LDC1000 SENT Bridge
MSP-EXP430F5529LP based
Getting Started

Contents

1. Quick Start 2
2. Needed Equipment 2
   - 2.1 Hardware 2
   - 2.2 Software 2
   - 2.3 Recommended Documents 3
   - 2.4 Additional Documents 3
3. Hardware Setup 3
   - 3.1 LDC1000EVM/GUI Usage 3
   - 3.2 LDC-SENT-Bridge Configuration 4
4. Launchpad Driver Installation 4
5. SENT Output 5
6. MATLAB Logging Script 6
7. MSP430 Code Composer Studio Firmware flashing 8
8. LDC EVM Software/ RPmin-RPmax Configuration 11
9. Hints 11

created: 20140103 by Julian Becker
last revised: 20140221 by Julian Becker

1. Quick Start

Out of the box the Launchpad is programmed with the LDC-SENT Bridge firmware.
As soon it is powered up by connecting it via Micro USB to any USB power source it will start transmitting the LDC's proximity and frequency data via SENT according to Chapter 5.
(Don't care about USB drivers)

The SENT Signal is available at the jumpers on top of the Launchpad add-on board. (Refer to Figure 1)

If you want more in-deep explanations for LDC1000, LDC1000EVM, USB driver installation, SENT data logging in MATLAB and firmware modifications… refer to the next Chapters of this Guide.

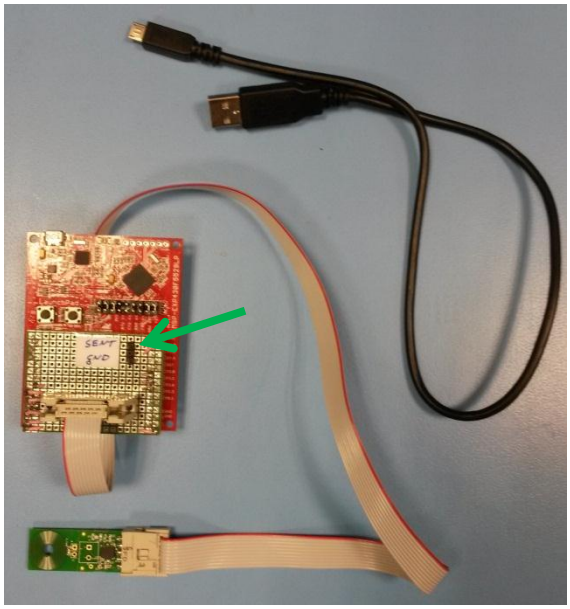

Figure 1 Out of the Box LDC SENT Bridge setup – arrow pointing to the SENT header

2. Needed Equipment

2.1 Hardware

- Launchpad MSP-EXP430F5529LP including Micro USB cable
- Add-on Board for Launchpad
- LDC1000EVM (prepared with connectors for Add-on Board connection)
- 10 pin ribbon cable
- Inductive Sensing demonstrator

2.2 Software

The complete setup was created and tested with the following software setup:

- Windows 7 Professional Service Pack 1 – 64bit @4GB RAM
- MATLAB R2013a (8.1.0.604) 64bit (win64)
- Code Composer Studio 5.5.0.00077 with MSPware installed (http://www.ti.com/tool/ccstudio)

- LDC1000EVM Software
  (http://www.ti.com/tool/ldc1000evm)
  LDC1000 EVM Software GUI - http://www.ti.com/lit/zip/snac058
  LDC1000 Firmware Library for the MSP430 - http://www.ti.com/lit/zip/snac059
  (The used firmware is based on this library)
- SENT_EXP430F5529LP.zip coming with this document containing:
  - MATLAB scripts
  - Firmware source code –
    Including the COM Port Driver used for Lauchpad-MATLAB connection (Chapter 4)

During installation and working "Administrator Rights" are necessary (driver installation, Device Manager Access)

2.3 Recommended Documents

- LDC1000 Datasheet - http://www.ti.com/lit/gpn/ldc1000
- LDC1000 EVM User's Guide - http://www.ti.com/lit/pdf/snau150

2.4 Additional Documents

- Interfacing LDC1000 with the MSP430 LaunchPad - http://www.ti.com/lit/pdf/snaa213
- MSP-EXP430F5529LP LaunchPad Development Tool User's Guide - http://www.ti.com/lit/pdf/slau533
- MSP430F5529 Datasheet - http://www.ti.com/lit/gpn/msp430f5529
- MSP430x5xx Family User's Guide - http://www.ti.com/lit/pdf/slau208

- In-deep MSP430 USB Information: http://www.ti.com/tool/msp430usbdevpack
- SENT Specification (J2716 JAN2010)

3. Hardware Setup

There are two supported possibilities to use the LDC with this equipment:

3.1 LDC1000EVM/GUI Usage

The Hardware setup can be plugged together to provide full LDC1000EVM functionality (GUI…):

Figure 2 LDC board and MSP-USB board with Inductive Sensing Demonstrator

Please double check polarity before powering up. (Take picture for reference – LDC board and MSP-USB board both should have the components on the same side) Plugging in the LDC board the wrong way around will cause malfunction and could damage both LDC and USB board.

- To use the hardware in this setup follow the instructions in LDC1000 EVM User's Guide (http://www.ti.com/lit/pdf/snau150)

3.2 LDC-SENT-Bridge Configuration

The following setup provides a LDC – SENT bridge functionality with additional SENT raw data logging feature via USB as long as the Launchpad is flashed with correct firmware.
The SENT output is available at the two pin header. (3.3V)
Refer to Figure 1 for how to plug the setup together.

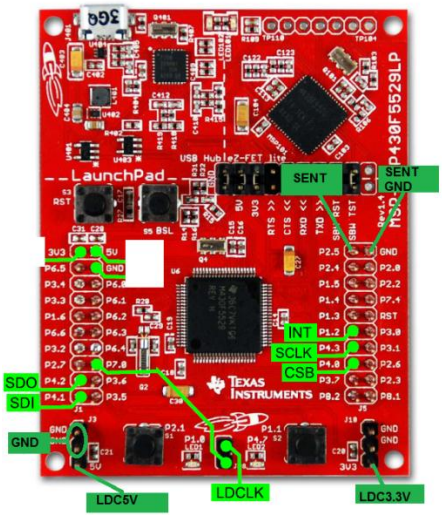

Figure 3 - Connection between LDC-Board and Launchpad,
adapted from Interfacing LDC1000 with the MSP430 LaunchPad page 7 (snaa213)

- Setup the hardware this way when working with the Chapters 4-7 of this document.

4. Launchpad Driver Installation

- When plugging in the Launchpad the first time some drivers need to be installed.
- Open the Windows Device Manager select and install the correct drivers.

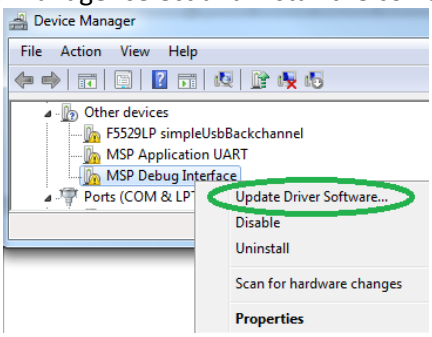

Figure 4 Device Manager with Launchpad connected the first time

- If the Launchpad is not flashed with the LDC-SENT-Bridge Firmware the device “F5529LP simpleUsbBackchannel” will not appear – skip the steps on the first run and load the needed firmware according Chapter 7 after installing the other Launchpad’s drivers.
- Right Click on each device:
    - Update Driver Software… - a new window pops up
    - “Browse my computer for driver software”
    - For MSP Application UART and MSP Debug Interface:

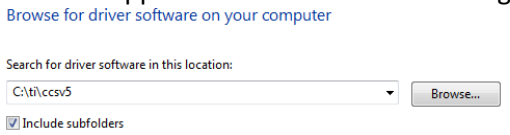

Figure 5 Browse for drivers

        - The drivers are in the installation folder of Code Composer Studio
        - Tick Include subfolders
        - (typically C:\ti\ccsv5)
        - Click on next – searching takes time…
        - The drivers of these devices will be installed
    - For F5529LP simpleUsbBackchannel:

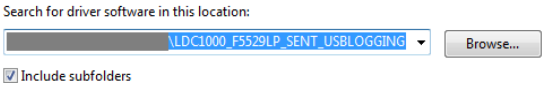

Figure 6 Browse for drivers

        - Extract the folder LDC1000_F5529LP_SENT_USBLOGGING from the SENT_EXP430F5529LP.zip coming with this document
        - The drivers are in this folder
        - Tick Include subfolders
        - Click on next
        - A new window will pop up
        - Click on Install this driver software anyway
        - The drivers of this device will be installed
- Now all these devices should appear in the Device Manager category Ports like this. (COM numbers may differ from picture)

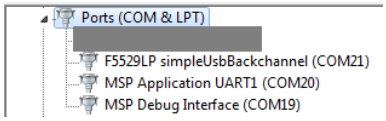

Figure 7 Device Manager with Launchpad installed

5. SENT Output

- The MSP430 is powered with 3.3V – The SENT max. output level is therefore 3.3V, too.
- To use the SENT Output the hardware should be setup like in Chapter 3.2.
- According to Configuration Shorthand (7. in SENT specification) the Launchpad’s output is:

SENT2010-03.0us-6dn-npp-nsp-ns

- CRC calculation according to SENT2010
- Clock tick length 3µs
- 6 data nibbles
- No pause pulse
- No serial protocol
- Sensor type not specified in spec (but close to A.1 in SENT specification)

- The transferred data frame is close to A.1 in SENT specification:

```
// STATUS : 0
// DATA1  : PROXDATA MSN
// DATA2  : PROXDATA MidN
// DATA3  : PROXDATA LSN
// DATA4  : FREQDATA LSN
// DATA5  : FREQDATA MidN
// DATA6  : FREQDATA MSN
// followed by crc generated automatically
```

- The transferred data can be suited to the actual needs by modifying the Launchpad's firmware. The mapping of the LDC's 16bit PROXDATA to the SENT data nibbles 1-3 can be changed, for example. (See sent.c)
- As long as no LDC data are available (during startup) all data nibbles transferred will be 0x0.
- If the LDC can't provide any new data (LC-tank's damping too high, out of range, connection failures…) the last valid values will be transferred.

6. MATLAB Logging Script

The MATLAB Logging Script is able to log all the data which are transferred via the SENT interface utilizing the virtual COM port of the Launchpad. When running the script with the correct port setup, all data will be logged. During logging a automatically refreshing plot will show the current values. After finishing the logging by pressing any key (while having the plot window selected) all received data will be plotted again and is available in the MATLAB workspace in the sent_packet struct for further processing. (Please keep in mind, that the data transferred by the SENT interface is logged only. If you want to have all LDC data – at a higher sampling rate please use the LDC1000EVM with the GUI or dedicated MATLAB/ Labview environments)

- Extract the SENT_Matlab folder from SENT_EXP430F5529LP.zip coming with this document.
- Make sure that the Launchpad is properly connected to the PC.
- Open MATLAB and change the current folder to the extracted SENT_Matlab folder.
- Your MATLAB Window should look like this now:

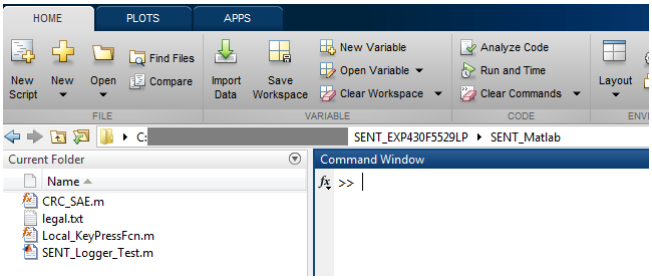

Figure 8 SENT_Matlab folder

- Before running any MATLAB script, find out which COM Port the Launchpad is assigned to. (like in Chapter 4 – Device Manager – COM Port of the F55529LP simpleUsbBackchannel)
- COM21 in this document (might defer)

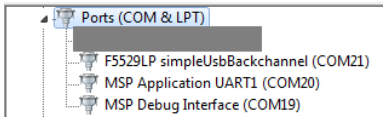

Figure 9 Device Manager Ports

- Open the SENT_Logger_Test.m file in MATLAB Editor (e.g. by double clicking on it) and change the COM Port in about line 52 according the information taken from your Device Manager.

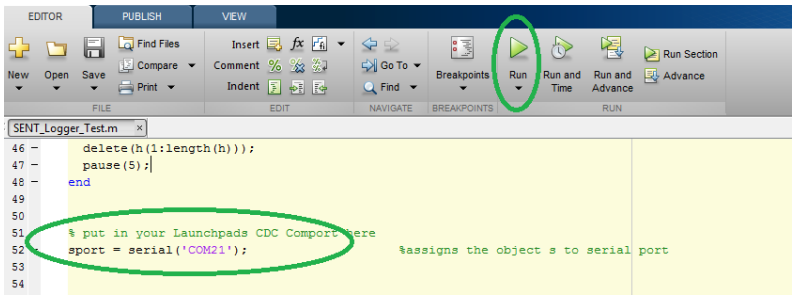

Figure 10 MATLAB Editor

- Now save the script and either run it out of the editor (press the green run button or F5) or close the editor and run it from the MATLAB Command Window by just typing in the filename without extension and pressing enter.
- If the Launchpad is connected properly and the correct COM Port is selected a plot window will open up, which is showing the latest proximity and frequency data transferred via the SENT interface.

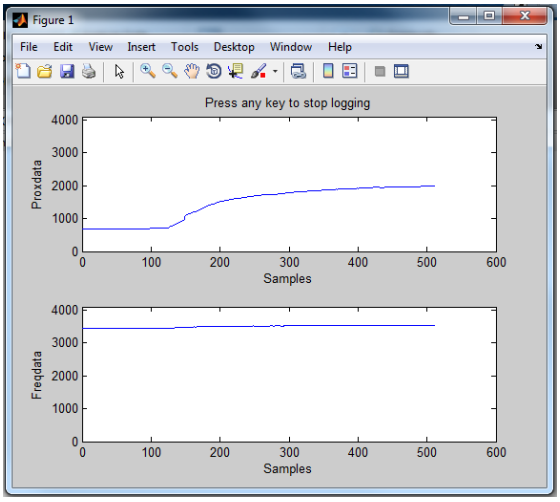

Figure 11 self-refreshing MATLAB plot showing the last SENT samples

- This plot window will refresh automatically as soon new data are received from the virtual COM port.
- Stop the logging by pressing any key with window focus on this plot.
- A new plot will pop up showing all the data during the logging.

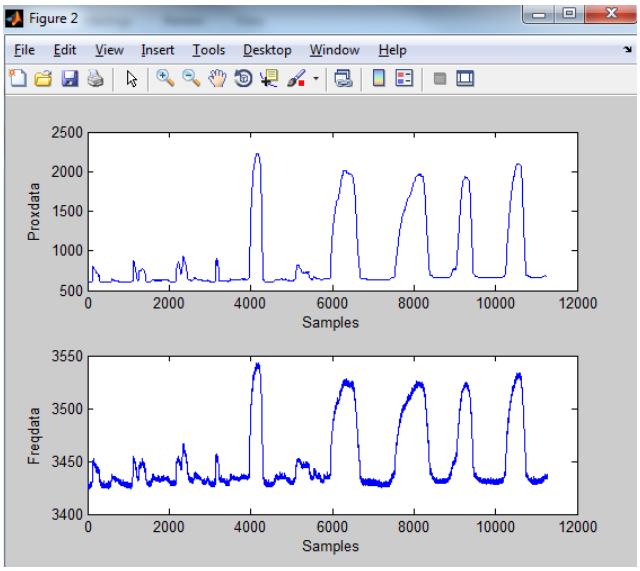

Figure 12 Example Log, showing the LDC's coil being pushed around between two fingers randomly

- The logging is done and all data are available in the sent_packet struct for further processing in MATLAB in the workspace.
- Before starting a new logging session make sure you save all needed data from the workspace.
- For starting a new logging session run the SENT_Logger_Test script again.

7. MSP430 Code Composer Studio Firmware flashing

If the Launchpad is not flashed with the LDC-SENT-Bridge firmware or the firmware is modified the original source code is available in the SENT_EXP430F5529LP.zip coming with this document in the subfolder LDC1000_F5529LP_SENT_USBLOGGING.
This folder contains a Code Composer Studio project which can be imported into any Code Composer Studio workspace for further work. The Launchpad has an eZ-FET lite on board capable of flashing and debugging the Launchpad's MSP.
After flashing/ loading the firmware of the MSP (for debugging) it stays in the MSP and will be executed as soon appropriate power is available. Therefore Code Composer Studio is needed for modifications, initial flashing and debugging (only).

This Chapter will show how to import the LDC-SENT-Bridge firmware to Code Composer Studio and flash it to the Launchpad.

- Extract the SENT_EXP430F5529LP.zip.
- Open Code Composer Studio (CCS) – the Workspace Launcher will appear, select a new folder to be the used workspace and press OK.

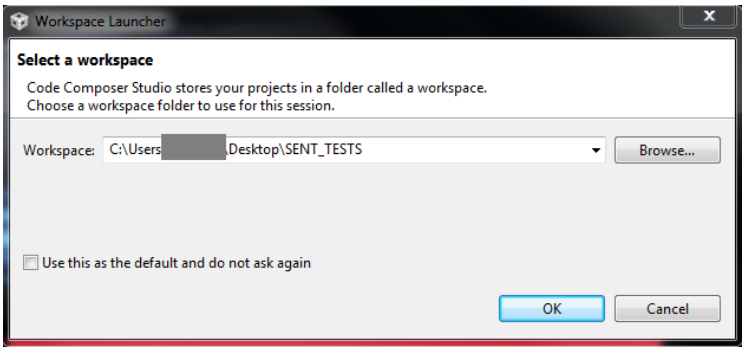

Figure 13 CCS Workspace Launcher

- CCS will open up, go to Project -> Import Existing CCS Eclipse Project

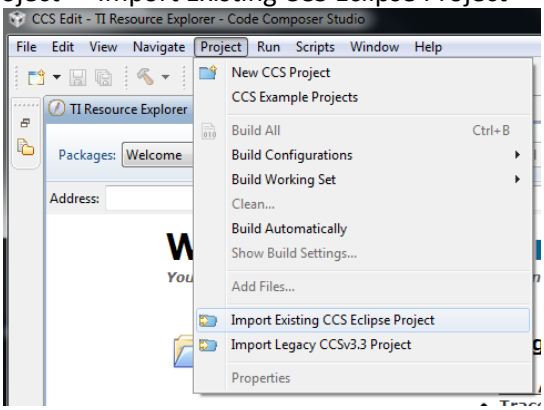

Figure 14 CCS Project Import 1

- A new window will pop up: Import CCS Eclipse Project
    - Select search-directory: Browse to the location where the SENT_EXP430F5529LP.zip was unpacked to
    - Discovered Projects: tick the …USBLOGGING Project
    - Place a tick at Copy projects into workspace
    - Press Finish

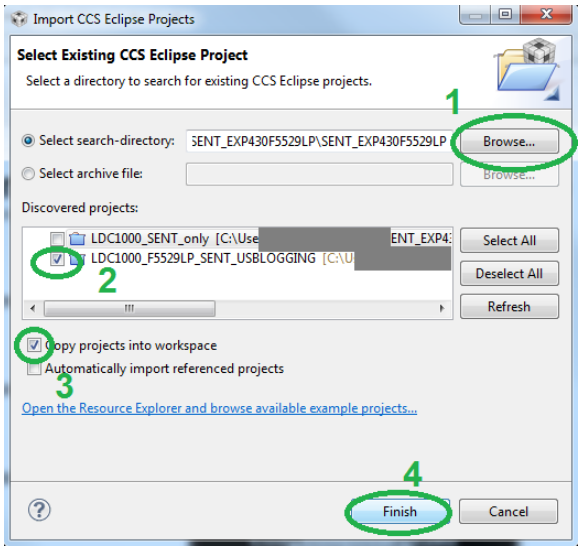

Figure 15 CCS Project Import 2

- CCS will now import the project and look like this:

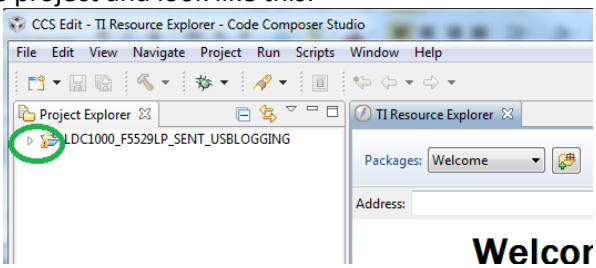

Figure 16 CCS initial view after importing the project

- Expand and select the project by double clicking on it in the Project Explorer.
- (Open main.c, sent.c and sent.h to check through the source code to get familiar with it and make some adjustments.)
- (use CTRL+B to build inbetween while coding.)
- Connect the Launchpad to the PC if not done already.
- Click on the green bug/ hit F11 to compile and start debugging.

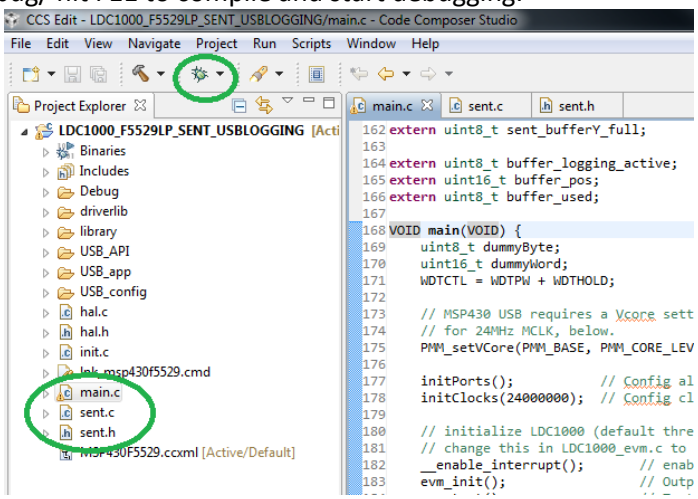

Figure 17 CCS Edit view

- The firmware will be loaded into the MSP and the CCS perspective will change to debug mode.
- Either start the execution of the code with the green arrow (you can place breakpoints by double clicking at the white space to the right of the line numbers) or press the red square to stop the debugging session.

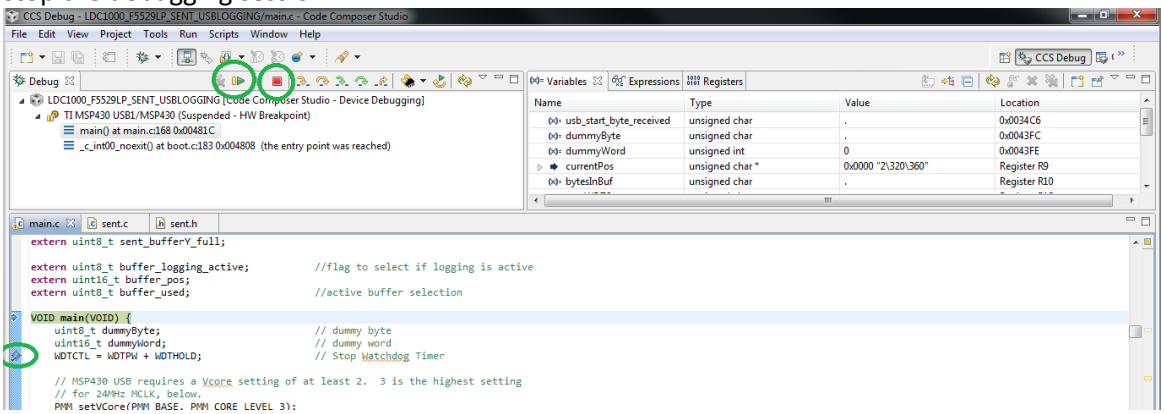

Figure 18 CCS Debug view

- After closing the debugging session the MSP can be reset by pushing the RST (S3) button on the Launchpad and will execute the firmware as soon/ as long appropriate power is applied.

8. LDC EVM Software/ RPmin-RPmax Configuration

Refer to the LDC1000 datasheet first, to get to know about the effects of changes in the RPmin and RPmax registers.
Using the default LDC-SENT-Bridge Firmware the RPmin, RPmax registers are configured within certain limits automatically at power up. If static settings or additional LDC register settings are necessary it is recommended to test them first with the LDC EVM Hardware (Setup according to Chapter 3.1) and the LDC1000 EVM GUI. All register settings can be tested here and copied later to the LDC-SENT-Bridge Firmware.
Depending on the targets/ coil setups in use, the RPmin and RPmax settings are most interesting.

After carefully evaluating the setting with the EVM GUI the values can be looked up there and copied to a section in the main.c of the source code.

Remove the comment // from the shown lines like in Figure 19.
Keep in mind, that RPmin and RPmax only can be modified if the LDC is in PWRCONFIG 0x00.
Later on the LDC needs to be put back in PWRCONFIG 0x01 mode and needs some settling time.

```
205
206   while(!spi_writeByte(NULL,LDC1000_CMD_PWRCONFIG,0x00));
207
208   while(!spi_writeByte(NULL,LDC1000_CMD_RPMAX,0x00)); //3926.991k
209   while(!spi_writeByte(NULL,LDC1000_CMD_RPMIN,0x3F)); //   0.798k
210
211    /*  // less interesting
212    //while(!spi_writeByte(NULL,LDC1000_CMD_SENSORFREQ,  0x00));
213    //while(!spi_writeByte(NULL,LDC1000_CMD_LDCCONFIG,   0x00));
214    //while(!spi_writeByte(NULL,LDC1000_CMD_CLKCONFIG,   0x00));
215    //while(!spi_writeByte(NULL,LDC1000_CMD_THRESHILSB,  0x00));
216    //while(!spi_writeByte(NULL,LDC1000_CMD_THRESHIMSB,  0x00));
217    //while(!spi_writeByte(NULL,LDC1000_CMD_THRESLOLSB,  0x00));
218    //while(!spi_writeByte(NULL,LDC1000_CMD_THRESLOMSB,  0x00));
219    //while(!spi_writeByte(NULL,LDC1000_CMD_INTCONFIG,   0x00));   // this software is inten
220    */
221
222   while(!spi_writeByte(NULL,LDC1000_CMD_PWRCONFIG,0x01));
223
224
225
226   __delay_cycles(EVM_TIME_1MS * 2.5); // need this many cycles to settle
227
```

Figure 19 Enabling individual RP configuration in main.c

A wrong selection of RPmin and RPmax will cause the LDC1000 not to work as intended (no LC tank oscillation – no output data)

9. Hints

- COM Port problems using the MATLAB script:
  - If the script is not running properly (showing errors in the MATLAB Command Window) disconnect the Launchpad from the PC and then reconnect it.
  - Check the COM port number in the device manger. (it might have changed, might take some time until it appears after connecting the Launchpad)
  - Correct it if necessary in the script and don't forget to save it.
  - Now run the script again.

- Code Composer Studio problems with view – cannot find anything
  - if you miss some subwindows in CCS which were initially at … places – you might want to restore the default perspective.
  - go to Window -> Reset Perspective…
  - After answering the question with yes, CCS should look like opened the first time.

- Code Composer Studio missing line numbers:
    - Go to Window-> Preferences – a new window pops up
    - General->Editors->Text Editors – place a tick at Show line numbers

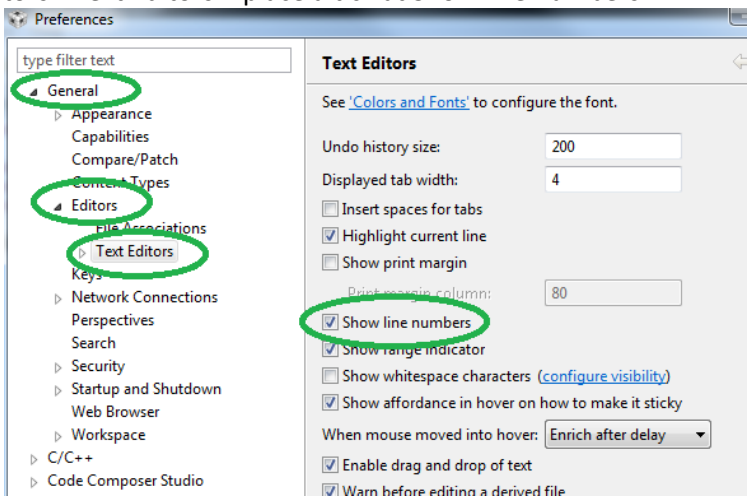

Figure 20 Preferences - (re)enable line numbers