the two jumpers X1 and X2. Jumper J6 must be open.

The included CAN Lib works with this option



Jumper J6



Jumper B3

CAN controller cascading *(important!)*

Both CAN controllers are cascaded. This means that the quartz oscillator is at CAN controller 1 (U12), the CLK output of the CAN controller is connected to the XTAL1 input of the second CAN controller (U1). The clock frequency after power-up or RESET is
fosc/14=2 MHz.
Bits Cd.0-Cd.2 of the clock divider register (CDR) must be set to „1“. This gives a clockout frequency of fosc=24 MHz. Programming of the register is quite easy, for instance : ***move.w #$07,CDR***

Chip select programming

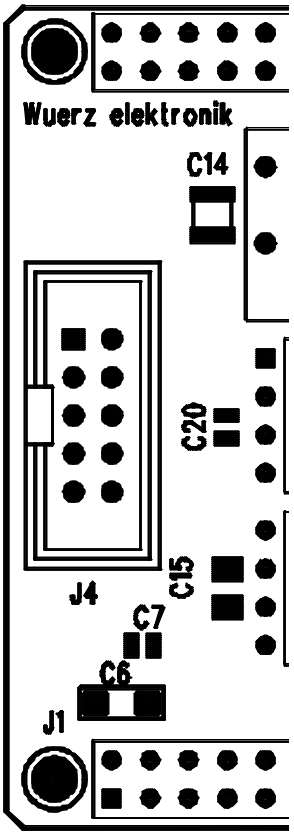
```
 ***                    Autovectrors

00000064 0000 0d30          dc.l    AINT* 1
00000068 0000 0d30          dc.l    AINT* 2
0000006c 0000 0d30          dc.l    AINT* 3
00000070 0000 0d30          dc.l    AINT* 4
00000074 0000 0d30          dc.l    AINT* 5

00000078 0000 0fdc          dc.l    DualCAN    * Autovector 6
                *
0000007c 0000 0d30          dc.l    AINT* NMI
                *************************************************************
                * Chipselect /CS5 DualCAN

000005a8              InitCS5    equ *
000005a8 33fc 6000 ffff move.w #$6000,csbar5    * set CS5  to $600000   2KB
       fa60
000005b0 33fc 5bf0 ffff  move.w #$5bf0,csor5     * set CS5 upper read/write ext. DSACK0
       fa62
```

** Generation of /avec , /iack

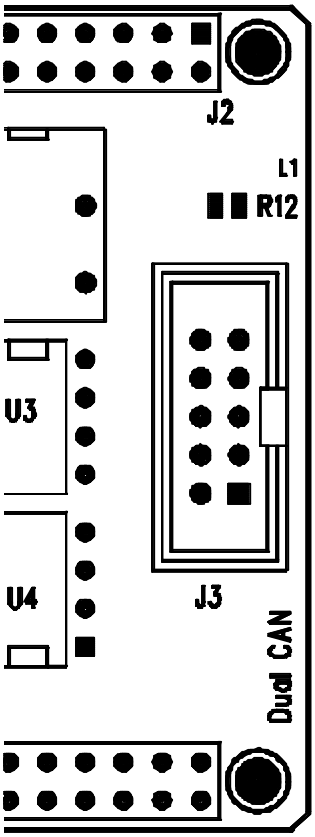
```
move.w #$fff8,csbar10    * set CS10

*Irq 5
move.w #$280b,csor10     * set CS10 upper read/ 0 wait irq 5

 *Irq 6
move.w #$280d,csor10     * set CS10 upper read/ 0 wait irq 6
```

Connector allocation of first CAN interface

J4 -1		I - GND	J4 - 2
J4 - 3	CAN - L	CAN - H	J4 - 4
J4 - 5	I - GND		J4 - 6
J4 - 7			J4 - 8
J4 - 9			J4 -10
Connector J4 für CAN1			



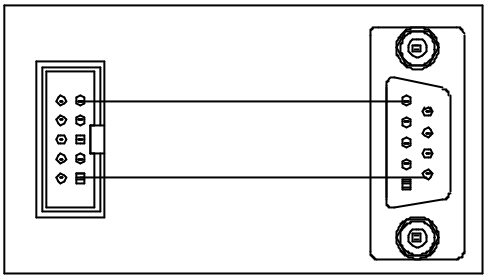
Connector allocation of second CAN interface



J3 -10			J3 - 9
J3 - 8			J3 - 7
J3 - 6		I - GND	J3 - 5
J3 - 4	CAN - H	CAN - L	J4 - 3
J3 - 2	I - GND		J3 -1
CAN2 - Connector J3			

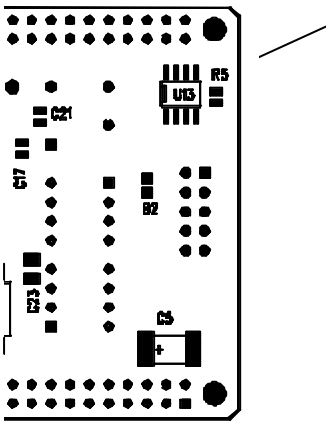
With a ribbon cable and a 9-pin male D-Sub connector you obtain (according to pin allocation recommended by CiA/DS 102-1):

2	CAN-L
3	GND
6	GND
7	CAN-H
9	V+ (optional)

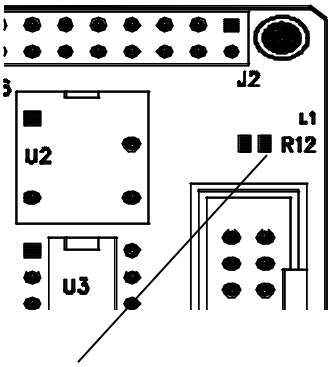


Bus termination resistor 120 Ohm.

A 120 Ohm bus termination resistor can be installed between the signal lines CAN_H and CAN_L. R5 and R12 are 120 Ohm 1/8 Watt SMD type 0805.



Bustermination resistor
R5 = 120 Ohm
CAN-Controller 1



Bustermination resistor
R12 = 120 Ohm
CAN-Controller 2

The CAN-BUS driver 82C250T

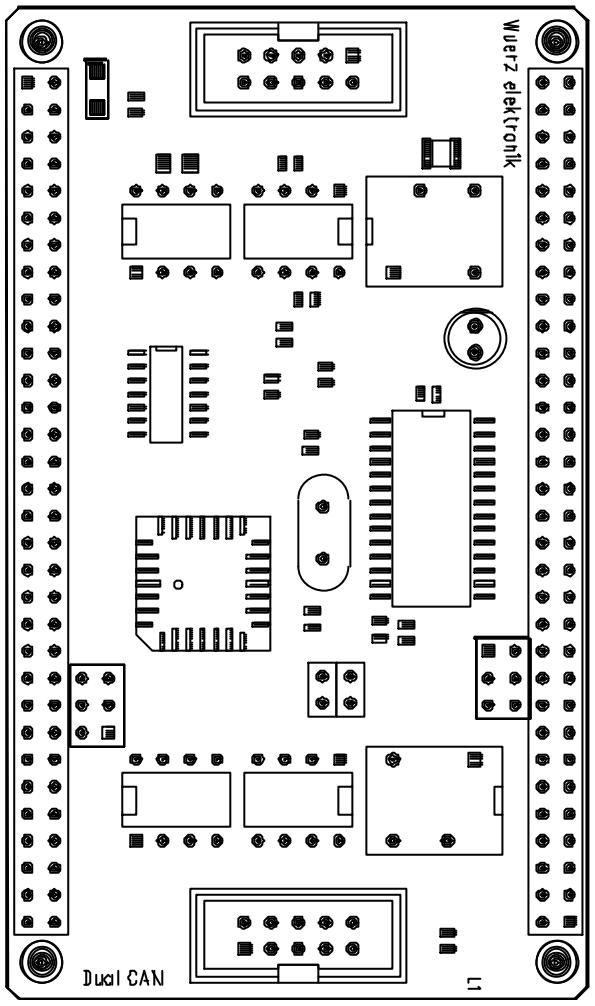
is galvanically isolated by optocoupler 6N137 and DC/DC converter and meets the ISO/DIS-11898 standard. Possible transfer rates up to 1MBit/s.

LED

The module has a LED. This LED can be controlled by MODCLK by configuring pin MODCLK as an output. MODCLK Low=LED active.

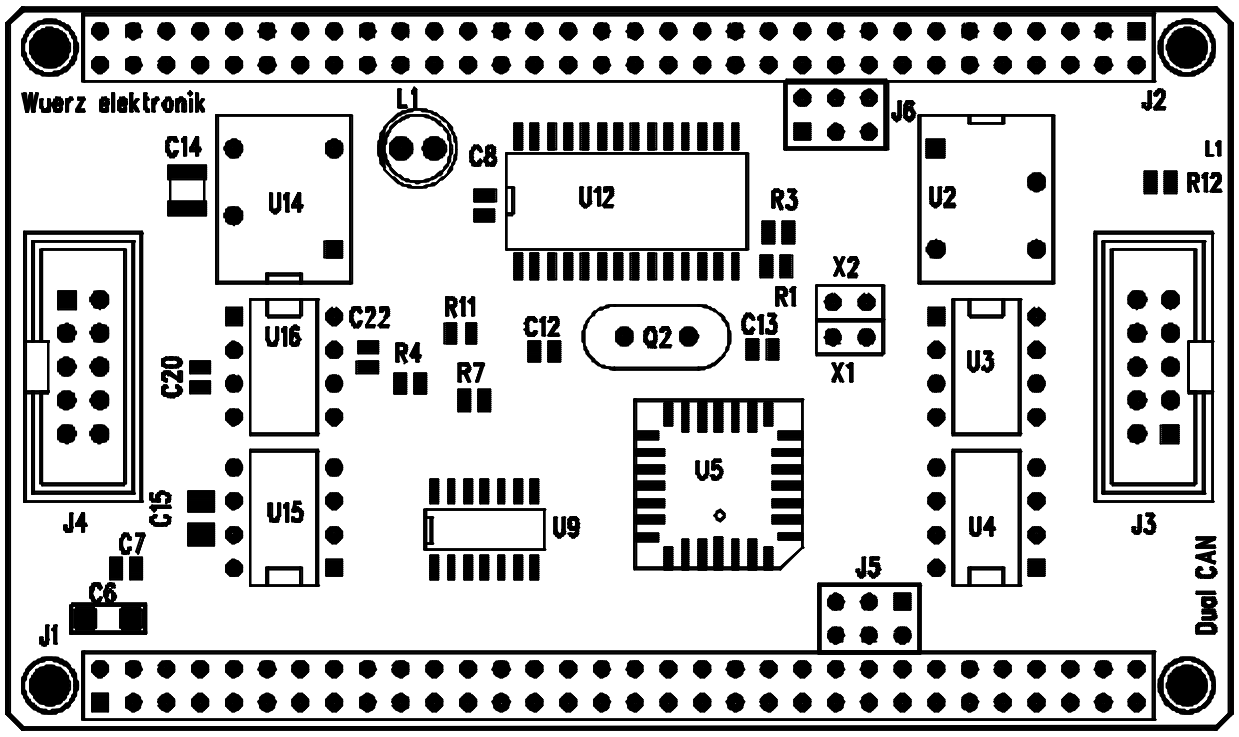
Module bus allocation

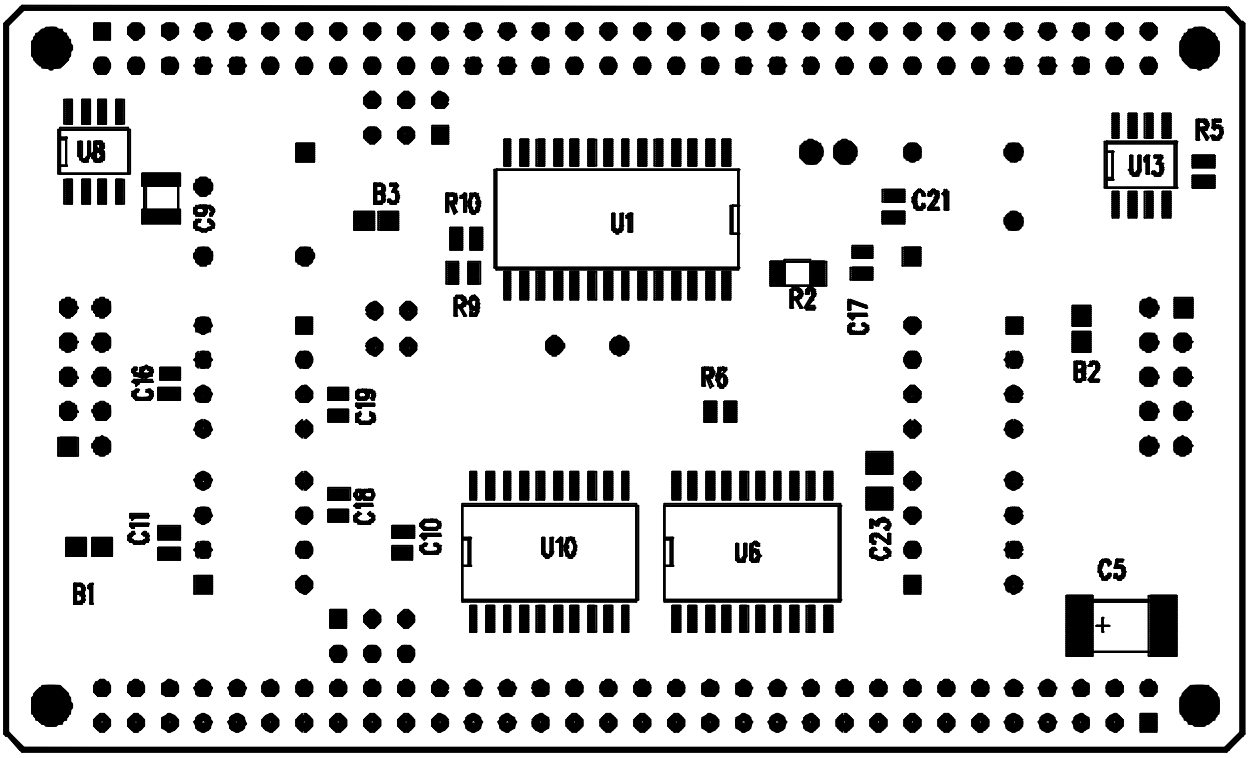
J1-A		J1-B
VCC	**1**	VCC
GND	**2**	GND
D15	**3**	D14
D13	**4**	D12
D11	**5**	D10
D9	6	D8
	7	
	8	
	9	
	10	
*RESET	11	
*IRQ5	12	CLKOUT
	13	
	14	
*DSACK0	15	
	16	
*AS	17	R/*W
	18	
	19	
FC2/*CS5	20	
A23/*CS-10	21	
	22	A20/*CS7
	23	
	24	
	25	
	26	
	27	
	28	**A8**
A7	29	A6
A5	30	A4
A3	31	A2
A1	32	A0
J1-A		**J1-B**



J2-B		J2-A
	32	
	31	
	30	
	29	
	28	
	27	
	26	
	25	
	24	
	23	
	22	
	21	
	20	
	19	
	18	
	17	
MODCLK	16	
	15	
	14	
	13	
	12	
	11	
	10	
	9	
	8	
*IRQ4	7	*IRQ6
	6	
	5	
	4	
	3	
	2	
	1	
J2-B		**J2-A**

Systembus **I/O Bus**


Wuerz elektronik
L1
J6
J2
C14
U14
C8
U12
R3
U2
L1
R12
X2
R1
C22
R11
C12
Q2
C13
U16
R4
R7
C20
X1
U3
U5
C15
U15
U9
U4
J4
J3
C7
J5
C6
J1
Dual CAN


U8
R5
U13
B3
R10
U1
C21
C9
R9
R2
C17
R6
B2
C16
C19
C18
C10
U10
U6
C23
C5
B1
C11
+