

readme.txt

```
/******  
*  
* @author      Markus Kilian, Mechaless Systems GmbH  
* @date        Created:    2010-05-31  
*  
* This file contains important information about the changes from previous  
* firmware to firmware 4.00 and later versions  
*  
* Summary of the main changes in firmware  
* previous firmware version 3.02: only works with E909.05  
* actual firmware version 4.00: works with E909.05 and E909.06  
*  
*****/
```

instruction how to update an existing user application with previous firmware to firmware 4.00 and later versions:

1. Start a new project with E909.06 or E909.05.
Therefore use an applicaton note with firmware 4.00 from Mechaless
2. Remove the example c code you dont't need.
E.g. remove c3d functions and the library c3d if you dont't intend to use the coordinates calculation
3. Add the additional c source files you wrote for your application.
4. Add the variables and the user functions from your existing application.
Put them to the same location in c code like in your previous application.
5. Compile, link and run the project.

```
/****** application (main.c) *****/
```

- coding style for variable declaration established
- version for application will be written to flash area
- user space reorganized
- device initialization for IAR IDE is now done in firmware function haliosInitialize()
- a specific project number is set to SFR
- last reset reason is handled in firmware function failState()
- watchdog is set with firmware function deviceSetwatchdogTime(uint32_t)
- LIN initialization added but not used
- Due to hardware registers for E909.06 new paramters for the loop configuration added.
- HALIOS tools initialized with function init_haliostools().
- GPIO used for the level switch module now set and cleared in the application.
- software temperature compensation removed. Hardware temperature compensation is recommended. In E909.06 it is done in internal hardware.

```
/****** Firmware (lib_firmware_05 / lib_firmware_06) *****/
```

- speperated firmware libs for E909.05 and E909.06
- SFR and userspace are speperated. Size of SFR is depends is firmware dependend. Size of user space is 255 words.
Refer to firmware_spec.pdf for further information.
- interrupt handling is now integrated in firmware library

readme.txt

- The maximum number of loops increased. Actual number is 16.
- Function paramSetProjectNr(char*) is new.
- Function deviceSetWatchdogTime(uint32_t) is new.
- Function haliosInitialize and halios measure:
The pin set before measurement and cleared after measurement is initialized with the function parameter on haliosInitialize
RDY_ON and RDY_OFF now depend from the pin which was initialized in haliosInitialize.
- halios loop initialization
there are two additional members in the struct LoopConf available:
One variable is for the preAmplifier settings (preAmp) and the other variable is used for the settings in the hardware register
"Measurement Configuration HALIOS clock"
- Type for struct LoopConf renamed in loopConf_t due to current coding style.
Following members were added for E909.06 and its corresponding hardware registers:
-> uint16_t DC_offset
-> uint16_t PreAmp
-> uint16_t ClockConf
- The reset reason is now handled in a separated firmware function called failState.
- The buffer for SFR and user space (gui_buffer) is separated into gui_sfr_buffer and gui_user_buffer.
The access to user space with the macro User(x) and the direct access to any address with macro BufferConf(n) is obsolete. Now use paramSetSFR and paramGetSFR instead.
- The call of deviceCheckVersion() on every sample now is obsolete.

/****** HALIOS tools (lib_haliostools) *****/

- Variables from application imported to the library as members for HALIOS tools.
- Function init_haliostools() is new. initializes the variables of HALIOS tools.
- Function haliosCompCalib() now returns an enum type.
- Access to GPIO port removed.
- software temperature compensation removed

/****** USB (lib_usb) *****/

- USB communication protocol changed
- Function usbInitialize() takes initializes the USB interrupt request pin by its function parameter.