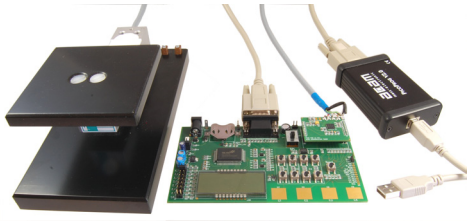
mess • electronic

PICOSTRAIN

Manual

PSØ9-EVA-KIT

Evaluation System for PSØ9



19th May 2011
Document-No.: DB_PSØ9-EVA_en V0.1

Published by acam-messelectronic gmbh

Disclaimer / Notes

„Preliminary" product information describes a product which is not in full production so that full information about the product is not available yet. Therefore, acam messelectronic GmbH („acam") reserves the right to modify this product without notice. The information provided by this data sheet is believed to be accurate and reliable. However, no responsibility is assumed by acam for its use, nor for any infringements of patents or other rights of third parties that may result from its use. The information is subject to change without notice and is provided „as is" without warranty of any kind (expressed or implied). Picostrain is a registered trademark of acam. All other brand and product names in this document are trademarks or service marks of their respective owners.

Support

For a complete listing of Direct Sales, Distributor and Sales Representative contacts, visit the acam web site at: http://www.acam.de/company/distributors or refer to chapter 8 in this datasheet

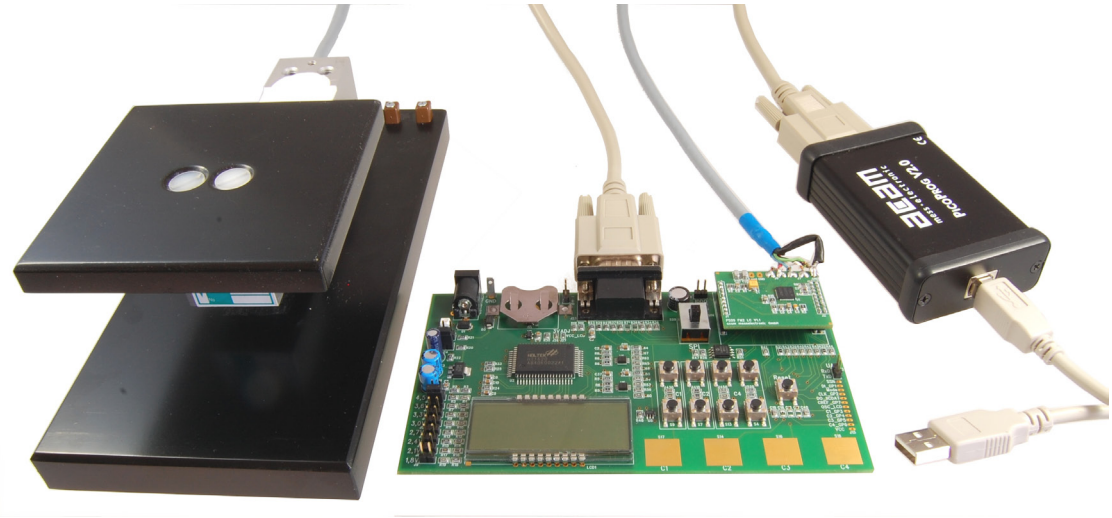
For technical support you can contact the acam support team in the headquarter in Germany or the Distributor in your country. The contact details of acam in Germany are:
support@acam.de or by phone +49-7244-74190.

Table of Contents

1 Introduction

1.1 General

The PSØ9-EVA-KIT evaluation system provides a complete weighing system made of a main board with LCD panel, two plug-in modules, a 10kg load cell platform and Windows-based evaluation and assembler software. Optionally, the TTL-232R-3V3 cable, a USB TTL serial cable from FTDI is provided if the UART in the PSØ9 is to be tested. The EVA-Kit offers extensive and user-friendly configuration and evaluation of the PSØ9 single-chip solution for weigh scales.



1.2 System Overview

PSØ9-EVAL-MB Main Board

- LCD display 22 x 51mm²
- Interface to external LCD controller, Holtek HT1620
- Power select by 2 jumpers:
  - Battery holder for CR2032
  - Wall power supply, Voltage selectable by on-board jumpers
- 9 Push buttons (resistive keys)
- 4 Capacitive (touch) keys
- A SPDT switch to select either SPI or IIC serial interface between the PSØ9 and the PICOPROG programmer
- UART interface

PSØ9-EVAL HR-Module

- For up to 100,000 stable scale divisions & solar applications
- Up to 4 half bridges / Full bridge / Wheatstone bridge
- 4 layer PCB

PSØ9-EVAL LC-Module

- For up to 30,000 stable scale divisions and low cost applications
- Up to 4 half bridges / full bridge / Wheatstone bridge
- 2 layer PCB

PICOPROG V2.0 Programmer

- (USB to SPI/IIC interface)

10kg Load Cell

- 350 Ohm sensor
- Mounted on platform
- Wired as 2 half bridges with 1 span compensation resistor

Optional FTDI TTL-232R-3V3 cable (available upon request). This cable can be provided to test the UART of the PSØ9 by establishing communication between the UART of PSØ9 and a serial terminal application on a

Windows PC.

1.3 Component List

- PSØ9-EVA-MB Motherboard
- PSØ9-EVA-Module-HR Plug-in module- High resolution
- PSØ9-EVA-Module-LC Plug-in module- Low cost
- PICOPROG V2.0 Programmer
- Demo scale 10kg load cell with platform
- High density DSUB15 cable Connecting the Evaluation board to the programmer
- USB cable Connecting PICOPROG to the PC
- Wall power supply
- CD-ROM Incl. software and data sheets
- Optional - FTDI TTL-232R-3V3 cable (upon request)

2 Connecting Strain Gauges 2-2

2 Connecting Strain Gauges

The evaluation system comes with the load cell already connected to the plug-in module.

In the following we will explain how a user can apply his own load cell. In case of sensors with other than 350 Ohm resistance the discharge capacitor Cload has to be adapted.

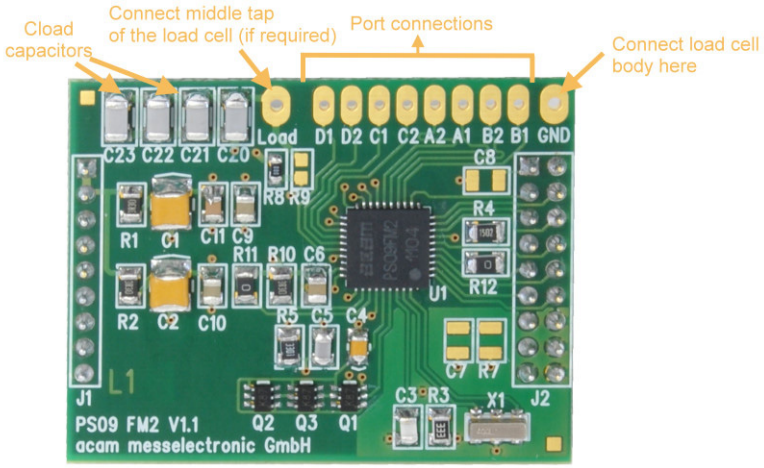
The 2 different plug-in modules are designed to support various applications of PSØ9. For high resolution and solar applications, for up to 100,000 stable scale divisions, the external bipolar comparator circuit is used. For applications with lower current consumption and low resolution requirement, the LC variant is used, running with the internal comparator. The LC module has minimum components and is the 'low-cost' variant therefore; however resolution will be clearly lesser than with the HR module.

2.1 HR - High Resolution Module

The HR module is targeted for high resolution and solar applications, for up to 100,000 stable scale divisions. The external bipolar comparator circuit is used in this module.

It is possible to measure up to 4 half bridges. Due to the PICOSTRAIN measurement principle the system does not need a full bridge. Two resistors, in the following called half bridge, are sufficient.

Figure 2-1: High Resolution Module



2.2 LC – Low Cost module

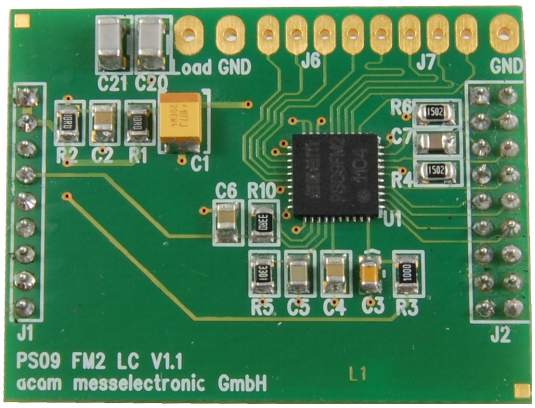
This is a low-cost version of the High resolution module, with minimum necessary components for operation. The following are the distinct features that differentiate the LC module from the HR module:

1. No external comparator. This makes the LC module suitable for applications with high, but not the highest resolution. The internal comparator is used. This reduces the base resolution by 0.8 bit compared to the external comparator.

2. The 4 MHz ceramic crystal oscillator is not present and there is no possibility to connect an external RC oscillator to PSØ9. Thus the PSØ9 can be operated only with the built in RC-oscillator. Operating the PSØ9 with the

internal RC oscillator is known to limit the resolution to approximately 16 bits. An advantage is the lower current consumption, which is around 1.2 mA current at 3V for maximum performance. To configure the internal oscillator for operation please refer to section 4.1 of PSØ9 data sheet.

Figure 2-2: Low-cost Module



2.3 Connecting the Modules (HR and LC)

2.3.1 Capacitor Selection and Assembly

The PICOSTRAIN measurement principle is based on measuring the discharge time of a capacitor. For this reason the correct size and material of the capacitors is significant to achieve best measuring results. In general we recommend a discharging time in the range of

$$\tau = 0.7 * R * C = 80 \text{ to } 120\ \mu s.$$

As material we recommend COG or CFCAP (Multilayer ceramic from Taiyo-Yuden). X7R capacitors can be used, too, but will show some minor loss in temperature stability.

The recommend values are:

Rsg = 350 Ohm → Cload = 300 nF to 400 nF

Rsg = 1000 Ohm → Cload = 100 nF to 150 nF

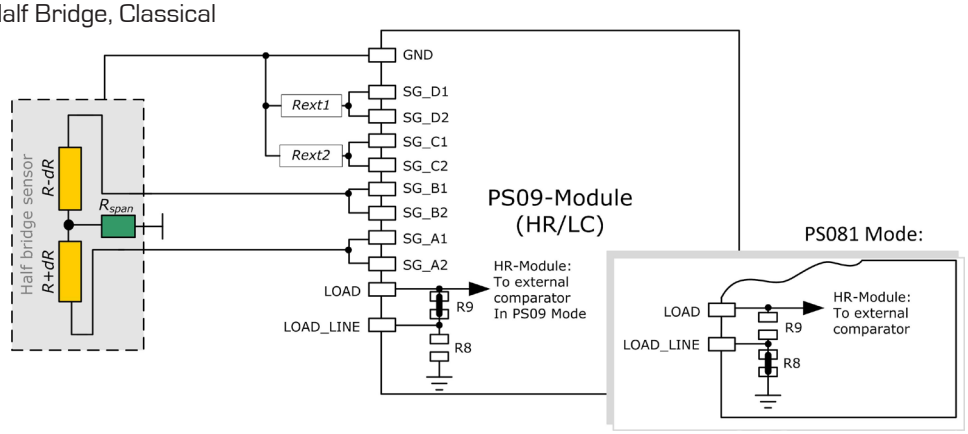
The plug-in module is pre-assembled with Cload = 4 x 100 nF = 400 nF.

2.3.2 Half bridge

A half bridge is sufficient to run the PSØ9 evaluation system. The following picture shows how to connect the half bridge the conventional way. It is basically connected to the pads A and B (SG_A1 and SG_A2, SG_B1 and SG_B2 are shorted). For the gain and temperature measurement, external resistors Rext1 and Rext2 are connected the pads C and D (also shorted).

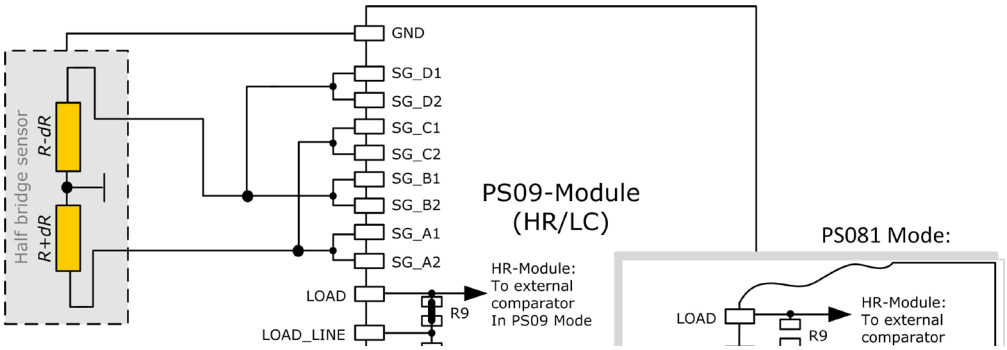
Value of Rext1 = Rext2 = Rsg, the strain gage resistance. For e.g. with respect to Figure 2-3a, for a load cell with Rsg = 350 Ohm and Rspan=45 Ohm, Rext1=Rext2= 350+45 = 395 Ohm

Figure 2-3a: Half Bridge, Classical



An alternative way of connecting a half bridge to PSØ9 is shown below, where the unused ports C and D are connected parallel to Ports A and B respectively. The external resistors are avoided in this connection. The option for using an external comparator is available only on the HR-module.

Figure 2-3b: Half Bridge, Alternative

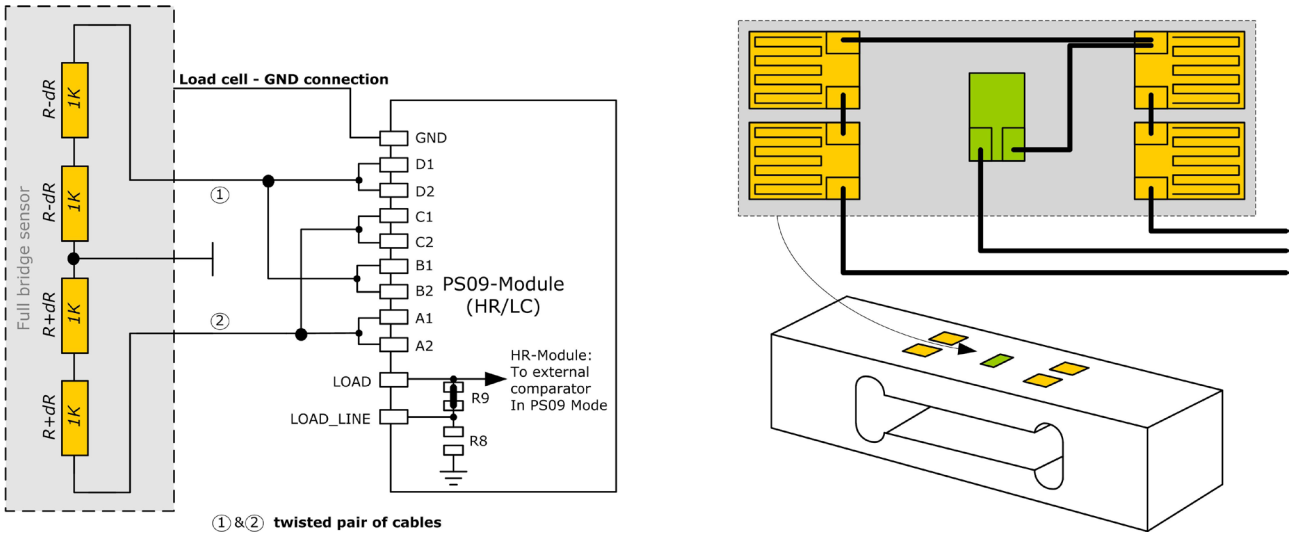


Please note: Both wiring options shown in figure 3a and 3b have to be done externally, i.e. at the solder pad of the module. There are no wiring or placement options on the module itself.

2.3.3 Half bridge connection for Solar Applications

In solar applications the reduction of the current consumption has the highest priority. The unique capability of PICOSTRAIN allows us to modify a full bridge load cell in such a way that it becomes a half bridge with twice the resistance. So with 1 kOhm strain gauges the load cell shows a total resistance of 2 kOhm. The current into the sensor is reduced by a factor of 2. This option is reasonable in case all the strain gauges are on one side of the load cell.

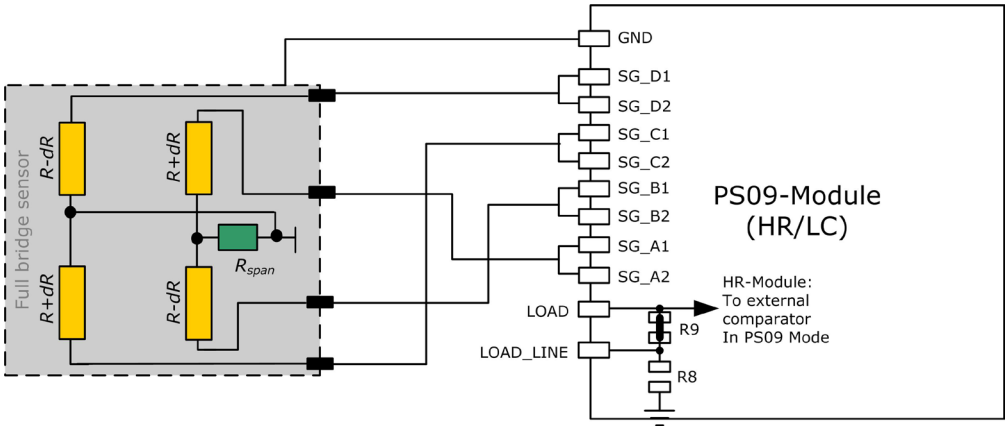
Figure 2-4: Half Bridge, Solar



2.3.4 Full bridge

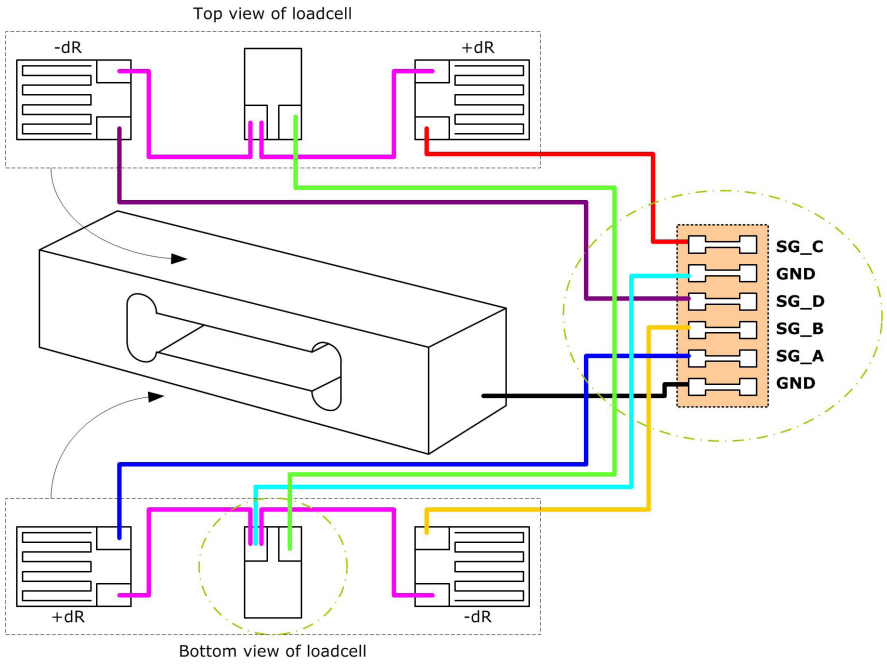
For PICOSTRAIN a full bridge is ideally separated into two half bridges. This wiring can increase the resolution compared to Wheatstone bridges by 0.6 bit.

Figure 2-5: Full Bridge



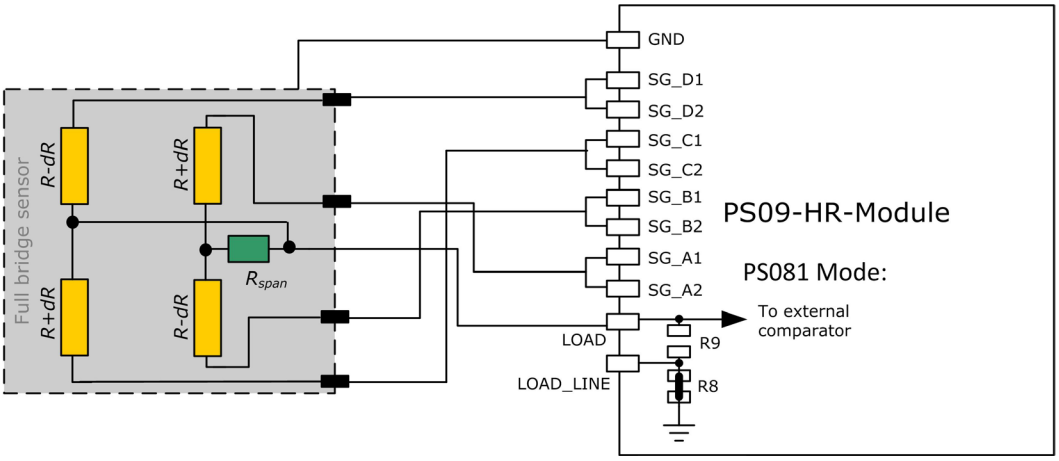
Existing sensors with Wheatstone bridge connection might be adapted just by changing the wiring according to the following picture.

Figure 2-6: Adapted Load Cell Wiring



For certain advantages like resolution, PSRR etc. depending on the application, it might be necessary to use the PSØ81 compatible mode of PSØ9. For details on this mode and its associated advantages, please refer to section 3.3.5 of PSØ9 data sheet.

Figure 2-7: PSØ81-compatible wiring

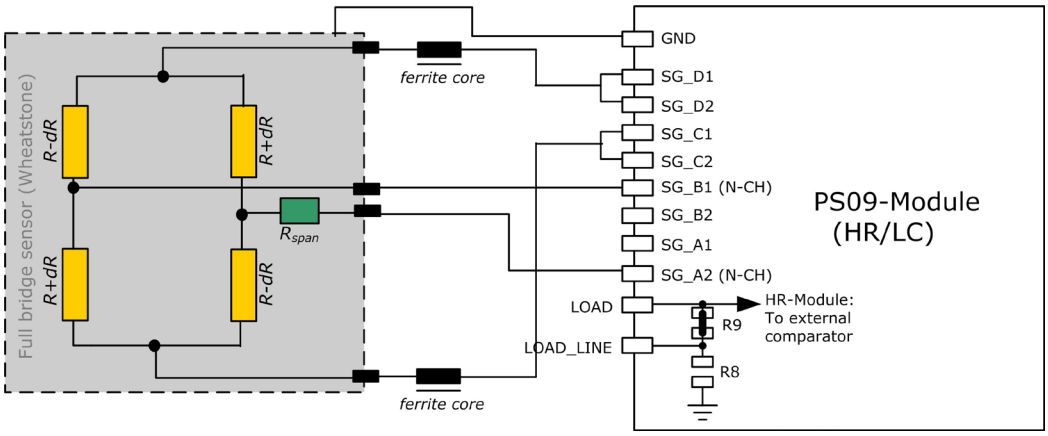


2.3.5 Wheatstone Full Bridge

Existing load cells in Wheatstone configuration can be connected to the module without any modification as long as they have only one or no compensation resistor. In case the bridge has two compensation resistors one of those needs to be shortened.

In general Wheastone wiring will end in 0.6 bit less resolution compared to PICOSTRAIN wiring. It might be reasonable to use Wheatstone bridges in case of cables to the sensor longer than 0.5 m. The following figure shows the connection of the Wheatstone bridge.

Figure 2-8: Wheatstone Bridge



The PICOSTRAIN measurement principle is based on measuring the discharge time of a capacitor. For this reason the correct size and material of the capacitors is significant to achieve best measuring results. For Wheatstone, the discharging time is furthermore reduced by the factor of 0.7, The following formula can be used to calculate the discharging capacitance.

$$\tau = 0.7 * 0.75 * R * C = 60 \text{ to } 110\ \mu s.$$

As material we recommend C0G or CFCAP (Multilayer ceramic from Taiyo-Yuden), X7R capacitors can be used, too, but will show some minor loss in temperature stability.

The recommend values for Wheatstone mode are:

Rsg = 350 Ohm →Cload = 300 nF to 400 nF

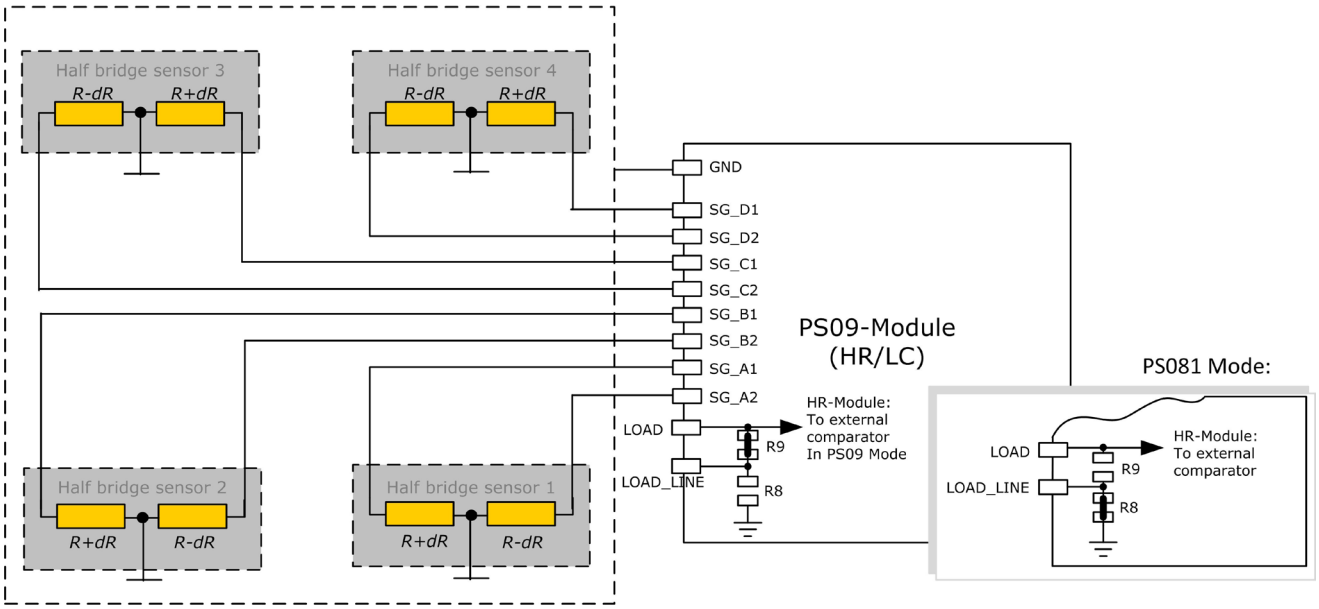
Rsg = 1000 Ohm →Cload = 100 nF to 200 nF

Please note: In Wheatstone mode the operation via SPI interface is recommended. If IIC is to be used, please operate in Single Conversion Mode. See also the bug report in the PS09 data sheet.

2.3.6 Quattro mode

In quattro mode the PSØ9 measures 4 half bridges. The 4 half bridges are measured independently and the gain of each half bridge can be corrected separately. Typical applications are bathroom-scales, baby or platform scales.

Figure 2-9: Connecting four half bridges or two full bridges in One sense mode



3 Motherboard

The motherboard connects to the PICOPROG programmer. It serves the various power options. It holds the LCD panel. The 9 push buttons (resistive keys) and 4 capacitive keys can be used in stand-alone operation.

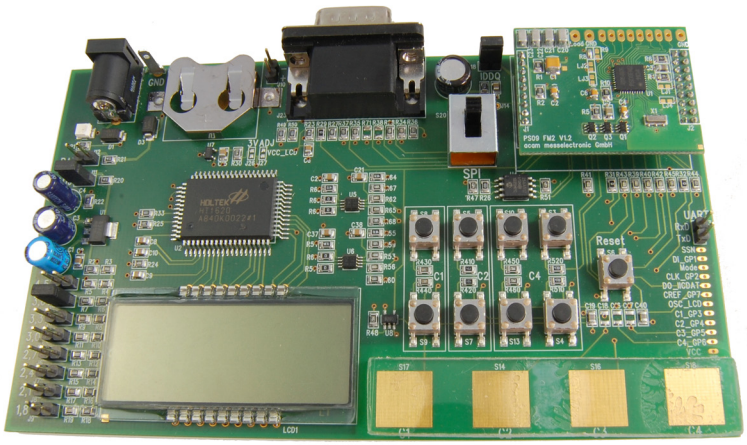
The jumpers for power select can also be used to measure the current consumption of the system.

3.1 LCD

The LCD has the following specification:

Duty ¼, Bias 1/3, Operating voltage 2.5V, Operating temperature 0°C to 50°C.

Figure 3-1: Motherboard

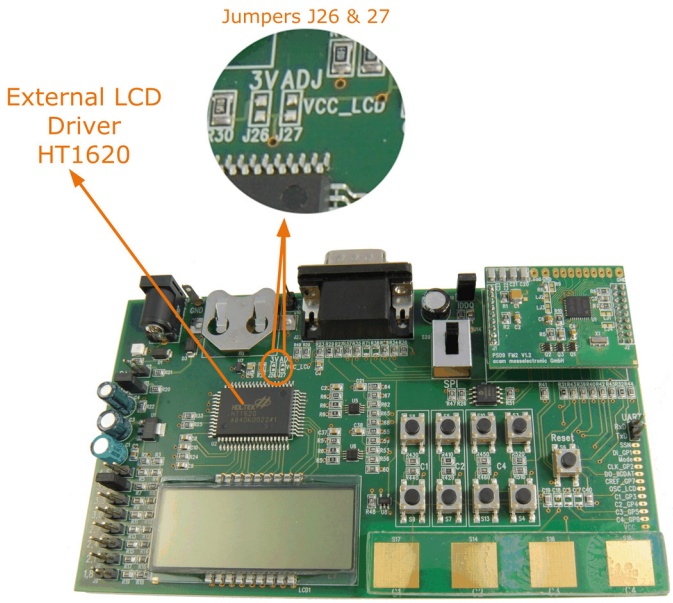


3.2 External LCD driver

PSØ9 offers the possibility to support an external LCD driver. Therefore a simplified SPI master mode is programmed in the PSØ9, especially adapted to Holtek HT1620 LCD driver. Three GPIOs of the PSØ9 are used to realize the SPI master interface to connect to the Holtek driver. The PSØ9 can generate a configurable 32 kHz clock needed to drive the HT1620 driver, thus avoiding the necessity of an external crystal oscillator for the LCD Driver. One GPIO is additionally used to for this clock.

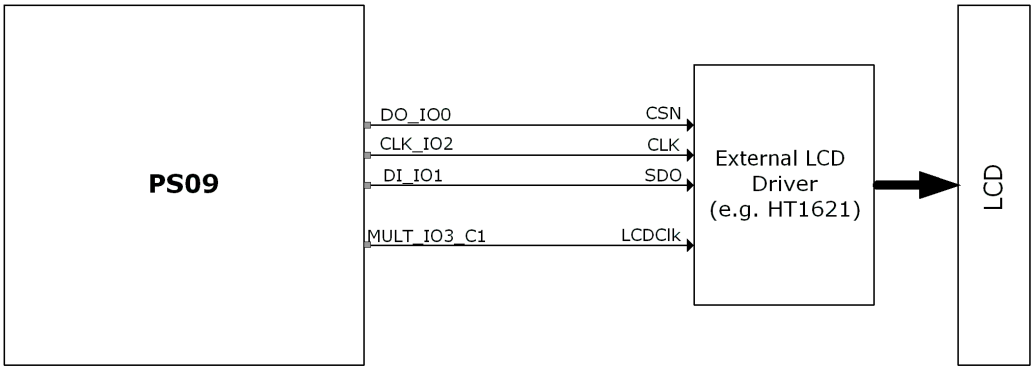
The LCD driver on the PSØ9 Motherboard can be powered by the output of a voltage regulator (fixed voltage of 3V). Optionally the LCD driver can be powered by the voltage selected by the on-board jumpers on the motherboard (voltage adjustable). The appropriate jumper (J26 (fixed) or J27 (adjustable)) has to be soldered in order to select the source of the LCD's power supply.

Figure 3-2: External LCD Driver



The following picture shows the connection of an external LCD driver circuit:

Figure 3-3: External LCD Driver Wiring



A flowchart showing the general sequence to program the PSØ9 in order to operate the external LCD driver is illustrated in Section 4.8 of the PSØ9 Data sheet. The flowchart is however based on the idea that GPIOs 5, 6, 7 would be used as the SPI communication lines. The PSØ9 EVA board supports communication to the Holtek driver only via GPIOs 0, 1, 2. Sample programs which use the LCD for display, along with the appropriate header files that are specific to the Holtek driver, HT1620 are available as part of the PSØ9 Assembler examples.

4 Load cell

The evaluation system is shipped with a ready made demonstration scale connected to the high resolution module.

The load cell is model CZL601SE-10kg from Hua Lan Hai (http://www.chinesesensor.com/Single-point_Load_Cell.html).

4.1 Technical Specification

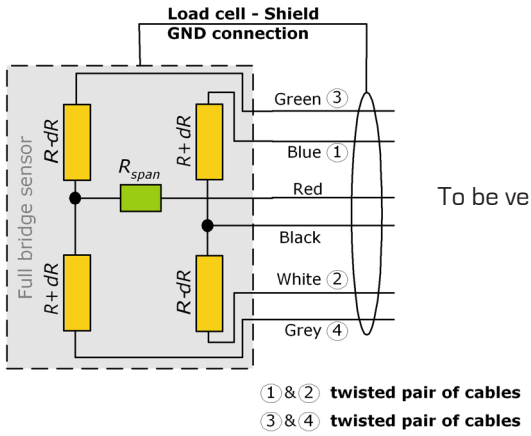
Table 4-1: Technical Spelcification

Maximum load	kg	10
Comprehensive error C2	%F.S.	0.02
Rated output	mV/V	typ. 1.85
Non-linearity	%F.S.	0.03
Hysteresis	%F.S.	0.03
Repeatability	%F.S.	0.02
Creep	%F.S./30min.	0.02
Resistance	Ohm	350 ± 5
Compensation resistor	Ohm	42 ± 5
Compensated temperature range	°C	-10 to +40
Operating temperature range	°C	-35 to +65
Safe overload	%F.S.	120
Ultimate overload	%F.S.	150

Mechanical dimensions:	
Base plate	200 mm x 100 mm
Weighing plate	90 mm x 90 mm
Total height	70 mm

4.2 Wiring diagram

Figure 4-2: Wiring diagram (to be verified)



5 Evaluation Software

5.1 Installing the PSØ9-EVA Software

The PSØ9-EVA software runs under the following operating systems

- Windows 2000
- Windows XP
- Windows Vista (please use the software/drivers for Windows 7)
- Windows 7

Please follow the described procedure to install the software and driver:

NOTE: Ensure that the Picoprog V2.0 programmer is disconnected before starting the procedure.

1. Install the device drivers by running setup.exe from Driver/PicoProg v2 Driver Installer Stand-Alone. Select the shown default paths and continue the setup procedure.
2. At the end of step 1, you will be asked in a separate window (batch-file) to connect the Picoprog V2.0 programmer. Please insert the PicoProg to your computer.
3. The driver is installed; a windows message may pop up asking you to install the driver though it is not signed. Please install it anyway.
4. The batch file asks you to plug in the programmer and it is now enumerated. Unplug and insert the Picoprog programmer.
5. Install the PSØ9 Assembler software from Software\PSØ9-Assembler-Software\XP-WIN7\Volume2-0-xx\setup.exe from the CD.
6. If required perform a system restart.
7. The assembler software is now ready to use. Open the PSØ9 Assembler software, the respective firmware is downloaded into the programmer automatically and the LED on the Picoprog programmer is illuminated.
8. You can test to find the programmer on the 'Download' page and press 'Get Device Info'.
9. Install the evaluation software now from Software\PSØ9-Evaluation-Software\Setup.exe
10. Once the software is installed, launch the application from the start menu. The software opens and a pop up window asks to select a firmware for Picoprog.
11. Select the hex-file (firmware) for Picoprog in the folder "data" The hex-file itself is named PSØ9_FWxx.hex
12. If everything is correctly installed, the USB identifier must be USB:: 0x194E:: 0x100F::NI-VISA.
13. Please confirm connection by clicking the button 'Verify Interface'. 'OK' should be shown in the pop-up window.

Optionally you can see a video-tutorial about the PS08 Assembler Software (which structurally the same as the PSØ9 Assembler) by watching the video from:

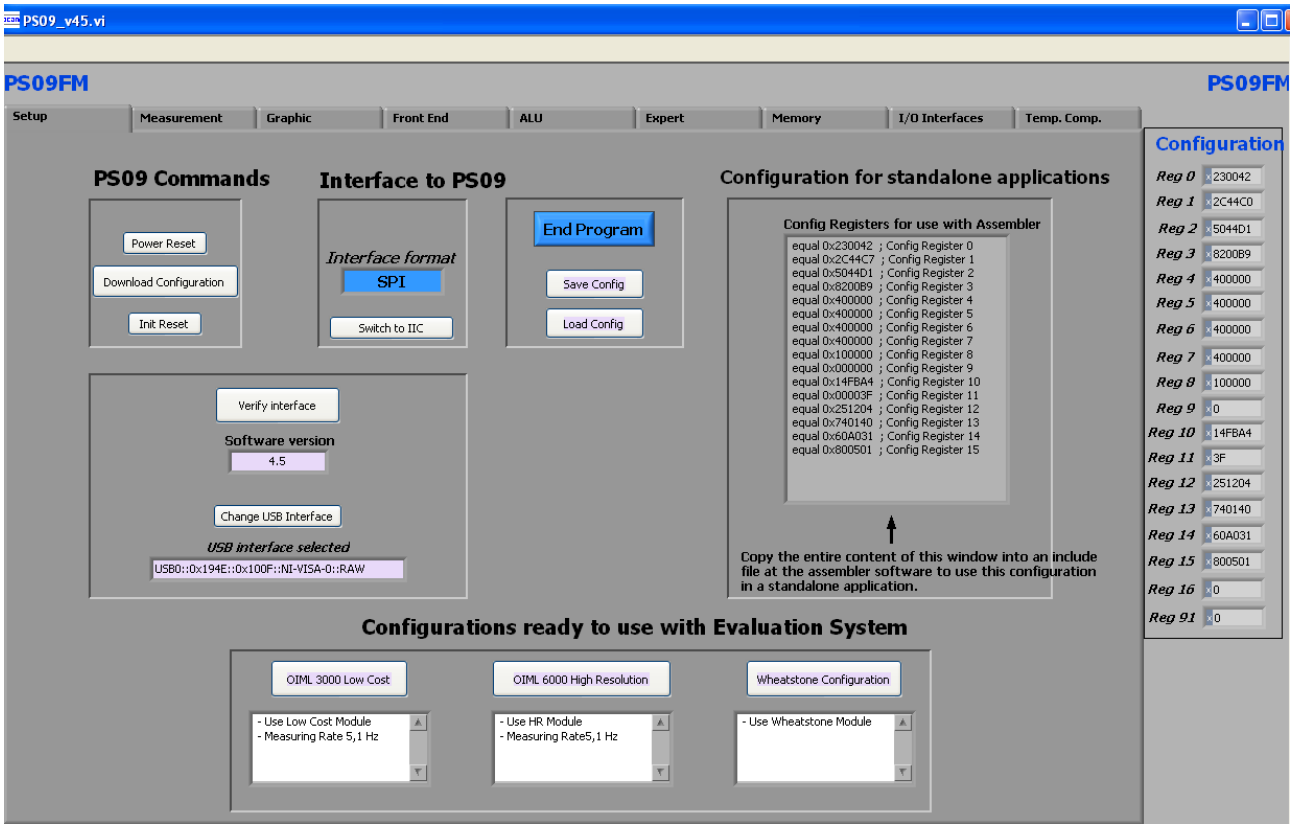
→ ASM-Screencast.exe in PS08-Assembler-Introduction\Flash folder.

5.2 Running the Evaluation Software

The software comes up with the following window:

5.2.1 Setup Page

Figure 5-1: Setup Page



Power Reset	Power reset of evaluation board
Download Configuration	Download the current configuration into the PSØ9
Init reset	Initialization of PSØ9 (keeps the configuration)
Verify Interface	Verifies the communication path between the PSØ9 and the PC
Switch to SPI	Establishes the communication mode between the Picoprog and the PSØ9 to SPI mode (Note: By default the com munication mode set in the evaluation software is SPI)
Switch to IIC	Establishes the communication mode between the Picoprog and the PSØ9 to IIC mode

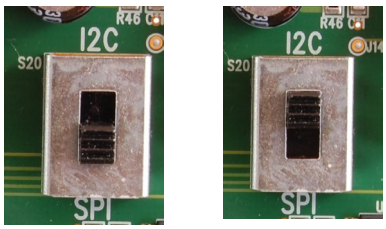


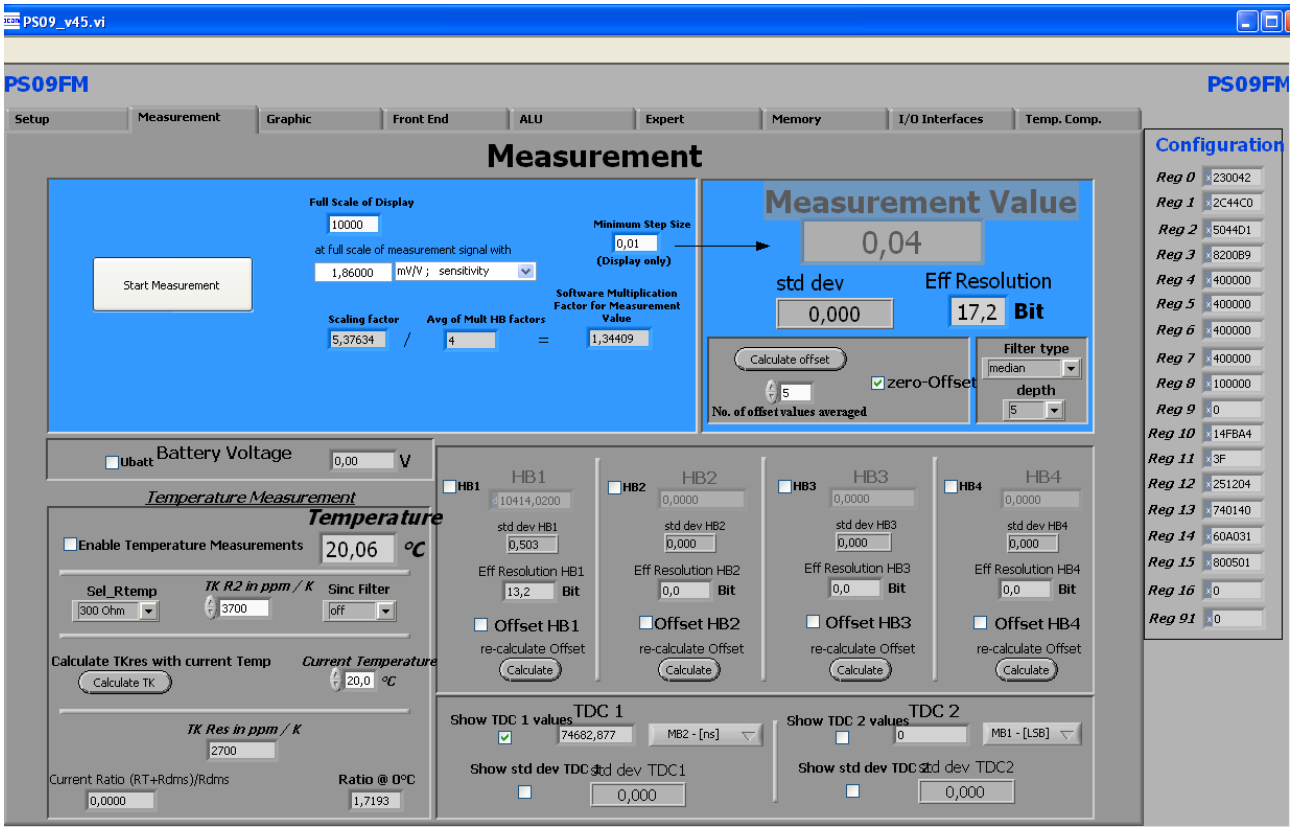
Figure 5-2: Switch IIC/SPI

Save Config	Save actual PSØ9 configuration to PC
Load Config	Load existing configuration from PC

1. It is recommended to start the PSØ9 evaluation by using the Ready-to-use configurations.
2. After loading a configuration please press ‚Power Reset' → ‚Download Configuration' → ‚Init Reset'
3. The communication to the chip is verified by pressing 'Verify Interface', the result is a pop up window with the software version, firmware version and the status of the PSØ9 communication interface.
4. Afterwards switch to the ‚Measurement' tab and press → ‚Start Measurement'

5.2.2 Measurement Page

Figure 5-3; Measurement Page

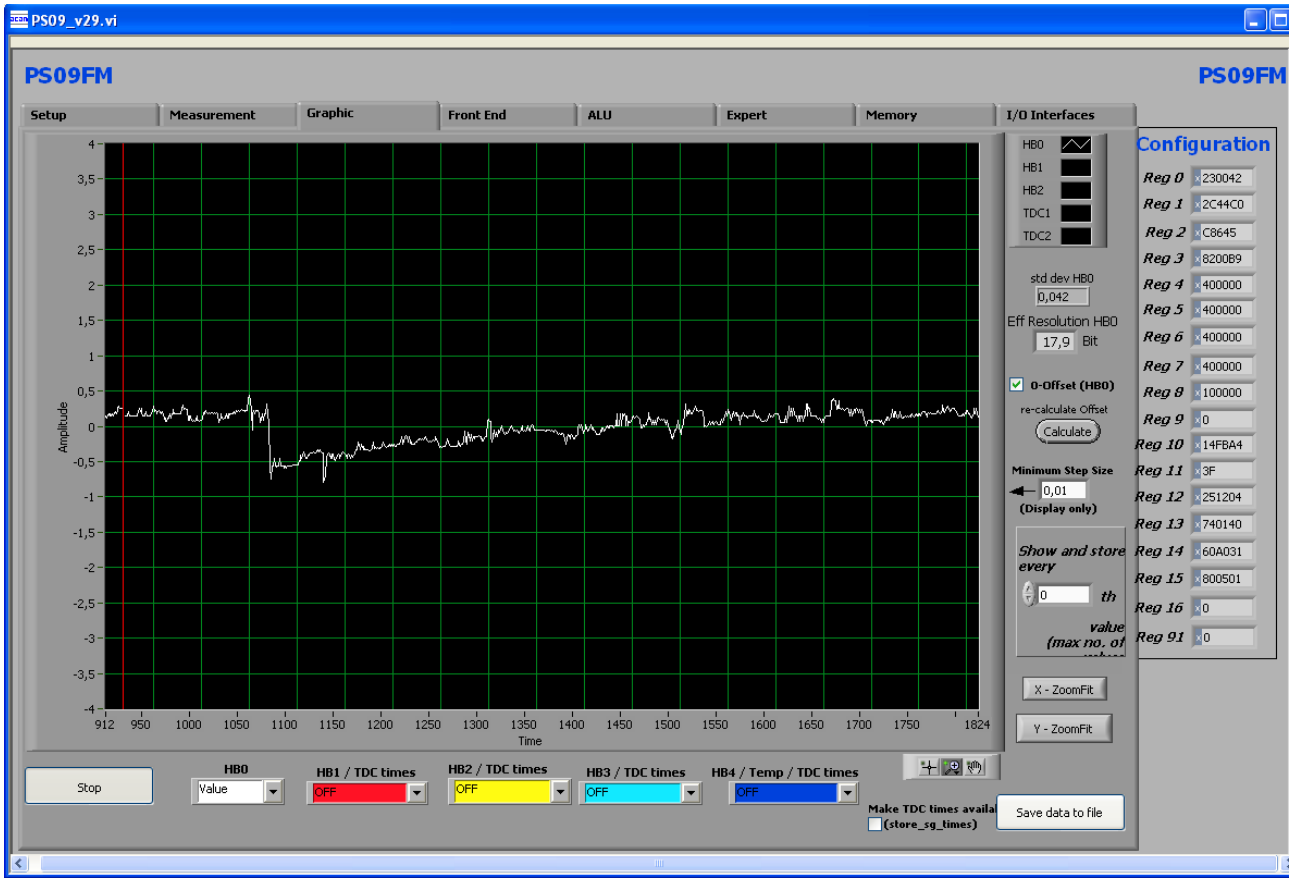


Full Scale of Display	Value that shall be displayed at maximum load at full scale of measurement...
	Sensitivity of sensor, output at maximum load
Minimum step size	Minimum step size of displayed result
Measurement value	Display of HB0 result using the software filters set under „Filter".
Eff. Resolution	Effective resolution with respect to maximum output
Filter	Selection of various software filters like SINC (rolling average) and Median (non-linear filter).

depth	Depth of the filter
re-calculate Offset	Software recalculates the offset, sets back the display to 0.
Ubat	Include voltage measurement, display in V.
Temperature Measurement	To enable the temperature measurement to be performed on chip.
Sel_Rtemp	Select the value of the internal temperature measurement resistance to be used for measurement.
HB1 ... HB4	Display the results of the half bridges (works only if Single Conversion Mode is configured)
Show TDC1 values	Shows the discharge time
Show TDC2 values	Shows resolution of TDC

5.2.3 Graphic Page

Figure 5-4: Graphic Page

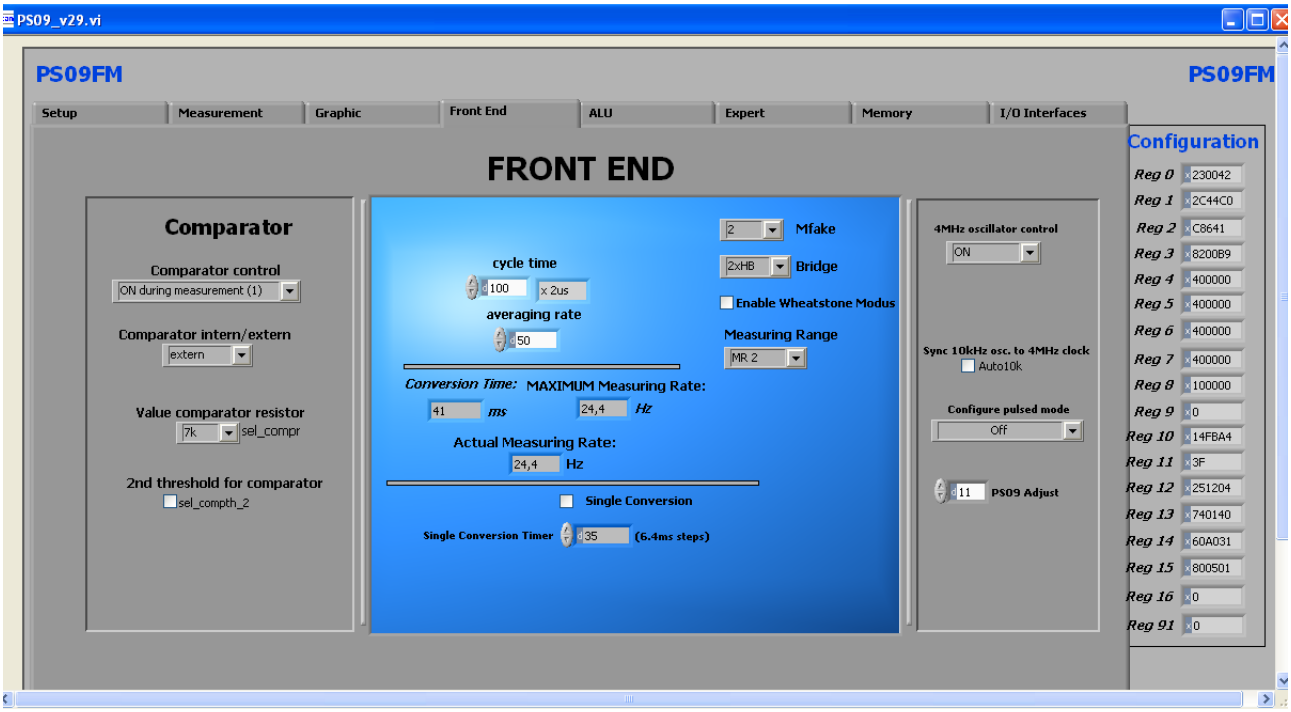


Graphical display of the results. The consolidated result HB0 as well as the separate half bridge results can be displayed. Additionally, Temperature can also be graphically displayed.

Pressing the → ‚Save data to file' button stores the data in a text file. The size is limited to 32k values. For long term drift investigations it is possible to store not each value. The number between values to be stored can be set.

5.2.4 Front-End Page

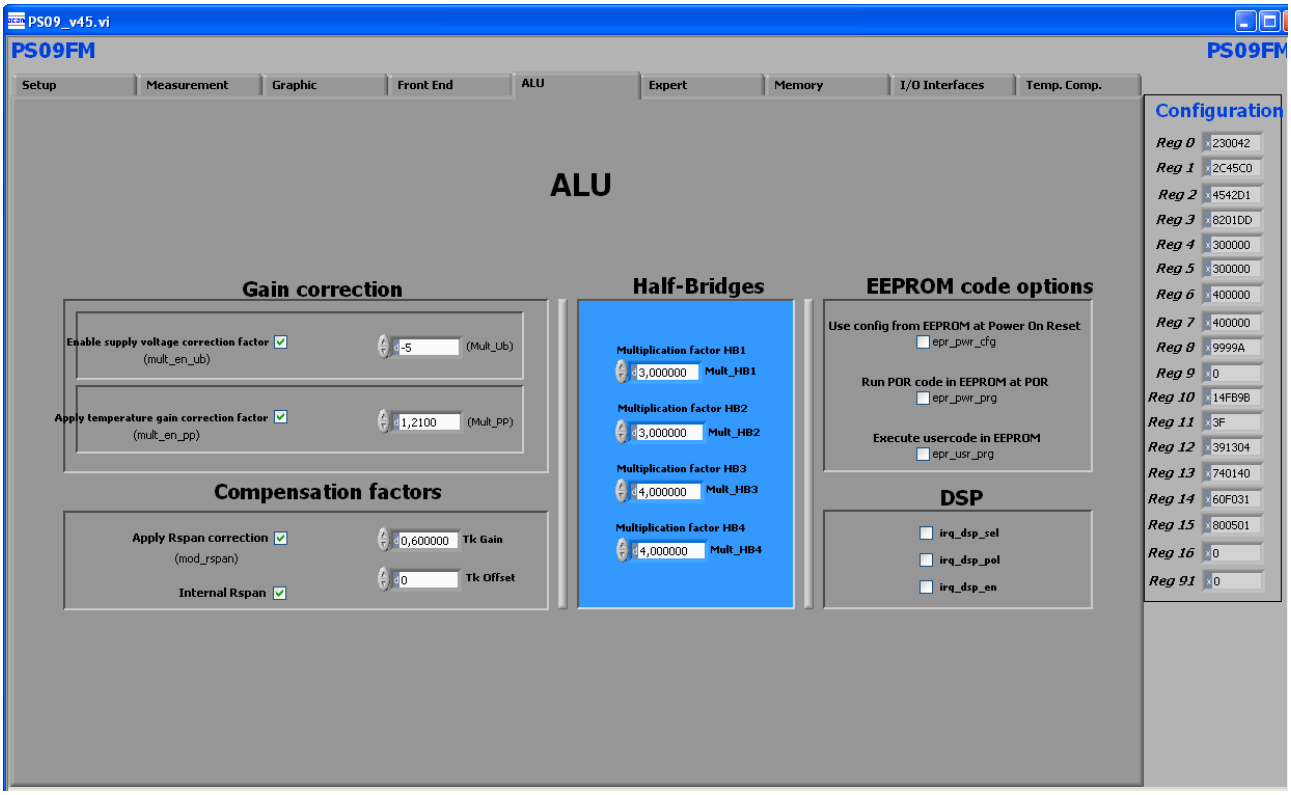
Figure 5-5: Front-End Page



Comparator control	= con_comp: Sets the switch on behaviour of the comparator
Comparator intern/external	= sel_compint: Selection between internal comparator (LC module) and external bipolar comparator (High resolution module)
Comparator resistor value	= sel_compr: Select comparator working resistor
Sel_cmp_thr2	= Selects the second threshold for the comparator
Cycle time	= cytime: Set the cycle time (see section 9.2.3 of PSØ9 datasheet)
Averaging rate	= avrate: Set the internal averaging rate
Single conversion	= single_conversion: Selects single conversion modes. The timer defines the time interval between conversions.
Mfake	= mfake: Sets number of fake measurements.
Bridge	= bridge: Selects the number of half bridges
Enable Wheatstone	= en_wheatstone: Selects Wheatstone mode
Measurement range	= messb2: Standard is measurement range 2. Option to select range 1.
4 MHz oscillator control	= sel_start_osz: Sets the switch on control for the 4 MHz oscillator

5.2.5 ALU Page

Figure 5-6: ALU Page



Settings for gain and offset correction.

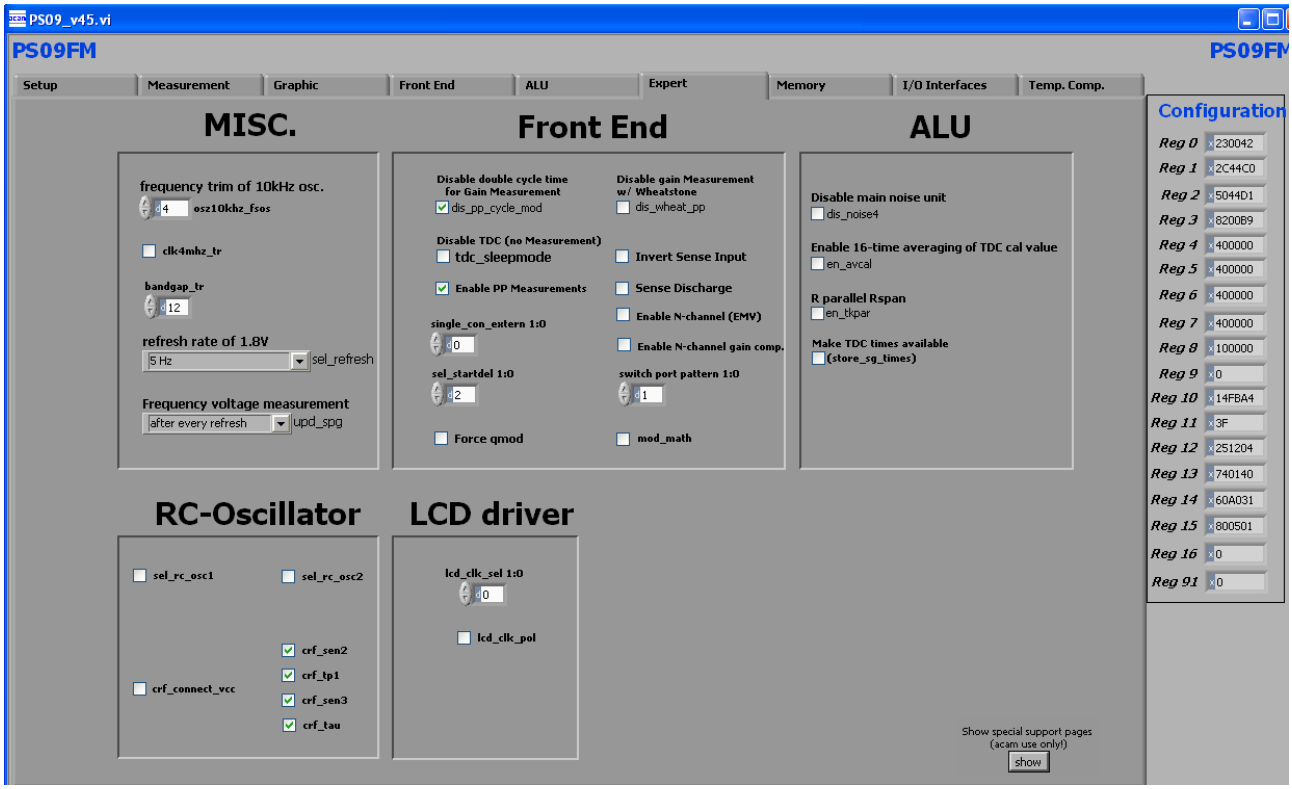
Half bridges: Setting the 4 independent multiplication factors for the 4 half bridge results.

EEPROM Code options: Enables to select when the user code has to be executed and the configuration to be used on reset.

DSP: These options enable to enable/disable the interrupt, select the interrupt polarity and select the pin for an external interrupt.

5.2.6 Expert Page

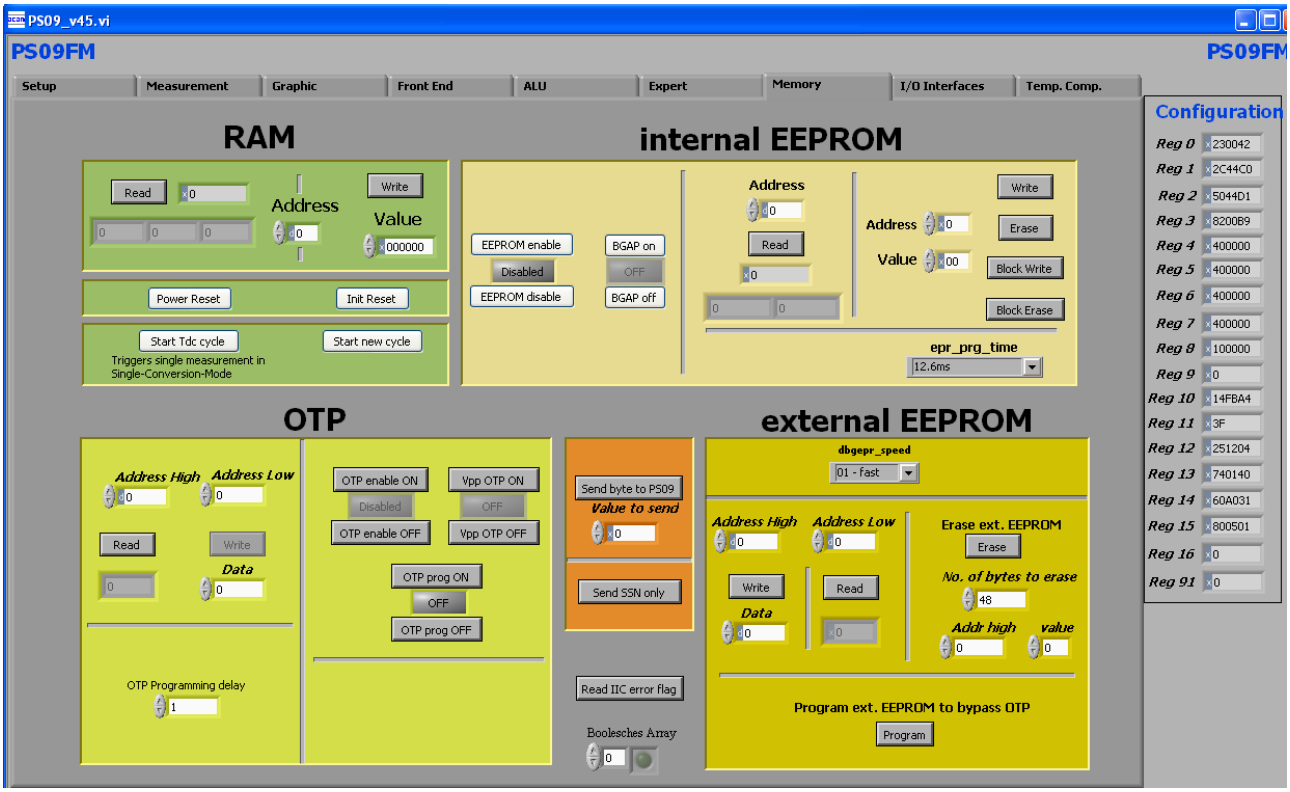
Figure 5-7: Expert Page



This page refers to expert setting only. Please change settings only after getting an introduction by an acam engineer.

5.2.7 Memory Page

Figure 5-8: Memory Page



RAM: Shows the contents of the RAM address selected

The RAM can be written and read directly from this page.

Internal EEPROM: Shows the content of the internal EEPROM at the selected address. Individual addresses can be erased, written and read. The entire EEPROM can be erased and written in one go also.

OTP: Shows the content of the OTP at the selected address. There are buttons to switch on the OTP, enable the VPP to the OTP and the PROG line to the OTP, too; these are required to write to the OTP from the evaluation software.

Caution: naturally the OTP can only be written once (or incrementally several times). That means, correction or re-writing of cells is not possible!

External EEPROM: Shows the content of the external EEPROM used during program development. All the addresses can be read and written. It is possible to erase a selected number of bytes.

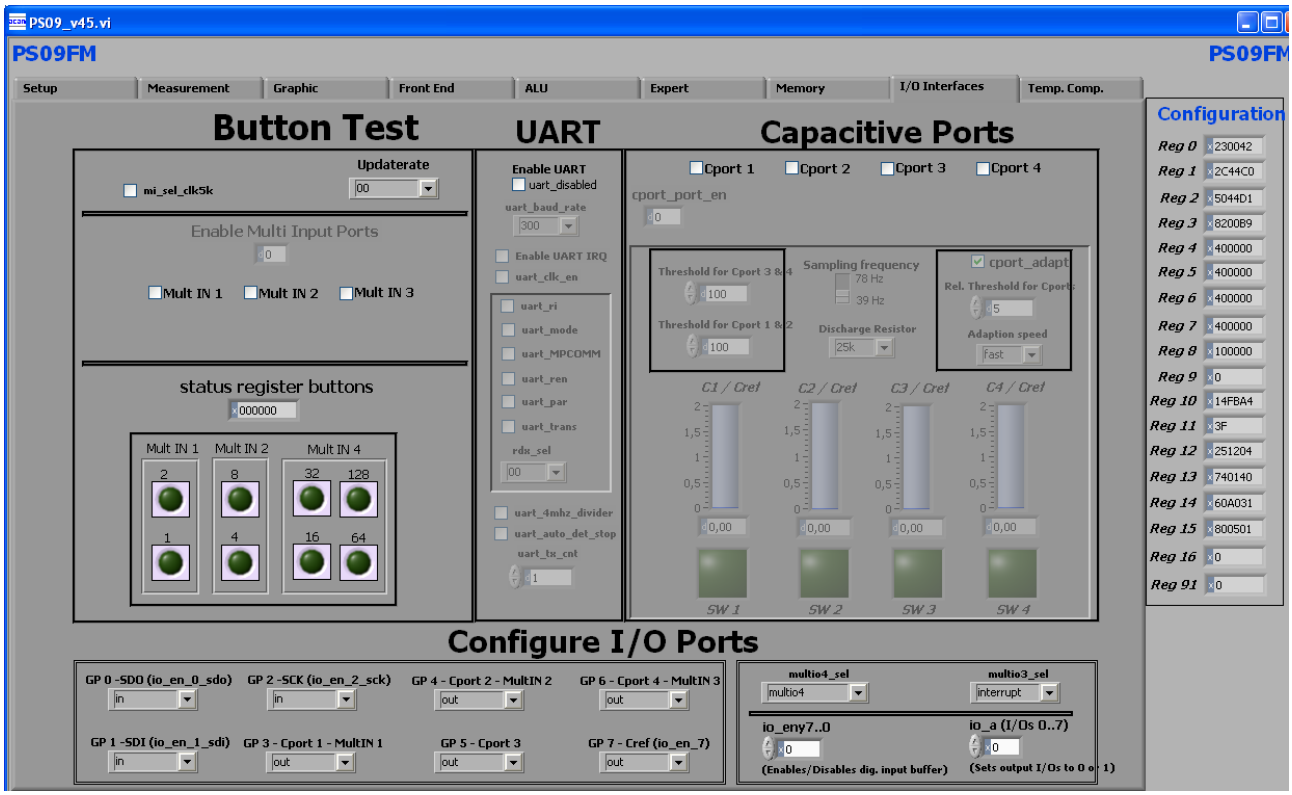
Start New Cycle: Start manually a new measurement.

Send byte to PSØ9: Sends the byte to the PSØ9 through SPI/IIC interface

Send SSN only: Toggles the CSN of the SPI interface to the PSØ9.

5.2.8 Interfaces Page

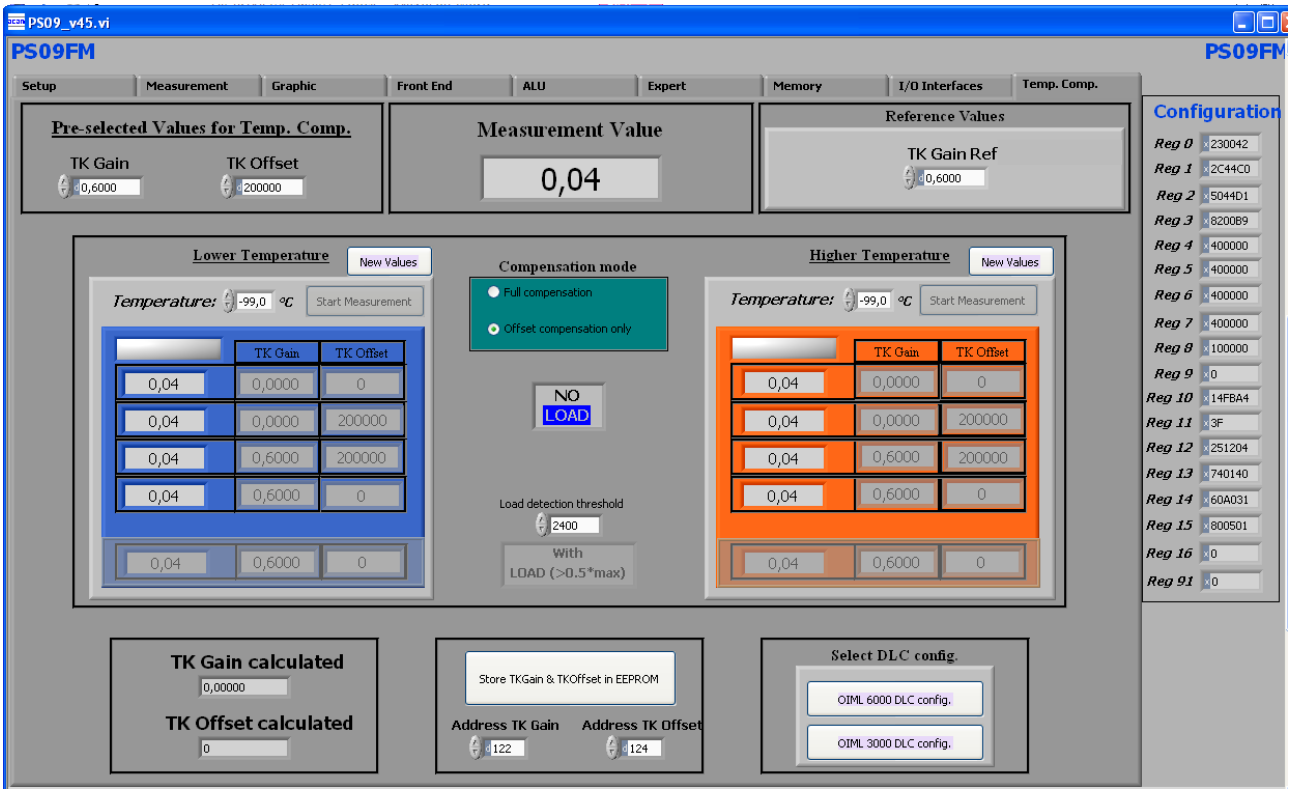
Figure 5-9: Interfaces Page



Button test:	Allows configuring the multi-input keys. There are only 8 out of possible 24 multi-input keys realized in hardware on the motherboard. When the keys are configured and when the buttons on the motherboard are pressed, the keys displayed in the software light up (like an LED).
Capacitive keys:	Allows configuring the 4 capacitive ports. Display shows the status of the 4 capacitive ports. When the capacitive keys are configured correctly, then the display in the software changes accordingly when the keys on the motherboard are touched or operated.
UART:	Allows enabling and configuring the UART.
Configure I/O ports:	Allows configuring each of the 8 GPIOs as input or output with various options. When output, allows to set the value on the output pin. When input, the digital input buffers ought to be enabled here. Additionally the multi- functional diagnostic pins (MULTIO3 & 4) can also be configured here. These configurations must also be done respectively when using the multi-input or capacitive keys or the UART.

5.2.9 Temperature Compensation Page

Figure 5-10: Temperature Compensation Page



This page contains all the settings that are used for temperature compensation of a load cell system connected to the PS09. Either full compensation can be performed or only Offset compensation can also be done.

Pre-selected values for Temp. Comp:	Arbitrary pre-selected values of TkGain and TkOffset before beginning the compensation procedure are set in this part of the page.
Reference values:	The parameters pertaining to the temperature coefficient of the internal Rspan are set here.
Lower temperature / higher temperature:	The measurement values taken at lower temperature and higher temperature with load and with NO load are all displayed here. The measurements are performed automatically after pressing 'Start Measurement'. After the compensation process, the newly calculated values of TkGain and TkOffset are shown at the bottom of the page.
Select DLC config.	Configurations for doing the adjustment procedure according to OIML 3000 or OIML 6000. The configuration should be selected before starting with the compensation procedure.

Store TkGain and TkOffset in EEPROM:	The values of the TkGain and TkOffset after the temperature compensation can be stored in data EEPROM at user selectable addresses.

The process of the temperature compensation with the evaluation software is practically shown in a dedicated screencast (a screencast is a basically an instruction video). It contains the theory behind the PICOSTRIAN temperature compensation as well as its practical application in the software. To download the screencast please go to: http://www.acam.de/download-center/picostrain

6 Assembler Software

Features:

Platform:	Windows 2000, XP, Vista (32-Bit), Windows-7 (32 and 64)
Type of Assembler:	Single Path Assembler
Source-File:	*.asm, according to the PSØ9 assembler syntax
Target-File:	Output. hex, downloadable Hex-File
Instruction:	<command> [<parameter1>, <parameter2>, <parameter3>] For example: nop or incr z or add x,y or gotoBitC x, 5, case1
Comment(s):	Single line comment: ; Multi line comment: <COMMENT> ... <ENDCOMMENT>
Includes:	Including files is possible with: #include „myfile.h“ No limit to the number of include files
Constants:	Constants can be defined with: CONST myConst8 15 The constants have to be declared before they are used for the first time.
Number formatting:	Numbers can be written in decimal or hexadecimal notation e.g.: add x,20 or add x, 0x14
Jump labels:	To jump within the code, you can use jump labels together with goto. E.g.: goto case1 case1: move x,y
Addressing:	Addressing is automatically done by the assembler. Please note, that you must provide at least 48 bytes of configuration data, maximum are 8k bytes of user programmable space.
Subroutines:	Subroutines can be executed with the special opcode ‚jsub‘ and ‚jsubret‘.

6.1 Installing the Assembler Software

Insert the CD-ROM. Run Software\PSØ9-Assembler-Software\Volume2-0-3\setup.exe to install the program. Re-start the computer if required. After the installation you will find in the START menu an item „PSØ9 Assembler“. Run this to start the assembler program.

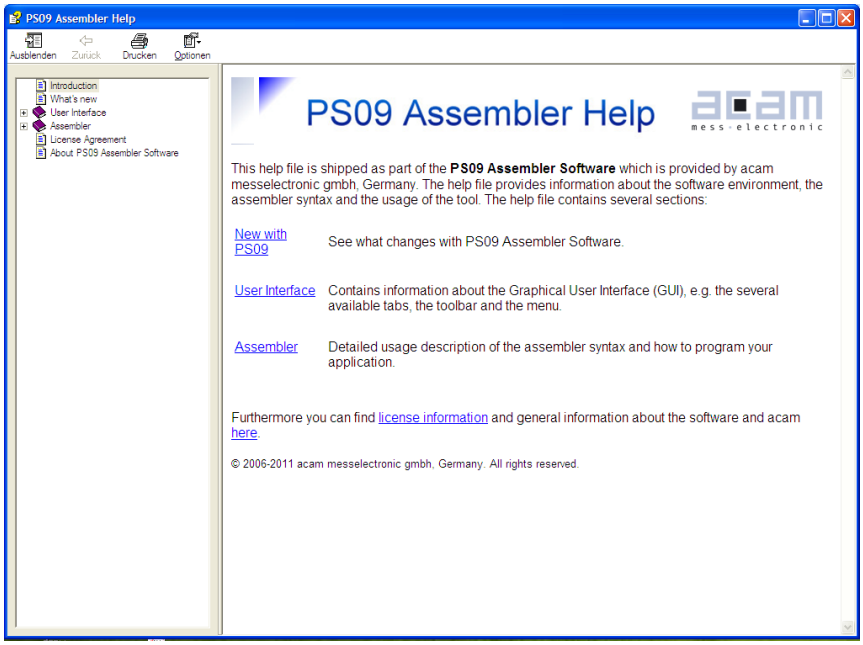
6.2 Running the Assembler

In the START menu there is an item „PSØ9 Assembler“. Run this to start the assembler program.

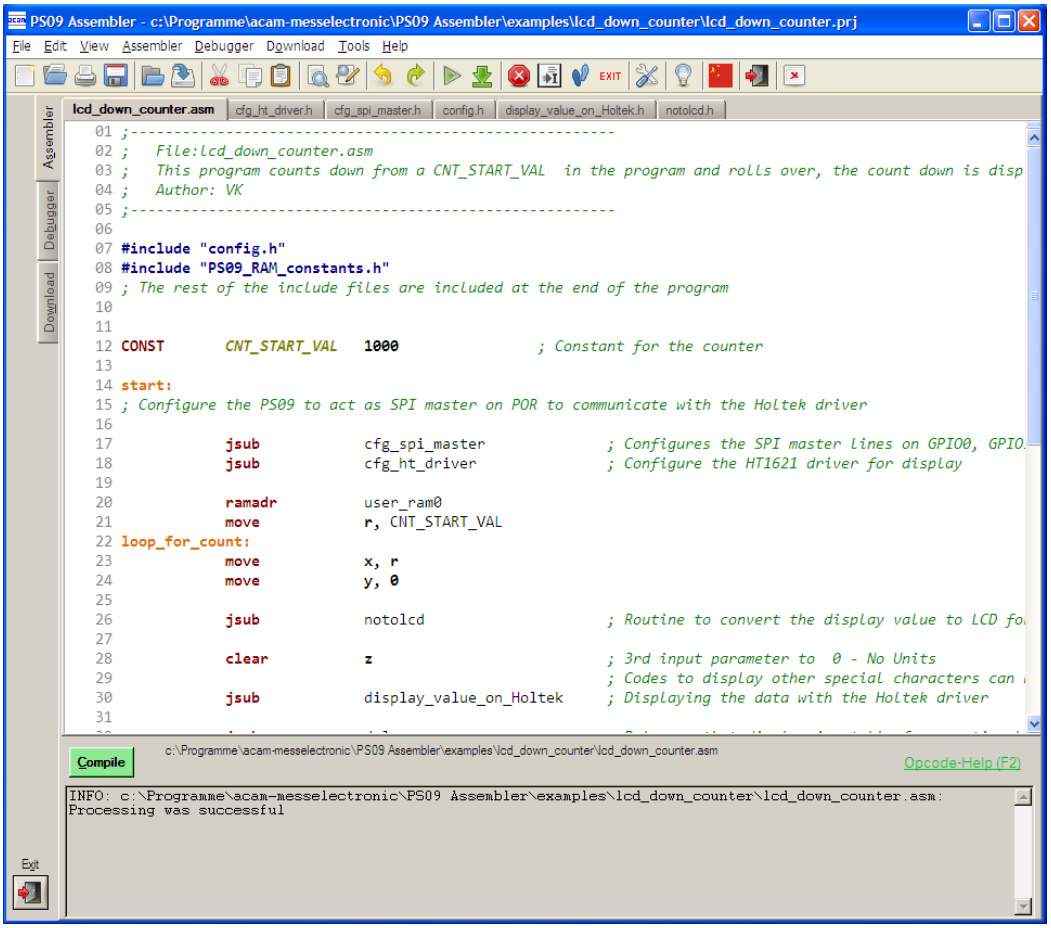
6.2.1 Assembler Online Help

The following sections give just a short description of the assembler program. For a detailed description of the assembler software please use the online help of the program. The online help can be opened from the Menu or by pressing the F1 button.

The following help window opens up:



6.2.2 Assembler Tab Page



The 'Assembler' tab page is the main window to open all the source code and include files. It consists of a status message window at the bottom and buttons to operate the assembler. By clicking the green link labeled ‚Opcode-Help (F2)'an integrated online help pops up which provides detailed information about the available op codes.

The controls in detail:

Source code window:	This window is used as the source code and include files editor for the assembler listings (*.asm & *.h files). This is a pure editor window; any number of tabs can be opened in this window. This page allows the user to display and edit any file. The files can be manually opened, saved and closed. The location of the file displayed in the window currently is displayed next to the 'Compile' button. Additional features are line numbering, syntax highlighting and integrated comment handling. Standard editor tasks like cut, copy & paste or find & replace, etc. can be performed. You can modify the settings for the editor via the tool settings dialog. To know more about 'Include' files and how to use them, see Section 6.2.2.1 below.
Assembler button:	The assembler button to compile the source code along with the respective header files is: Compile. On pressing the Compile button the source file is compiled to a downloadable hex code. If the assembly run was successful a message is displayed accordingly. If the compiling fails, an error message occurs. If more than one source files are currently open, then the *.asm that will be compiled on pressing the Compile Button, has the filename in bold letters on the tab. It is the firstly opened .asm file in the window that is always compiled.
Save Project button:	When a source code (*.asm) file and all the respective include (*.h) files are open in the Assembler tab, then all the files can be combined and saved in a project file (*.prj), by pressing the Save Project button. The .prj file always takes the name of the *.asm file. For e.g. A source code file sample.asm uses 'include' files include1.h and include2.h. Then the corresponding prj file would be saved by the name sample.prj. Once saved, the *.prj file can be opened from the menu File -> Open project. This automatically closes all currently open files and opens all the files (*.asm and *.h) belonging to the project in one go. The *.prj file can also be opened and edited in this Window. For e.g. A source code file sample.asm uses 'include' files include1.h and include2.h. Then the corresponding prj file would be saved by the name sample.prj. Once saved, the *.prj file can be opened from the menu File -> Open project. This automatically closes all currently open files and opens all the files (*.asm and *.h) belonging to the project in one go. The *.prj file can also be opened and edited in this Window. Note1: The .asm and .h files can be in different folders or locations. The *.prj file stores the location of the file while saving the project.

	Note2: When the PS09 Assembler software is closed, a .prj project file is automatically generated with all the files that are currently open in the window. The name of this project file will be that of the main source code file open with a .prj extension. This .prj file will be used to open all these files again when the PS09 Assembler software is started again.
Open Project button:	An already saved project (with a *.prj file), can be opened using this button.
Status message window:	In this window there are the output messages of the assembler displayed. In case the assembly process was successful, a corresponding message appears with the path of the file that was compiled. If an error occurs while assembling, an error message appears together with the line number and the file name in which the error occurred. An error in any of the source code (*.asm) file or "include" (*.h) files is identified and intimated with the path.
Opcode-Help:	Between the source code window and the status message window there can be found a green link labeled ‚Op code-Help (F2)'. By clicking this link another window pops up. The window contains the op code online help. Every available op code is explained there in detail. The additional windows are based on the integrated Microsoft Windows ® Help system and can be operated separately. (Windows is a trademark of Microsoft Corporation)

Note: When you point on the icon in the menu bar a tool tip describing the function of the icon is displayed.

Running the assembler in order to compile your source file is the first step when using the PSØ9 Assembler Software. Further steps like downloading the hex file or using the debugger are based on a successful assembler run. In case any errors occur please correct your source code and run the assembler again.

An integrated 'examples' folder provides some examples related to the assembler structure itself as well as specific applications like displaying something on the LCD on key press, configuration of multi input keys, capacitive keys or simple weigh scale programs. The user can also add his own examples to the 'Examples' folder with the respective prj file. The prj file can be written manually (See format of existing prj files in the Examples folder) or can be allowed to be generated by the assembler with the Save Project button. The Search-Examples dialog box can be accessed via the Help menu or by pressing F4 or by clicking on the 'Bulb' icon on the top. Only *.prj project files can be opened via the Search Examples option. To open individual files, use the File -> Open option.

6.2.2.1 'Include' files:

The basic idea of supporting 'include' files is to provide more modularity and flexibility within your coding. That means, instead of packing all needed source code to a single assembler listing (*.asm files) you can roll out some pieces of code to the include files (also called header files with extension *.h). The advantage of this method is, that the readability of your code will be improved and that once written parts of code can be reused (e.g. configuration information).

To include parts of code by using a include file there are two steps needed.

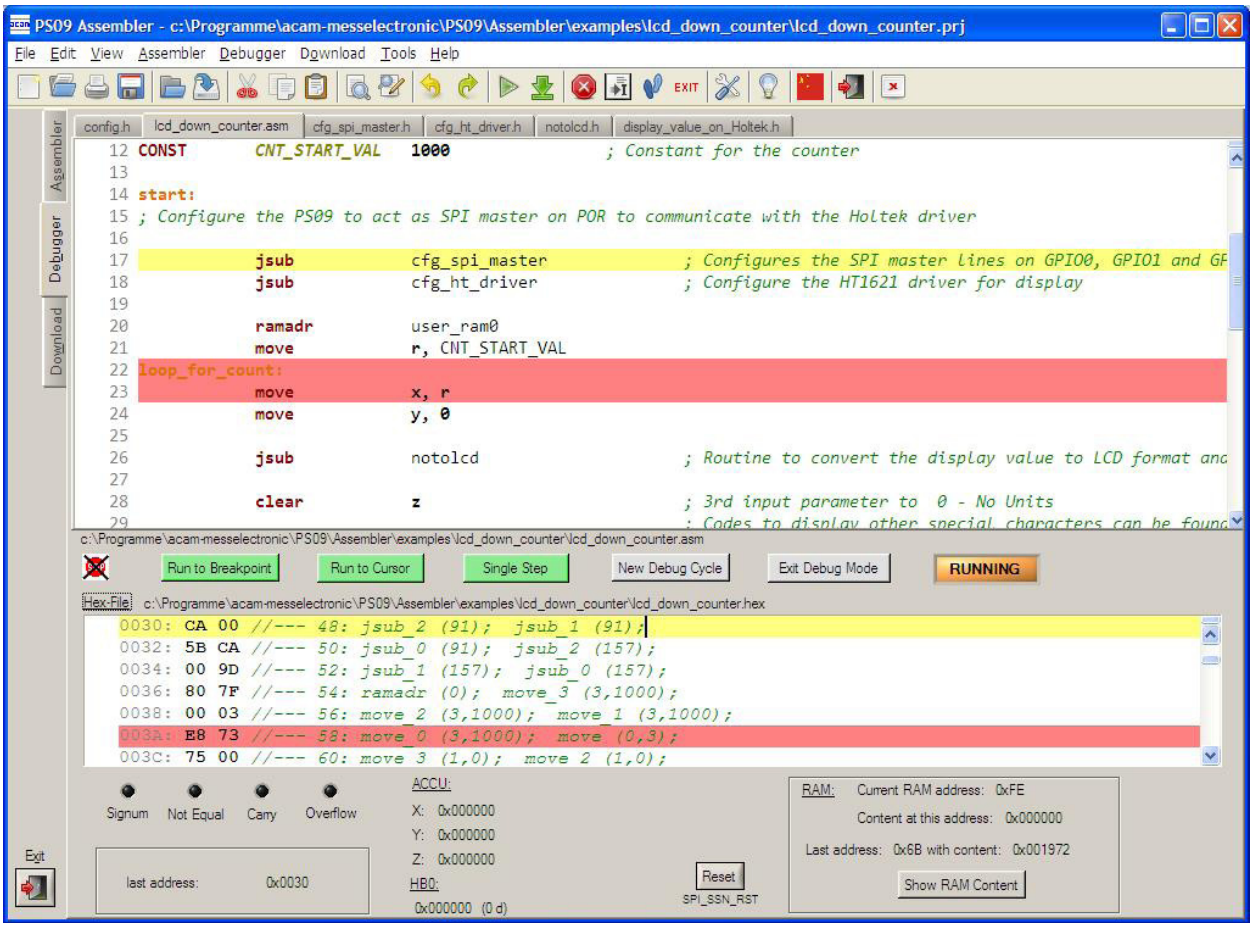
Step 1: Create an include file (e.g. config.h).

Step 2: Declare the inclusion in your assembler listing as follows: #include „config.h"

Include files that are in the same folder as the source file (*.asm) can be included with "filename.h". Include files that are in a path one level higher than the source file can be included with "../filename.h". If there is a set of common include files, they can be segregated in a folder called Lib in PS09 Assembler/Lib. The Include files in the Lib folder can be included in the program by <filename.h>. They are automatically included from the Lib folder.

The alternative method of using a single assembler files without using 'include' files at all is also supported.

6.2.3 Debugger Page



This tab provides a powerful debug interface. On this page you can debug the programs you wrote and make the processing of the code transparent. There are many data additionally available, e.g. status of the flags, content of the accumulator registers x,y,z or the content of the RAM. It is also possible to emulate the reset button. An active debugging process is indicated by a LED. There are several debug-modes available.

Please run the assembler on your listing first and make sure the file is processed successfully. Download the hex file then and switch then to the debugger tab. You should be able to debug the file now.

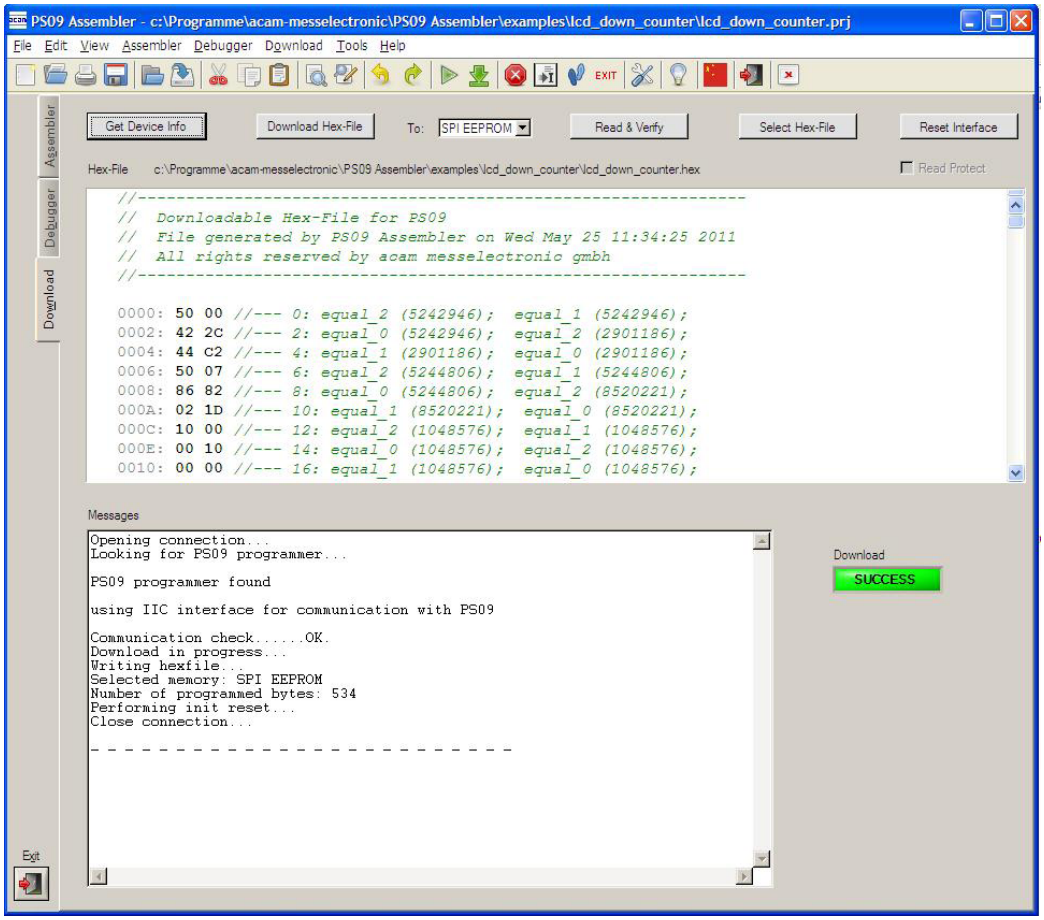
The controls in detail:

Source Code Window:	The source code and all the include files are shown in this window again (read-only) in multiple tabs. It is possible to set breakpoints on the left column of the window (next to the line numbering) in any of the files. When the brakepoint is set the code line is highlighted in red. During the debug process the line which is currently processed will be highlighted in yellow. Note: the break-point symbol will be shown on WIN 7 systems. On Win-XP systems there is only the line highlighted in red, but no stop symbol
Hex code Window:	The corresponding hex code is displayed in this window. During the debug process the line of the last received op code will be highlighted yellow. When a breakpoint is set in the source code the respective hex-code will be highlighted in red.
Debug Buttons:	There are six buttons available to control the debug process: Debug Buttons: There are six buttons available to control the debug process: - Run To Breakpoint: first set a breakpoint in the source code by double-clicking on the left column. Pressing the button lets the debugger run until the breakpoint is reached. - Run To Cursor: place the cursor in any line in the Source Code Window and press the button. The program will be processed until the line where the cursor is placed currently. - Single Step: press this button and one single instruction will be performed at a time. This mode is well suited for watching the step to step processing, e.g. the jump to a subroutine. - New Debug Cycle: by pressing this button the debug mode is exited and entered explicitly. That means also, that all settings are cleared. - Exit Debug Mode: by pressing this button the debug mode is exited - LED indicator: the LED indicates the current status of the debugger. If it is grayed the debugger isn‘t active. An active debugger is indicated through an orange-colored LED labeled with ‚RUNNING‘ on it.

Debugging information:	Debugging information: In this area of the tab there is additional information shown corresponding to the last processed instruction. For example, the corresponding address is shown, the state of the flags and the content of the accumulator registers x,y,z.
Reset button:	The reset button performs a reset of the PSØ9.
RAM information:	In this area RAM information is shown. While the last & current RAM address and their contents are shown every time, you can press the button ‚Show RAM content' and you will get another pop-up window where you find the RAM content of addresses 0..255. It shows the content of the entire RAM address space, the addresses being sorted into one of 4 colors. The color indicates if the RAM address is a 'Reserved address', 'User RAM address', 'System RAM address' or 'Configuration RAM address'. When the program is being debugged, the RAM contents displayed can be refreshed with the 'Refresh' button on the pop-up window.

The debugger tab page gives you the possibility to run your program step by step or only parts of it and a lot more. Additional information is obtained which is normally not obvious.

6.2.4 Download Page



The 'Download' tab provides an interface to program the PSØ9 with the compiled hex code. For this purpose the hex-file is displayed and several buttons are provided to handle the download. A big message window shows the status of the connection and the programming. Furthermore a LED indicates success or failure of the download.

The controls in detail:

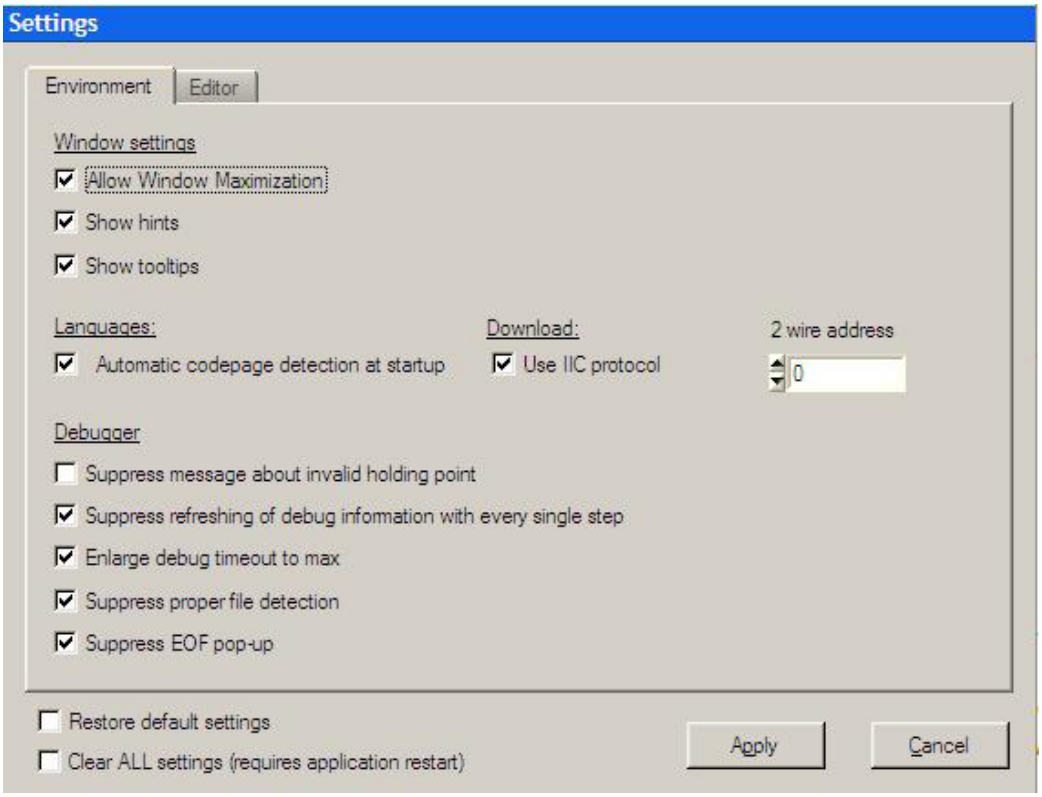
Hex code Window:	The previously compiled hex-file is displayed in this window. Furthermore the path to the file is displayed above the window in order to have the possibility to check if it is the proper file.
Download Buttons:	There are five buttons available to handle the download process: - Get Device Info: press this button to receive general information about the USB connection and the device itself (e.g. instrument handle, vendor-id, etc.). In case there are problems with the USB connection this button and the corresponding messages can help to find the problem's cause. - Download Hex-File: by pressing this button the current displayed hex-file is downloaded to the PSØ9. Whether the code has to be downloaded to the external SPI EEPROM or the on chip OTP can be selected with the drop down menu. The several steps which are performed, like erasing blocks or writing the data, can be monitored in the Status Messages Window. - To: Selects the download destination. Whether the code has to be downloaded to the external SPI EEPROM or the on chip OTP must be selected with this menu. Read & Verify: press this button to verify the OTP/EEPROM content matches the selected hex code file content. Caution: naturally the OTP can only be written once (or incrementally several times). That means that correction or re-writing of cells is not possible! - Select Hex-File: in the case you don't want to download the currently displayed hex-file (or none hex-file is selected yet) you can choose a previously compiled hex-file via this button. By pressing this button a file select popup window appears where you can select your file. The file is displayed then in the hex code window and can be downloaded by pressing the ‚Download Hex-File' button then. - Reset Interface: in case there are any problems or disturbances with the USB connection, the interface can be reset via this button. Read Protect: This check box is enabled only when the download destination is the OTP. When selected, the fuse address 8143 of the OTP is additionally written with a non-zero value automatically, thus read protecting the OTP. Hence this must be set very carefully only in the last step of programming completion. For more information refer Section 6.2 .1 of PSØ9 data sheet.

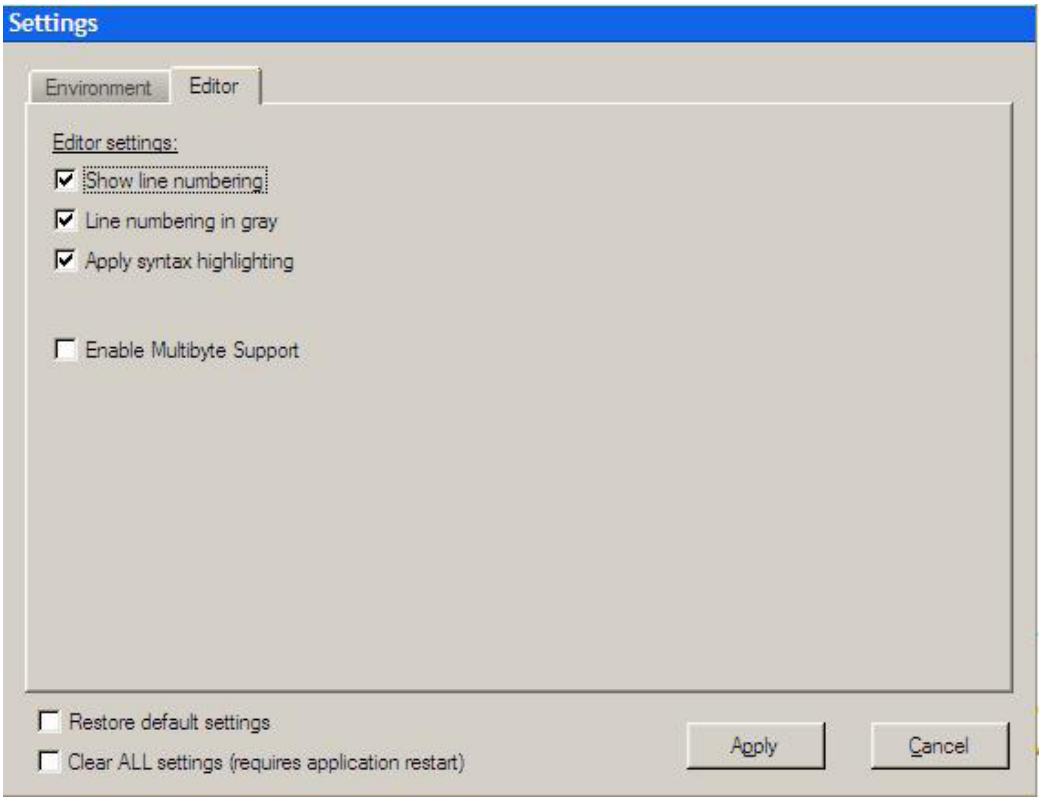
Status Messages Window:	In this window there are the status messages displayed regarding the download process.
Status Indicator:	The status indicator LED shows the success of the performed action. While an action is performed the LED color is orange and labeled with ‚IN PROGRESS'. After finishing the action the LED shows either ‚SUCCESS' on a green background in case the action was performed successfully or ‚FAILED' on a red background in case the last action failed.

Please note, that downloading the hex code to the OTP/ external EEPROM is a required step to perform before you can use the debugger.

6.2.5 Assembler Settings

Different settings for the assembler and the editor can be done by selecting the item “Settings” in the “Tools” menu. A screen appears that shows the following sheets:





Note: If the communication with the PSØ9 is to take place through SPI protocol or IIC protocol must be set accordingly in the Settings tab 'Use IIC protocol' option and also on the PSØ9-EVAL Motherboard using the on-board jumper.

7 Miscellaneous

7.1 Literature Guide

Datasheets

Title	Document-No	Date
PSØ9 Single Chip Solution for Strain Gauges	DB_ PSØ9 V0.3	May 2011
PSØ81 Single Chip Solution for Strain Gauges	DB_ PSØ81 V0.8	August 2010
ALCS-350 V2 Load Cell Simulator	DB_ALCS_V2 V0.1	July 2009

White Papers

Title	Document-No	Date
How to Lower Gain and Offset Drift Drift of a Load Cell by using TGGain and TKOffset Factors of PSØ81	WP002 V1.0	October 2008
Construction Guideline for solar driven Scales	WP001 V1.0	June 2008

Application Notes

Title	Document-No	Date
Meterological Investigations of PSØ81 Determining Zero Drift and Gain Drift	AN018 V1.0	July 2008
Strain Gauge Wiring with PICOSTRAIN	AN012 V1.0	August 2005
Rspan by Temp Compensation Compensation of Gain error for uncompensated Load Cells	AN021 V1.0	July 2009
Design Guideline for Building a Solar Kitchen Scale	AN022 V1.1	August 2009
Design Guideline for Building a Solar Body Scale	AN023 V1.3	September 2009

All available documents can be downloaded from the acam website at:

http://www.acam.de/download-section/picostrain

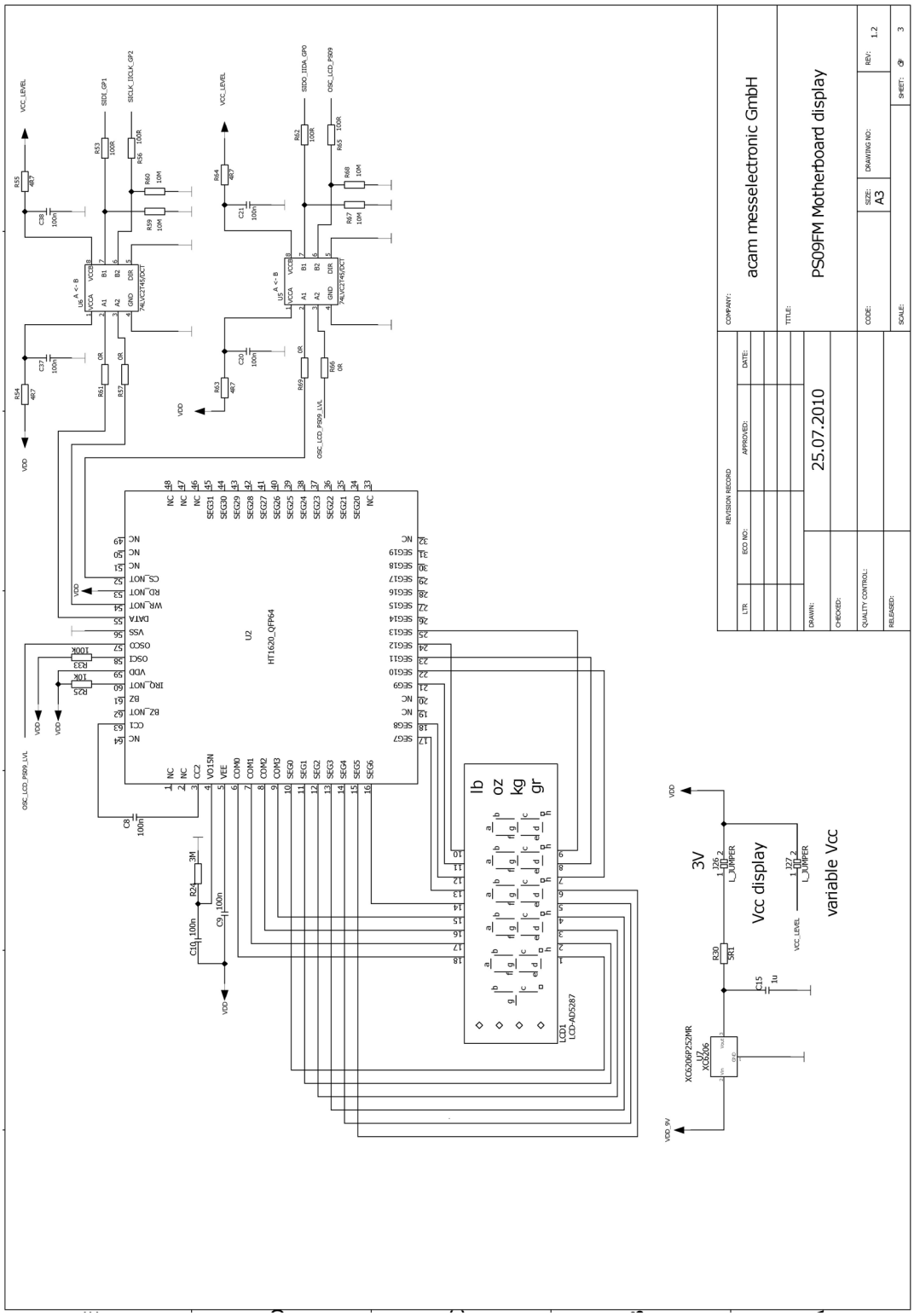
7.2 Document History

May 2011 First release

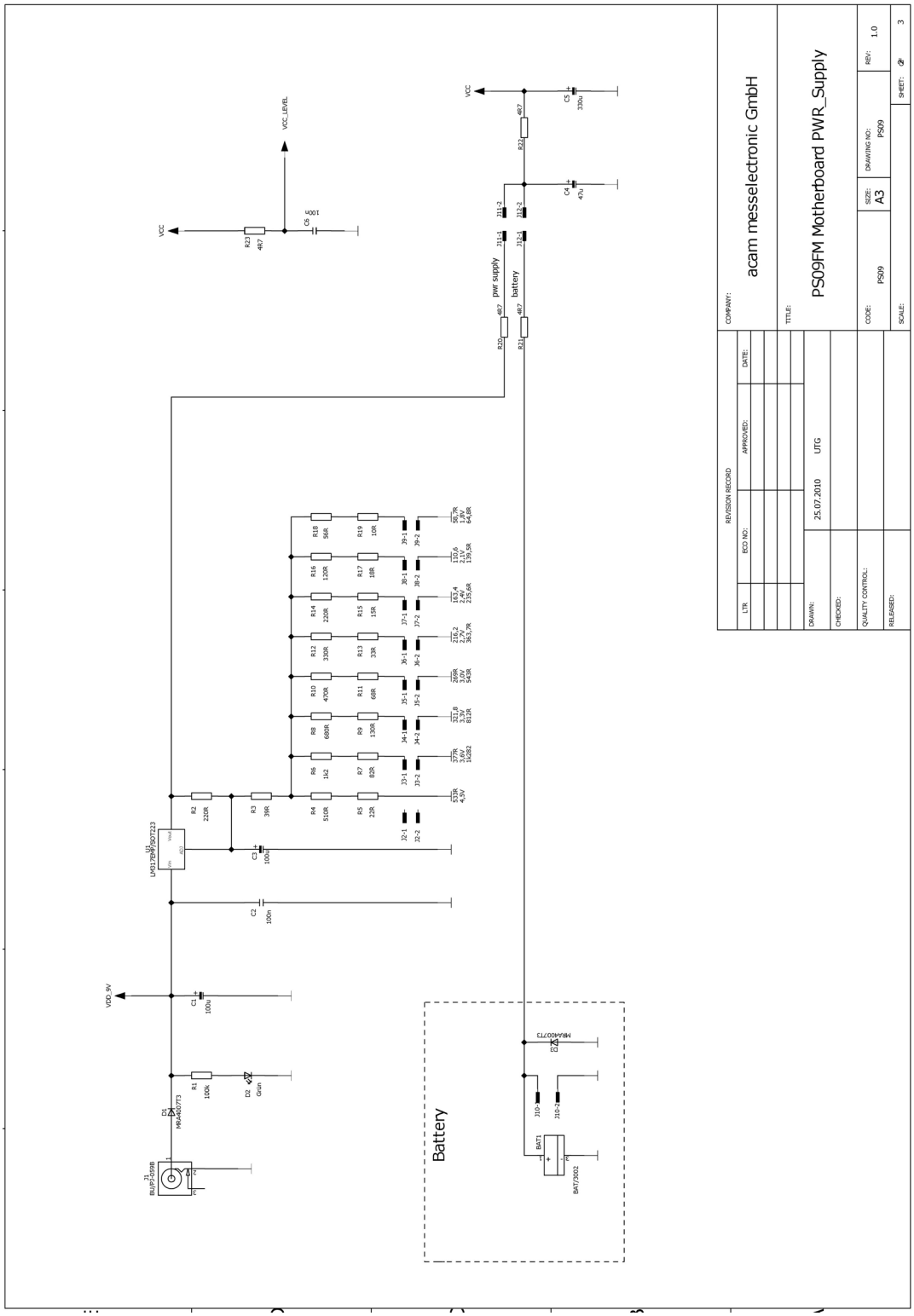
8 Appendix

8.1 Schematic Diagrams

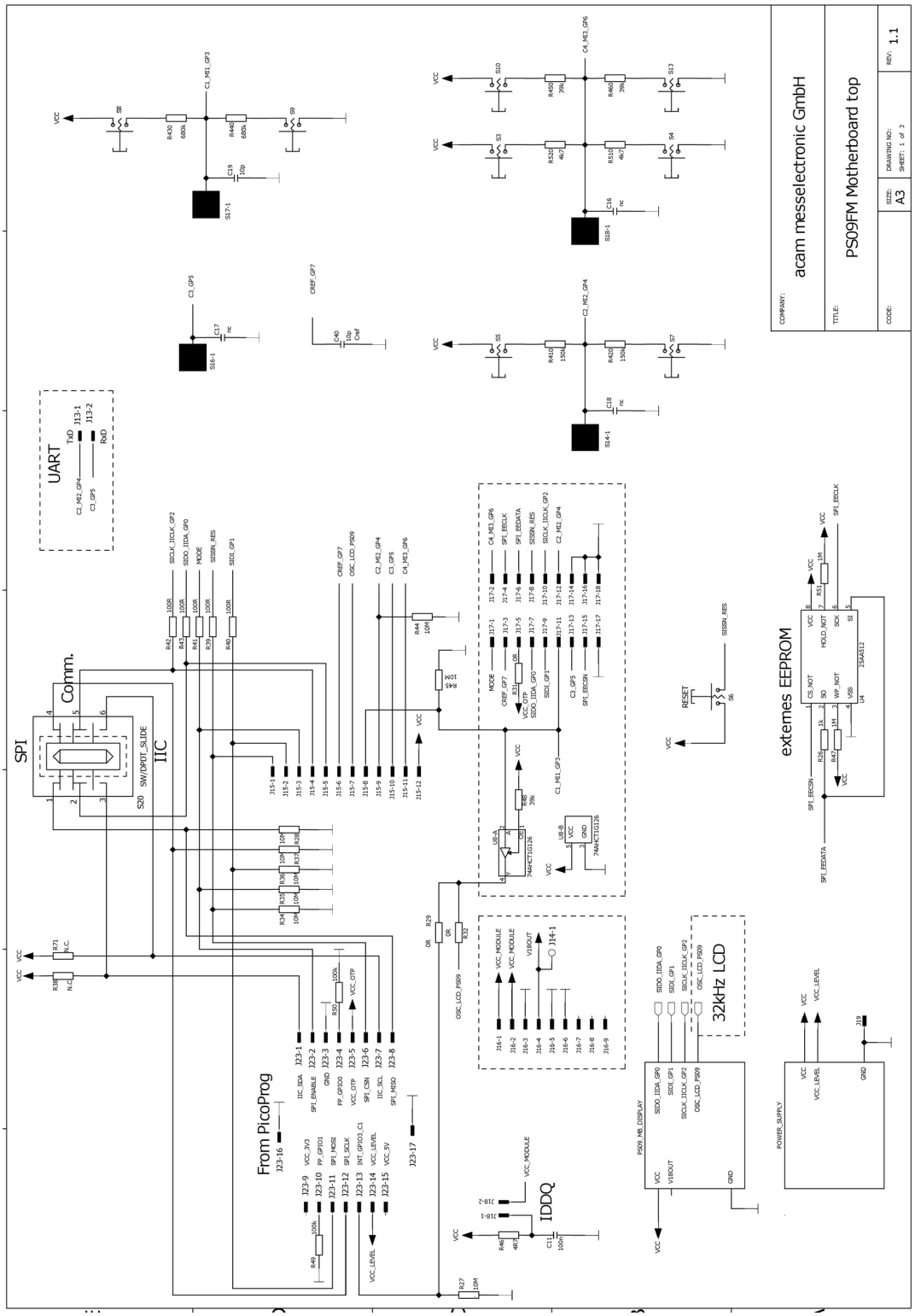
8.1.1 PSØ9-EVA-MB Main Board



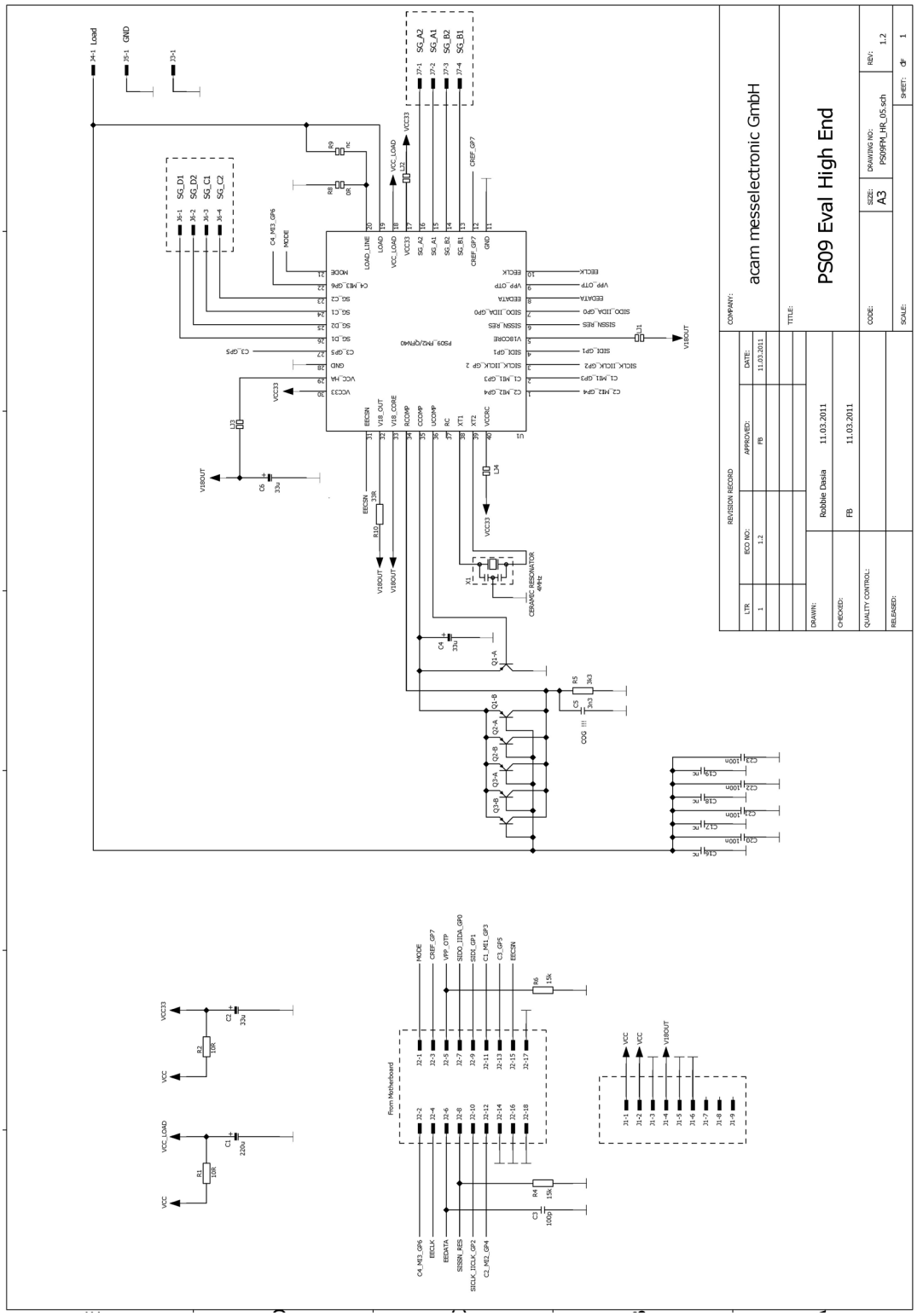
PSØ9-EVA-MB Main Board



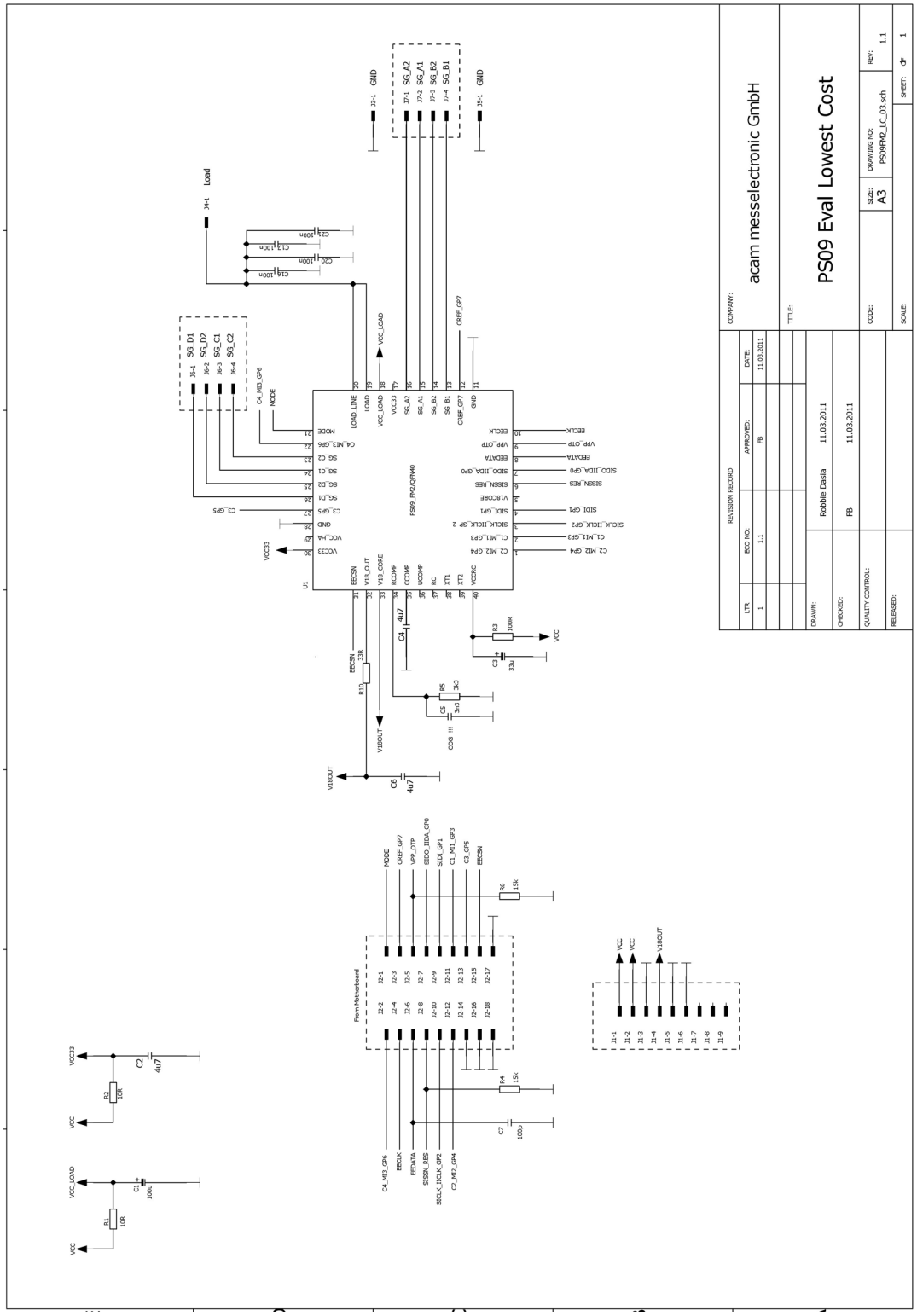
PSØ9-EVA-MB Main Board



8.1.2 PSØ9-EVA-HR High Resolution Module

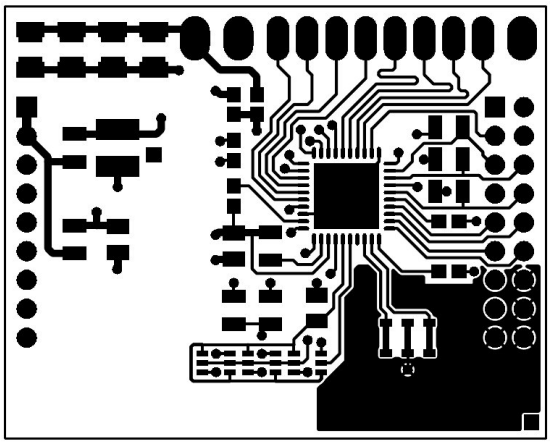


8.1.3 PSØ9-EVA-LC Low Cost Module

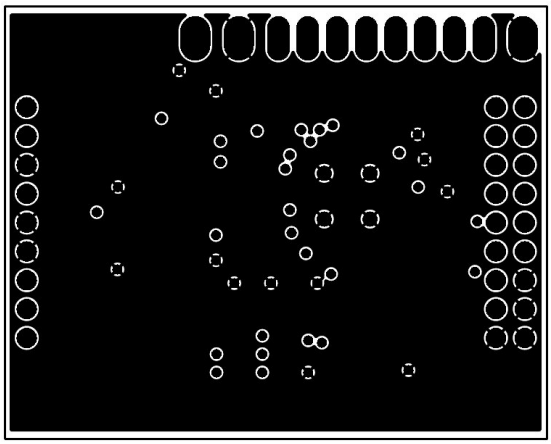


8.2 Layout PSØ9-EVA-HR Module

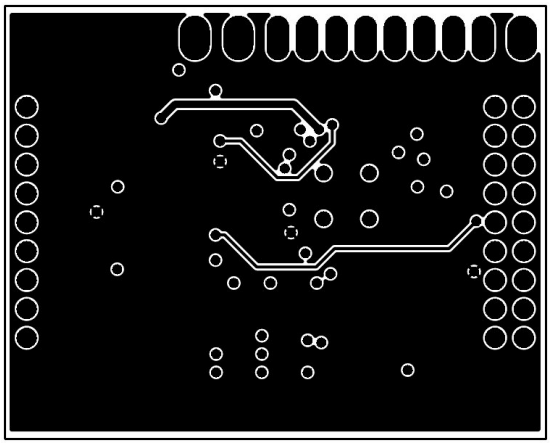
Layer 1



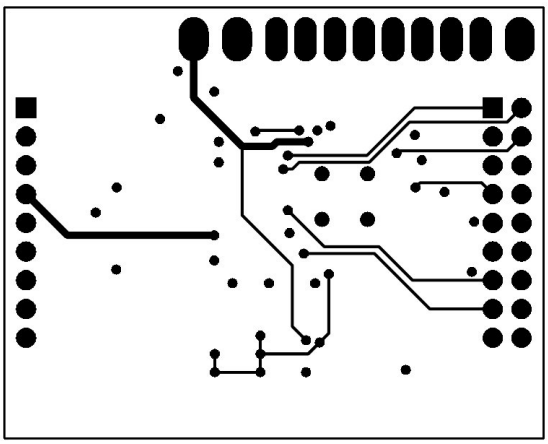
Layer 2



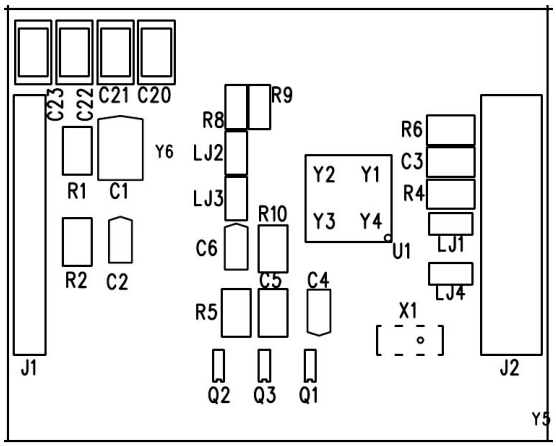
Layer 3



Layer 4

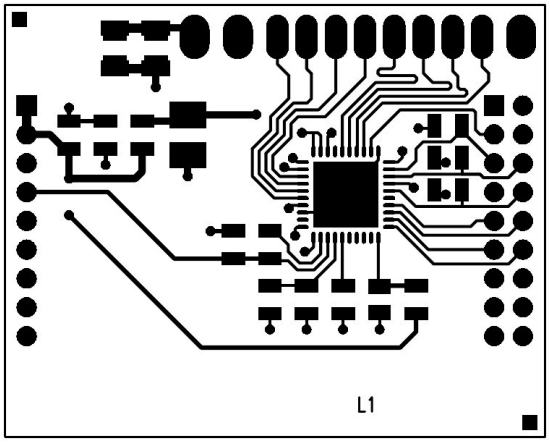


Layer 5

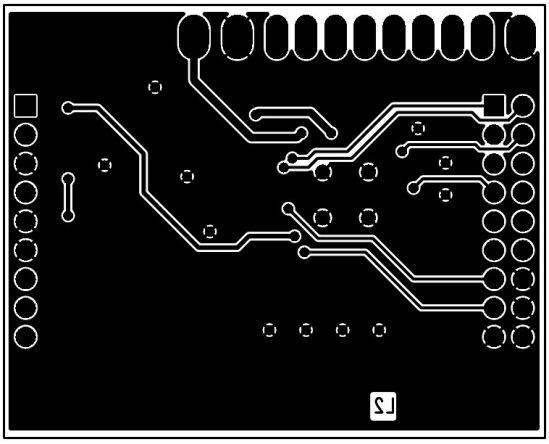


8.3 Layout PSØ9-EVA-LC Module

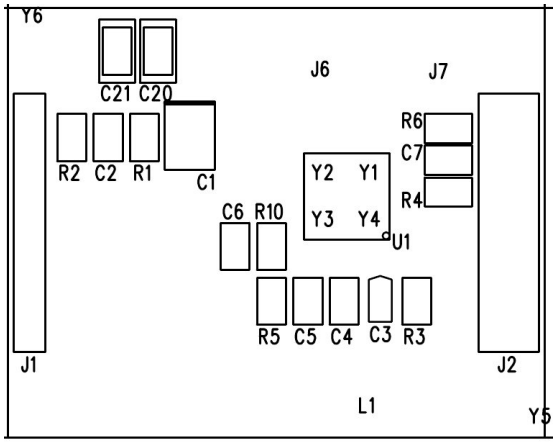
Layer 1



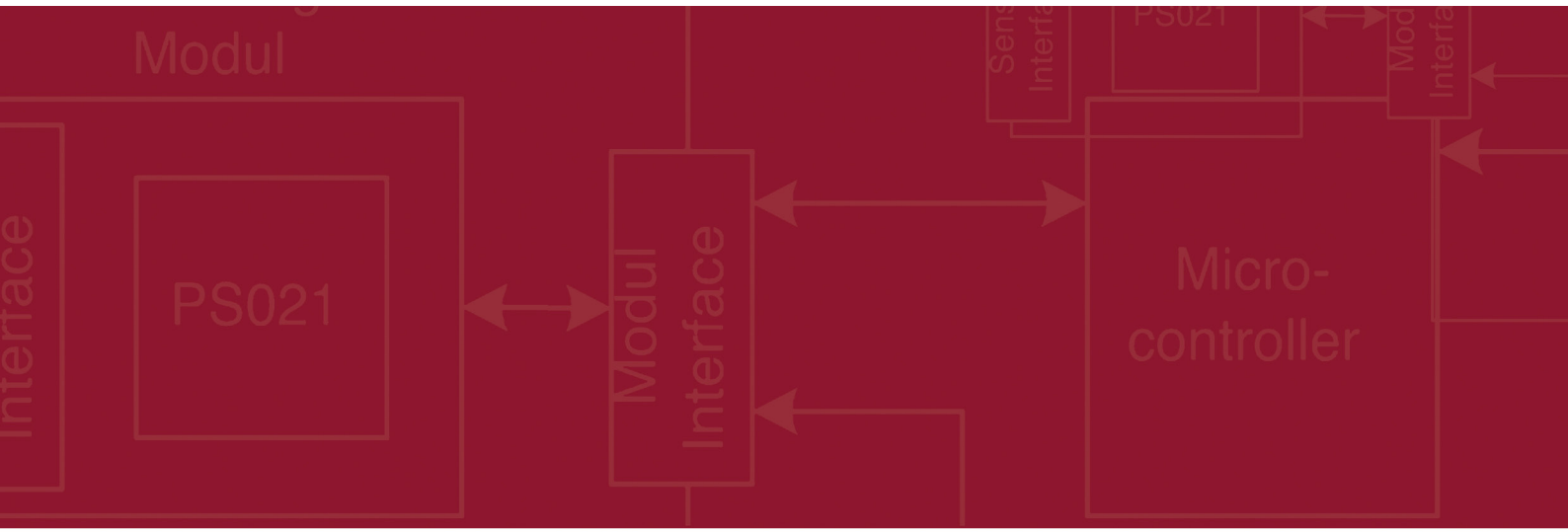
Layer 2



Layer 3



PICOSTRAIN



acam-messelectronic gmbh
Am Hasenbiel 27
76297 Stutensee-Blankenloch
Germany / Allemagne
ph. +49 7244 7419 - 0
fax +49 7244 7419 - 29
e-mail: support@acam.de
www.acam.de