ALON1 – BR
USB automation card
24 bit, AC/DC load cell front-end

User's manual

Dr. Gal Ben-David
POB 82, Adi 17940, Israel
www.galium.co.il
gal@galium.co.il
Tel: +972-54-4698289
Fax: +972-4-9860401

Information provided is believed to be accurate and reliable. However, no responsibility is assumed for its uses, nor for any infringements of patents or other rights of third parties which may result from its use.

Hardware and software are sold “as is”. We assume no responsibility for direct or indirect damages, including equipment fault, computer malfunction, loss of data, and loss of income.

We, at *Dr. Gal Ben-David,* hold a policy of continuing improvement of our products. We reserve the right to change and improve any of our products without prior notice. Please check our web site for last software versions.

Table of content

Revision History

Date	Details
17-MAR-2005	Original

Acknowledgment

Dear valued customer,

I would like to thank you for purchasing an ALON1 USB automation card. The card was designed to allow simple and efficient control for various real world applications with the ease and expendability of USB interface. I believe that you will find our boards to be reliable and easy to use.

The most important person in our company is our customer. Our products are as useful as what we know about customer needs, what his/her expectations are and how our products will better serve him/her. I will appreciate having any suggestion, feedback and ideas that will make your designs better. We are a small and agile company, and we fortunate to love what we do. You will be surprised by the short time that we take good idea into good products.

Regards

Dr. Gal Ben-David
gal@galium.co.il

The ALON1 family of USB interface boards

Please refer to the document ***Alon1 Documentation.PDF*** for technical information on the entire ALON1 family.

Hardware interface

The ALON1 BR card is a complete analog front end for weigh-scale and pressure measurement applications. The board provides 5V AC or DC excitation and samples the bridge output.

The board accepts low level signals directly from a transducer and outputs digital information via USB connection. The input signal is applied to a programmable gain front end based around an analog modulator.

The board features a buffered differential programmable gain analog input as well as a differential reference input. The board operates from a single +5 V supply, provided either from the USB bus or by an external power supply. It accepts four unipolar analog input ranges: 0 mV to +10 mV, +20 mV, +40 mV and +80 mV and four bipolar ranges: ±10 mV, ±20 mV, ±40 mV and ±80 mV. The peak-to-peak resolution achievable directly from the part is 1 in 230,000 counts. An on-board 6-bit DAC allows the removal of TARE voltages. Clock signals for synchronizing ac excitation of the bridge are also provided.

External power supply

The ALON1-BR analog section is powered by an external power supply - VCOMMON. The power supply should be connected between AGND and AVIN+ on J6.

The power supply is limited to 5V by a Transient Voltage Suppressor (TVS). The TVS will also short on reverse polarity.

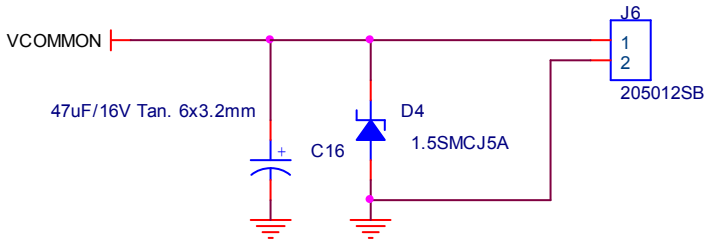


Figure 1 – Over voltage protection

It is possible to used USB 5V power as AVIN+. Connect the 5V out on J5 to AVIN+ on J6 using a ferrite bid provided with the board.

Analog front end

A conceptual circuit diagram of the analog front end is shown in figure 2.

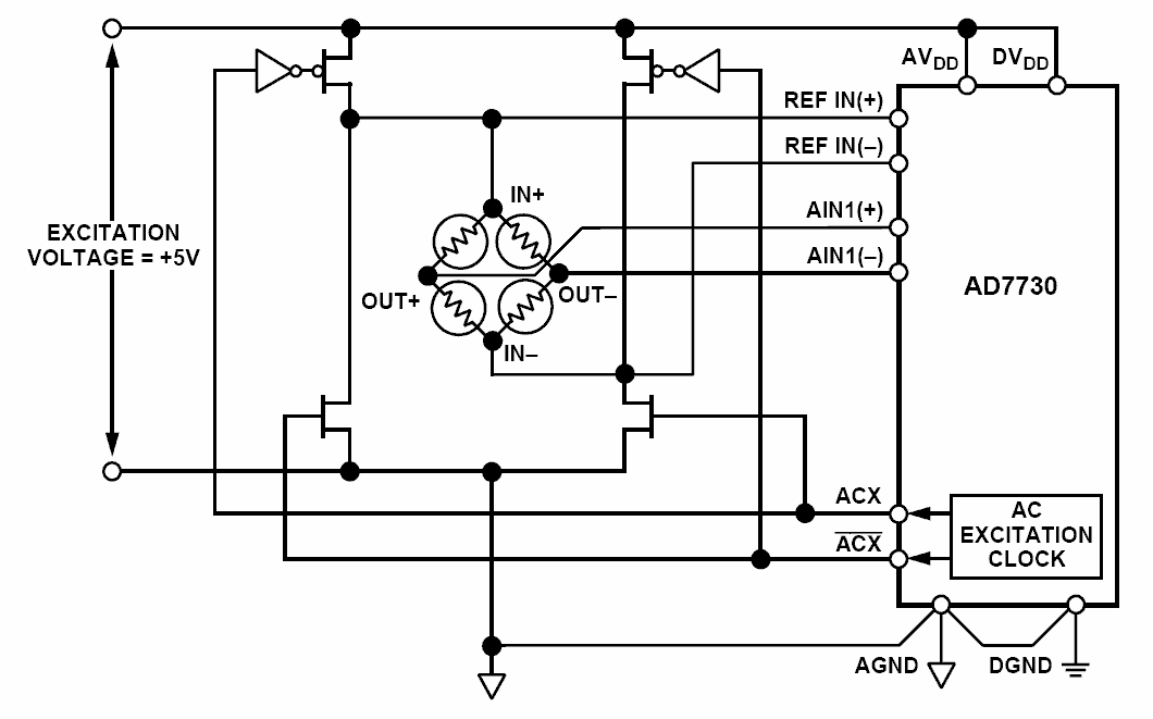


Figure 2 – AD7730 analog front end

The board is based on Analog Devices AD7730L Bridge Transducer ADC. The bridge may be excited by 5V AC or DC source. AC excitation is achieved by a MOSFET transistors bridge whose timing is provided by the AD7730 signals ACX and ACX#.

The AD7730 uses ratiometric measurement. The transducer excitation is used as A/D reference (REFINP and REFINN), while the A/D samples the bridge measurement.

The bridge connections are shown in figure 3

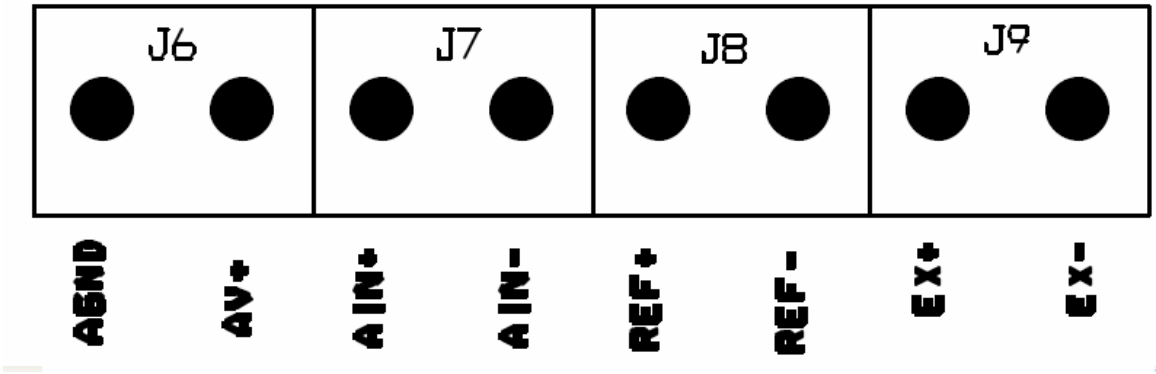


Figure 3 – bridge connection

Digital input

A circuit diagram of the digital input is shown in figure 4.

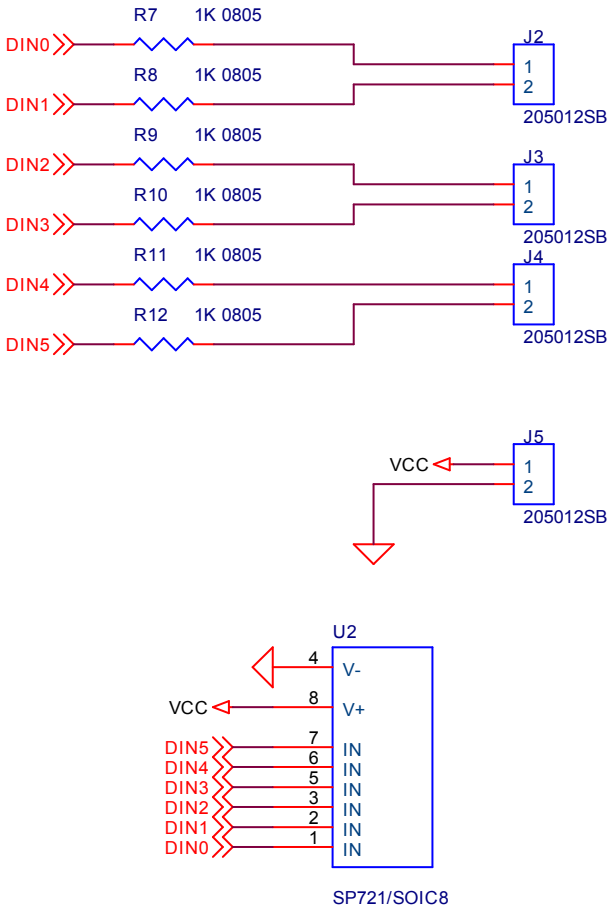


Figure 4 – Digital input

The digital inputs are connected to the CPU using current limiting resistors and over voltage protection. User may connect voltage 0-1V as logic ‘0’ and 3-24V as logic ‘1’.

Panel connections are shown in figure 5.

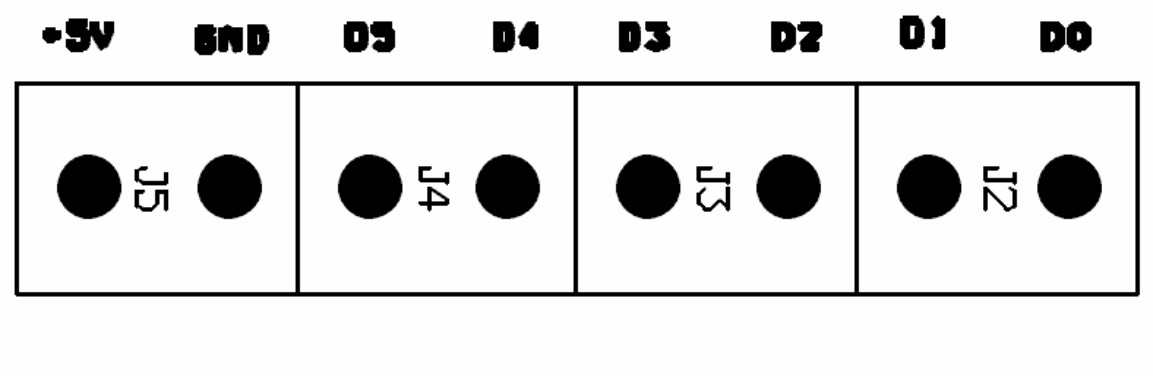


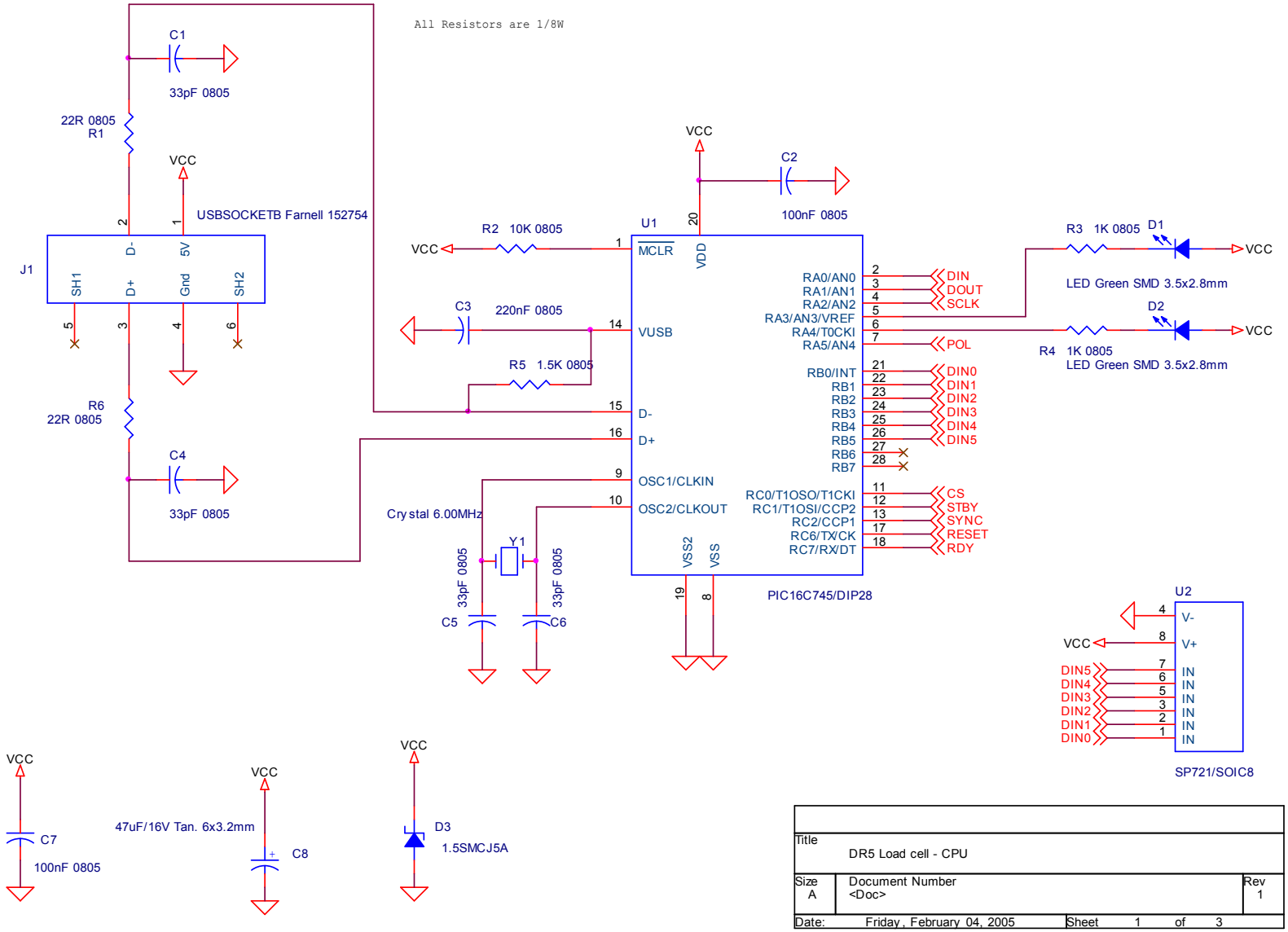
Figure 5 – Digital input connections

Using the 5V connection

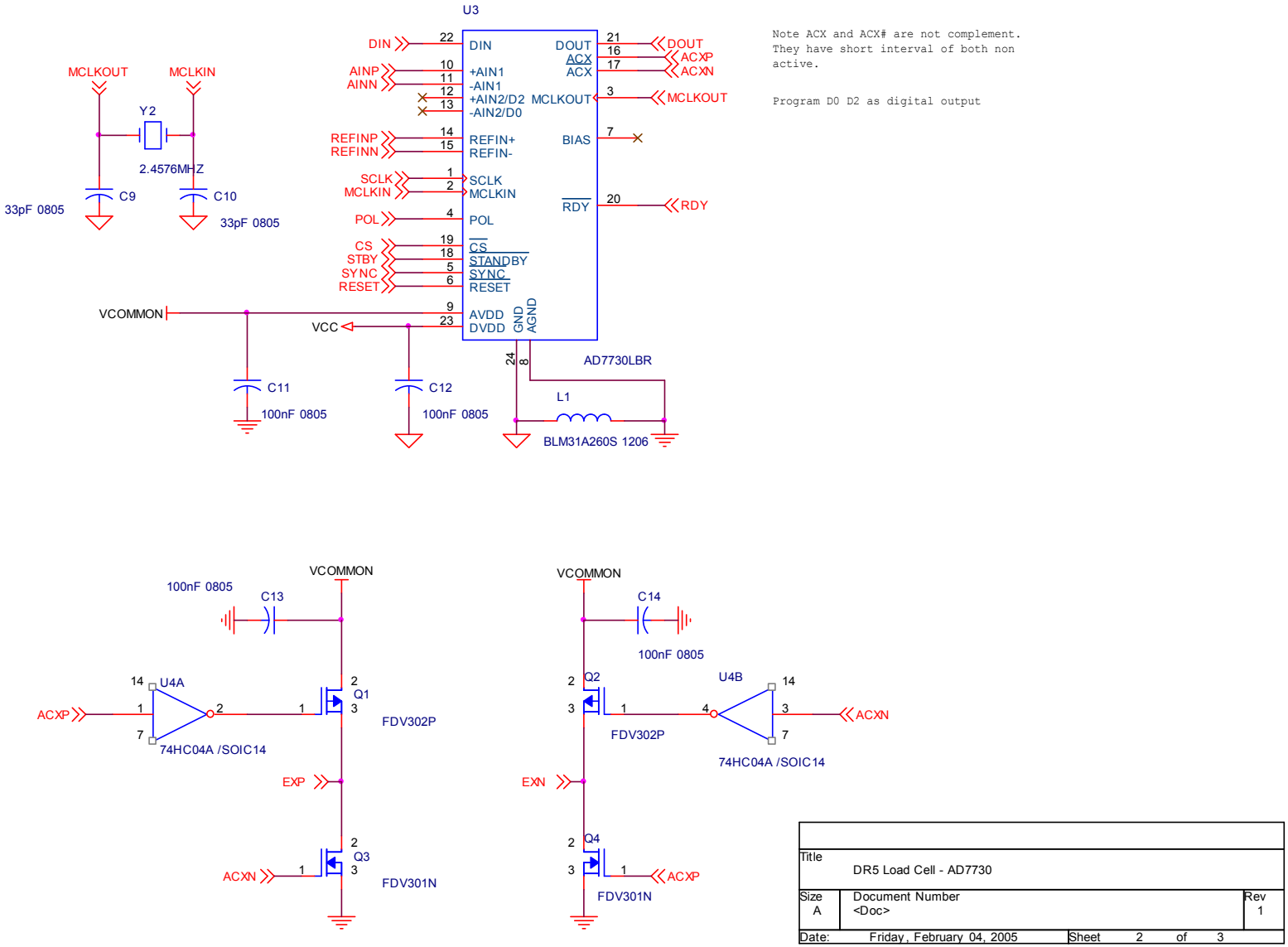
The board has a 5V connection for custom use. Since USB device may use 500mA and the self consumption of the board is 100mA, the user circuit may consume no more than 400mA.

Caution: The 5V supply is connected directly from the PC/Hub power supply. Care must be taken in order not to consume more power than a total of 500mA per card, including card's own function. High currents and short circuits may damage the PC/Hub. Do not make connections when the board is connected to the PC and PC is powered.

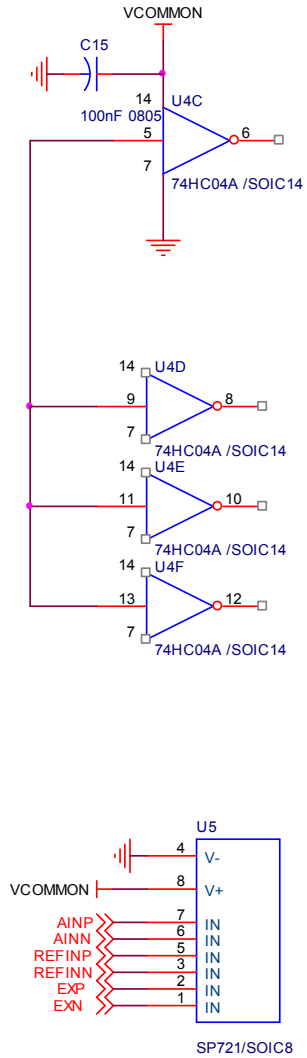
Hardware Schematics

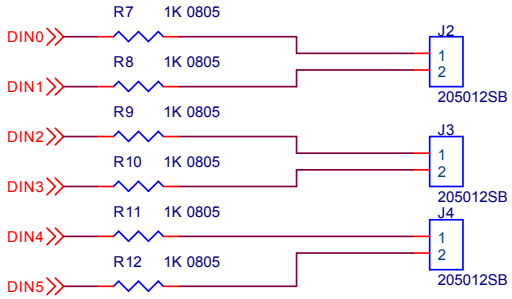


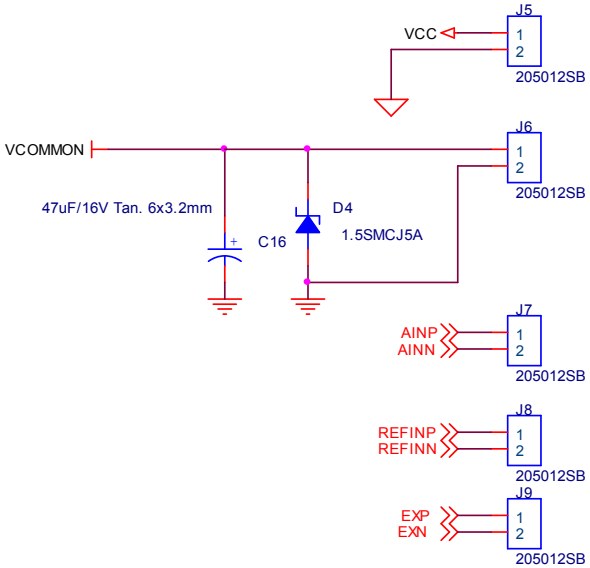
Title	DR5 Load cell - CPU	
Size A	Document Number <Doc>	Rev 1
Date: Friday, February 04, 2005	Sheet 1 of 3	



Title	DR5 Load Cell - AD7730		
Size A	Document Number <Doc>		Rev 1
Date: Friday, February 04, 2005		Sheet 2 of 3	

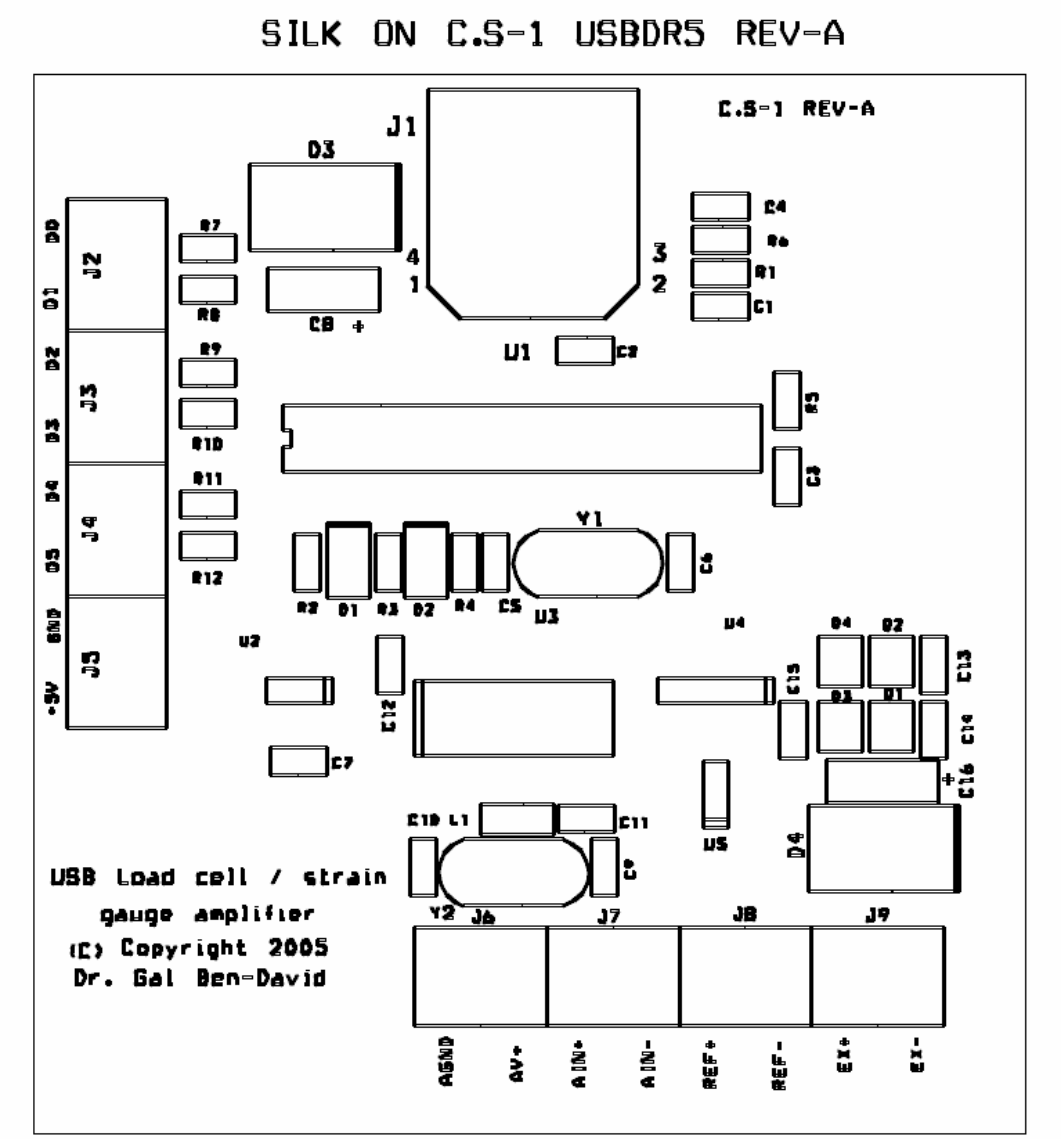






Title: DR5 Load cell - Connectors		
Size: A	Document Number: <Doc>	Rev: 1
Date: Friday, February 04, 2005	Sheet 3 of 3	

PCB layout


SILK ON C.S-1 USBDR5 REV-A
C.S-1 REV-A
J1
D3
J2
J3
J4
J5
R7
R8
R9
R10
R11
R12
C8
U1
C2
C4
R6
R1
C1
R5
C3
Y1
C6
U3
U2
U4
C12
C7
C15
C13
C14
C16
U5
D4
C10
L1
C11
C9
Y2
J6
J7
J8
J9
USB Load cell / strain gauge amplifier
(C) Copyright 2005
Dr. Gal Ben-David
AGND
AV+
AIN+
AIN-
REF+
REF-
EX+
EX-

Software programming

Please refer to the document ***Alon1 Documentation.PDF*** for general information on programming the entire ALON1 family. All board functions use an **`ALON1BOARDSTRUCT`** structure for communication. The structure is explained in the general document.

The product ID of the digital IO is 13.
The AD7730 uses few command register and provide data and status registers.
The board provides direct access to these registers.
All commands are right justified toward the Least Significant Bits.

Command registers

```
int ALON1_BR_SetModeRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned short Mode); // Input: 16 bit mode register
// returns zero if OK, nonzero if board was not found

int ALON1_BR_SetFilterRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long Filter); // Input: 24 bit mode register (LSB)
// returns zero if OK, nonzero if board was not found

int ALON1_BR_SetDACRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned char DAC); // Input: 8 bit DAC register
// returns zero if OK, nonzero if board was not found

int ALON1_BR_SetOffsetRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long Offset); // Input: 24 bit offset register
(LSB)
// returns zero if OK, nonzero if board was not found

int ALON1_BR_SetGainRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long Gain); // Input: 24 bit mode gain (LSB)
// returns zero if OK, nonzero if board was not found
```

Status/Data registers

```
int ALON1_BR_GetStatusRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned char *Status); // Output: 8 bit status register
// returns zero if OK, nonzero if board was not found

int ALON1_BR_GetDataRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long *Data); // Output: 24 bit data register
// returns zero if OK, nonzero if board was not found

int ALON1_BR_GetModeRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned short *Mode); // Output: 16 bit mode register
// returns zero if OK, nonzero if board was not found

int ALON1_BR_GetFilterRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long *Filter); // Output: 24 bit mode register
(LSB)
// returns zero if OK, nonzero if board was not found

int ALON1_BR_GetDACRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned char *DAC); // Output: 8 bit DAC register
// returns zero if OK, nonzero if board was not found
```

```
int ALON1_BR_GetOffsetRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long *Offset); // Output: 24 bit offset register
(LSB)
// returns zero if OK, nonzero if board was not found

int ALON1_BR_GetGainRegister(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned long *Gain); // Output: 24 bit mode gain (LSB)
// returns zero if OK, nonzero if board was not found
```

It is possible to read input state by

```
int ALON1_BR_GetInputs(
ALON1BOARDSTRUCT *, // Input: board structure
unsigned char *CurrentState); // Output : Digital In state
(LSB)
// returns zero if OK, nonzero if board was not found
```

The main callback function is notified in cased of device attachment, detachment and input change. Message is called once on board attachment (or power on), with **CardAttached=1** and once on board removal, **CardAttached=0;**
The message information is encapsulated in a single structure

```
struct
{
     unsigned char CardAttached;
     unsigned char InputState;
     unsigned long Data;
} ALON1_BR_MESSAGE;

int CALLBACK ALON1CallBackFunction
(ALON1BOARDSTRUCT *Alon1Boardtruct, CONST VOID *Param)
{
ALON1_BR_MESSAGE *BRMessage;
     switch (Alon1Boardtruct->ProductID)
     {
          case PRODUCT_ID_BR:
               BRMessage=(BR_MESSAGE)Param;
               // TODO: React according to new state
               // If card is detached,
               // do not use the boardsturct
          break;
     }
}
```