Gezamenlijke opleiding Universiteit Hasselt en KU Leuven

2015•2016
FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN
master in de industriële wetenschappen: elektronica-ICT

Masterproef

Real-time optical range sensing application development

Promotor :
Prof. dr. ir. Luc CLAESEN
Dhr. GAETAN KOERS

Tom Nulens

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektronica-ICT

2015•2016

Faculteit Industriële ingenieurswetenschappen

master in de industriële wetenschappen: elektronica-ICT

Masterproef

Real-time optical range sensing application development

Promotor :
Prof. dr. ir. Luc CLAESEN
Dhr. GAETAN KOERS

Tom Nulens

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektronica-ICT

universiteit hasselt | KU LEUVEN

Foreword

During my thesis work at Melexis I worked on creating an evaluation kit for an distance sensing integrated circuit still in development from concept to physical device. In this I was supported by many enthusiastic Melexis employees who not only helped me with this specific task but also gave me insight about its greater context. They taught me how their distance sensing product works, how it has progressed and the ups and downs in their design process. I was fortunate to be present in the United States with the chip design team when they had just received the silicon wafers for the engineering samples. I have only positive memories from my experience with Melexis and I am very grateful for the many opportunities I was offered.

It is for this reason I want to express my gratitude to the people who helped me throughout the project. First I want to thank my external promotor, Gaetan Koers, for guiding me during my thesis and giving me these opportunities. Secondly, I want to thank Jelko Huysmans for helping me during the validation phase. I also want to thank the entire team in Nashua, NH, USA for sharing their expertise. Finally I want to thank my internal promotor Luc Claesen for the encouragement and help with finalising my thesis.

Table of contents

Tables

Figures

Vocabulary and acronyms

IC:	Integrated circuit
LED:	Light Emitting Diode
LIDAR:	LIght Distance And Ranging
MOSFET:	Metal–Oxide Semiconductor Field-Effect Transistor
PC:	Personal Computer
PCB:	Printed Circuit Board
PIN photodiode:	Photodiode with an intrinsic ‘I’ region in between the ‘P’ and ‘N’ doped regions.
SNR:	Signal to Noise Ratio
SPI:	Serial Peripheral Interface
USBHID:	Sub protocol in USB for Human Interface Devices

Abstract (Nederlands)

In de business unit Sensors van Melexis is een nieuw LIDAR product voor afstandsmeting in ontwikkeling. Het doel van deze thesis is om een evaluatie kit en demoapplicatie te ontwerpen die de eigenschappen en mogelijkheden van dit product demonstreert en dient als referentieontwerp voor klanten. Het product is een zestien kanaal optische afstand sensor die gebruikmaakt van het 'time of flight' principe. Het meet de afstand tussen zichzelf en een object door de tijd te meten tussen een light puls uit te sturen en het ontvangen van de reflecties van deze puls. De zestien kanaal ontvanger geeft de afstandmetingsensor de mogelijkheid om reflecties van zestien verschillende hoeken van elkaar te onderscheiden.

Het project begint met het onderzoeken van het sensorproduct en gaat verder met het bepalen van de vereisten van de evaluatie kit. De kit integreert de sensor, de lichtbron en een microcontroller in één module die kan communiceren met een computer. Het is een alles-in-één oplossing zoals die in de praktijk gebruikt zal worden. Rond de evaluatiekit wordt een applicatie gemaakt die afstandsmetingen uitvoert en toont deze in een grafische gebruikersinterface op een computer. Uiteindelijk worden de genomen metingen vergeleken met de theoretische specificaties om te zien of de module hieraan voldoet.

Op basis van de uiteindelijke hardware en software wordt een lijst van problemen en oplossingen opgesteld om toekomstige iteraties te verbeteren en om klanten die bezig zijn met hun eigen modules beter te helpen.

Abstract (English)

In the business unit Sensors of Melexis a new LIDAR sensing product is under development. The purpose of this thesis is to create an evaluation kit and demo application to showcase the capabilities of this sensor product and serve as a reference design for customers once it is ready to market. The product is a sixteen channel optical distance sensor based on the time of flight principle. It measures the distances between itself and objects by measuring the time between emitting a light pulse and receiving the reflections of this pulse. The sixteen channel receiver allows the product to differentiate reflections from sixteen separate angles.

The project is approached by first understanding the sensor's operation and the requirements of the evaluation kit. The kit integrates the sensor, the light source and a microcontroller in one module which can interface with a host computer. It is an all-in-one solution which mirrors real-life implementations. The completed evaluation kit serves as a base for the demo application. The application takes measurements and displays the distances in a graphical user interface on a host computer. Finally, measurements made with the evaluation kit are compared to the sensor's specifications to see whether it fulfills the expectations.

Based on the completed hardware and software, a "lessons learned" list is compiled to improve future designs and to enable better support to customers who are creating their own solutions.

1 Introduction

1.1 Background

Melexis is a Belgian semiconductor supplier which designs and manufactures integrated circuits and sensors. Some examples of their products are motor control integrated circuits, Hall Effect sensors and infrared optical sensors. Their main target market is the automotive industry [1]. The thesis project serves the application engineering team in the business unit Sensors at the Tessenderlo site and a chip design team in Nashua, NH, USA.

1.2 Task description

The goal of this thesis is to create an evaluation kit and demo application to showcase the capabilities of an upcoming distance ranging product, MLX75320, based on LIDAR (LIght Distance and Ranging) technology and serve as a reference design for customers once it is ready to market. The evaluation kit also functions as the hardware platform for the application. Due to this requirement the kit has to be very flexible and development has to be very fast so there is still sufficient time to design an application after debugging the evaluation kit.

1.3 Approach

The project starts by researching the sensor product. This research is very project oriented and is therefore limited to understanding the technology in general and the inner workings of this specific IC. With the knowledge gained in this phase the design requirements are compiled to use as checklist in the subsequent steps. This leads to the concept phase in which the actual application and the different components are defined. Next up is the design phase which consists of developing the hardware, writing the low level firmware and writing the high level application software. The design is then validated to finally end with a review and conclusion of the whole project. The structure of this thesis mirrors the structure of the project.

2 Research

This part of the thesis explains the inner workings of the MLX75320 sensor product and how it applies the time of flight principle with the purpose of finding all the relevant information to design hardware around it and develop the low level control software.

The MLX75320 is an optical sensor IC which can measure distance by using the time of flight principle. This is done by counting the time between emitting a single light pulse and receiving its reflections. This time is directly proportional to the distance between the sensor product and an object which reflected the pulse. Internally the IC consists of an array of 16 PIN photodiodes to detect reflections, digital circuits and a microprocessor and analogue circuits in between. The IC does not contain the light source. Figure 1 shows a picture of the package with a window for the photodiode array used for the IC.

Figure 1 Picture of the package of the MLX75320 sensor product [2]

Each photodiode has its own analog to digital conversion circuit which samples at 100MHz or 10ns per sample. Each sample cycle consists of 74 samples. At 10ns for each sample, the total time is 740ns. Multiplying this time by the speed of light gives the maximum distance at which a reflection can be detected, which is about 220m for the entire flight or 110m for the distance between the sensor product and an object.

$$s_{object} = \frac{c \cdot t}{2}$$

$$s: distance\ in\ m \ \Big|\ c: speed\ of\ light\ in \frac{m}{s} \Big|\ t: time\ in\ s$$

The sensor product has two important capabilities to improve its measurements. First it can oversample to increase its distance resolution. Without oversampling and at 10ns per sample the distance resolution is 1.5 meters. To improve the resolution the IC can apply oversampling by repeating measurements with a phase shift added to the 100MHz sample frequency. The phase can be shifted in steps of 1/8 of the sampling period: 1.25ns. So by repeating the measurement cycle 8 times, each with a different phase shift, a distance resolution of 18.7cm can be attained without any processing. Secondly, the IC can accumulate multiple sample cycles by summing the results together to increase the SNR (signal to noise ratio). By summing successive sample cycles the SNR increases by the square root of the number of sums. The IC supports accumulation factors up to 1024, which means the SNR can be improved by a factor of 32. Figure 2 shows a diagram of a full measurement cycle with oversampling and accumulations.

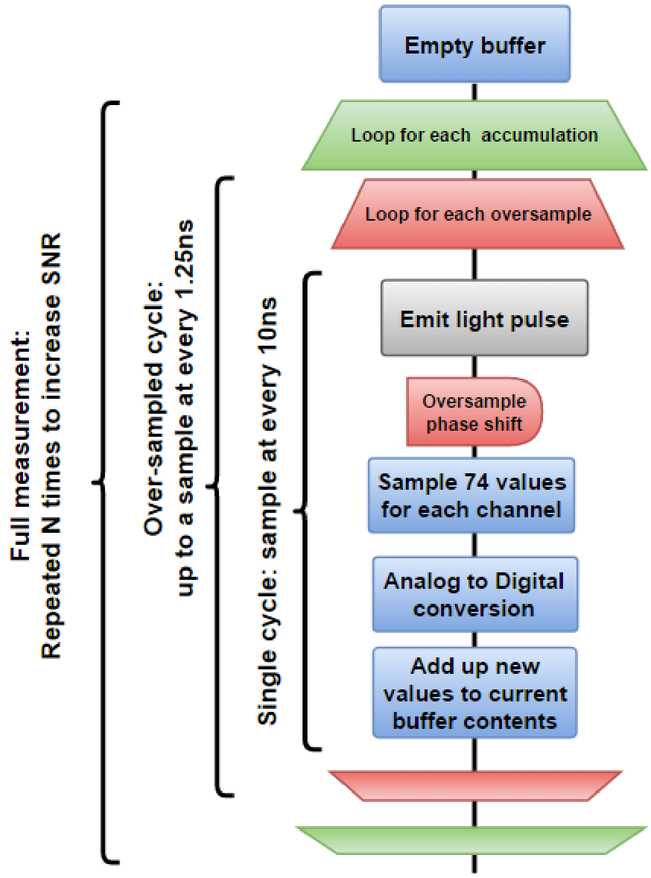

Figure 2 Diagram of a full measurement

After a full measurement cycle the IC has a digital trace buffer of the raw sample values. This buffer can be processed by the embedded microprocessor to calculate the distance. The embedded peak detection algorithm is capable of interpolating between samples to improve the distance resolution to about 5cm. After sampling and, optionally, after processing, the data is ready to be send to a host controller via SPI.

This section provided a high level overview about the inner workings of the MLX75320 sensor IC. Information like the trace buffer size provides the requirements for external processors while the measurement cycle is very relevant for the low level software controls. More information about the IC can be found in the specification or datasheet [2].

3 Requirement

The MLX75320 EVK is intended to be a flexible evaluation kit capable of demonstrating the full capabilities of the LIDAR sensor product and it must be able to work directly with the raw sensor data as well as the with processed distance data. Since the kit is created for evaluation, attention must be given to assure that the kit itself is flexible enough to allow applications to be built around it. In addition to these physical requirements, the development itself must advance at a fast pace to stay parallel with the advancements of the IC's design team.

The full application consists of the MLX75320 sensor product, a device to control the product via SPI and interface with a host PC and an illumination unit. The kit must be able to perform distance measurements up to distances greater than 10m, possibly do some additional processing and send the result to the host PC. The following requirements were set up to ensure the concept can fulfill this specification:

- Easy to use platform for fast software development;
- small and portable form factor;
- single kit with MLX75320, illumination, controller and optics;
- powered by a single external power supply;
- controller with built-in interfaces for both SPI and a host interface;
- controller fast enough for both SPI and host interface;
- controller to host interface with sufficient bandwidth for both raw and processed mode;
- illumination must have sufficient power to reach at least 10 meters.

While these are very rough requirements, they are sufficient to start an iterative design process where each requirement is fleshed out over time. The results of this process continues in the concept section.

4 Concept

The concept of the evaluation kit and application is created from the requirements list in the previous section. The flow of the application is as follows:

1. Host PC sends a command to the controller on the evaluation kit;
2. the controller on its turn sends SPI commands to the MLX75320 to initiate a measurement;
3. MLX75320 sensor IC sends signals to charge the LEDs and emit a light pulse;
4. MLX75320 samples light levels on its 16 channels to find reflections of the light pulse;
5. the results of the measurements are sent back by the controller;
6. the controller sends the results back to the host PC;
7. host PC visualizes the measurement in a graph;

Four components are described in this sequence: the host PC, the controller, the sensor IC and the LEDs or illumination unit. These are the main components of the application and are required for it to function. The host PC is any standard desktop computer or laptop and does not require extra attention. All the other components do need additional parts for them to work properly. The next subsection describes the three main and some peripheral components, their own high level requirements and the choices that were made to fulfill them.

4.1 Components

4.1.1 Sensor IC

The first component is the sensor IC. It controls the light source and contains the photo diode array to detect the reflections. The requirements are inferred from the preliminary datasheet [2] of the sensor. While it has multiple requirements for capacitors and crystal, only the supply voltages and the LED and photodiode for the functional safety feature of this chip are relevant at this stage. Functional safety is not within the scope of this project, however, so that part can be ignored.

To summarize the requirements:

- 3.3V analog and digital power supplies,
- 3.3V, preferably more for the bias voltage.

4.1.2 *Receiving optics*

In order to differentiate reflections from different angles, the MLX75320 sensor product requires a lens to bend the reflections from the emitted light pulse to the appropriate photo diode element.

The requirements of this receiving optical system are:

- small and light,
- matches size of the sensor's photodiode array,
- PCB mount.

4.1.3 Illumination unit

The next block is the illumination unit. This unit consist of the LEDs, charge circuit and switches controlled by the MLX75320 sensor IC. This component will be explained in detail in the hardware section. For now it is sufficient to know it requires a 12V supply. In addition to the circuity, the illumination also requires lenses like the sensor IC. The purpose of this lens is to emit light only within the angle of view of the lens over the sensor IC. Any light emitted outside this angle would be wasted.

Summary of the requirements:

- 12V power supply
- Optics to ensure the angle of illumination matches the sensor's angle of view

4.1.4 Controller

The final main component is the controller. It serves as the interface between the MLX75320 sensor IC and the host PC. To be able to serve as interface it should be able to match the MLX75320's SPI clock of 25MHz and capable of transmitting raw measurements of about 19kByte at 30 frames per second back to the host via either Ethernet or USB. In addition to this it should be able to do some processing at the same time in case a customer wants to use the evaluation kit in a standalone mode without host PC.

Besides the features mentioned above and the obvious power supply, the controller also requires a programming interface to program the custom firmware to the chip.

This brings us to the following requirements:

- 25MHz SPI
- Full speed USB or Ethernet
- Over 19kByte of RAM
- Programming interface
- Power supply

Due to the addition of a programming interface the controller requirements are more difficult to achieve than for previous components. Since the speed of development was also an important requirement, this block has to be streamlined in some way. An obvious way is to use a development board instead of a free standing FPGA or microcontroller. A single board computer like the Raspberry Pi was considered but it could not be confirmed whether their SPI interfaces would be sufficiently fast due to their operating system overhead. A viable candidate was found in the mbed LPC1768 development board, especially since this board was used in a previous project and an existing PCB with power supplies can be reused. This PCB, the base board, can also supply power to the other components.

4.1.5 Temperature sensors

This subsection ends with a previously unmentioned component. While there is nothing which requires temperature sensors in this evaluation kit, based on previous experience it seemed useful to include temperature sensors near the LEDs and switches for future characterization tests. The sensors are all controlled via I^2C and they require a 3.3V supply. But these requirements do not impact the rest of the design.

4.2 Block diagram

Combining all components in one diagram creates the figure below. This diagram forms the base of the hardware design and clarifies the communication flow used in the firmware and application software.

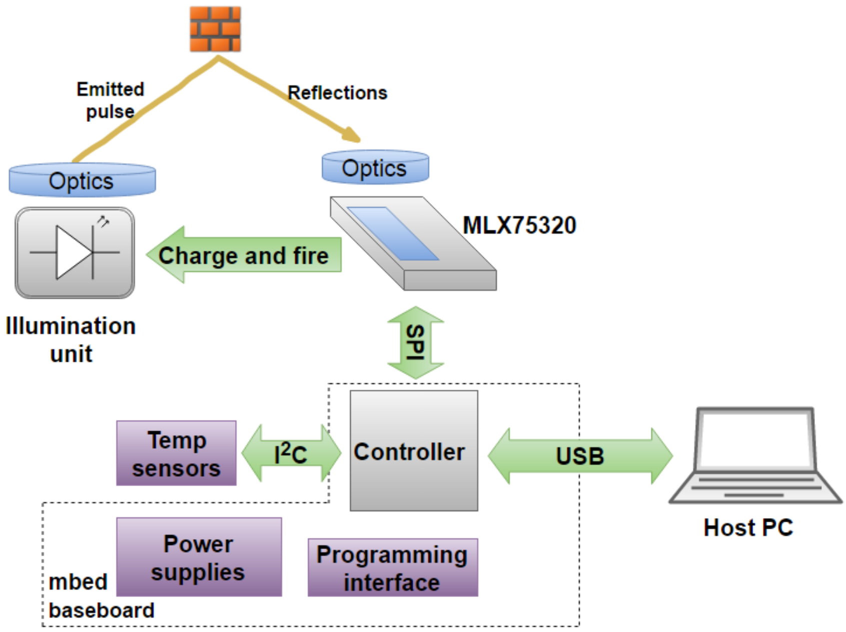

Figure 3 Block diagram of evaluation kit concept

5 Hardware

This section describes the hardware design of the evaluation kit. It consists of the microcontroller base board and the sensor board. The base board holds an mbed developer board and power rails for 3.3V and 12V while the sensor boards has the MLX75320 sensor IC, illumination unit, temperature sensors and optics.

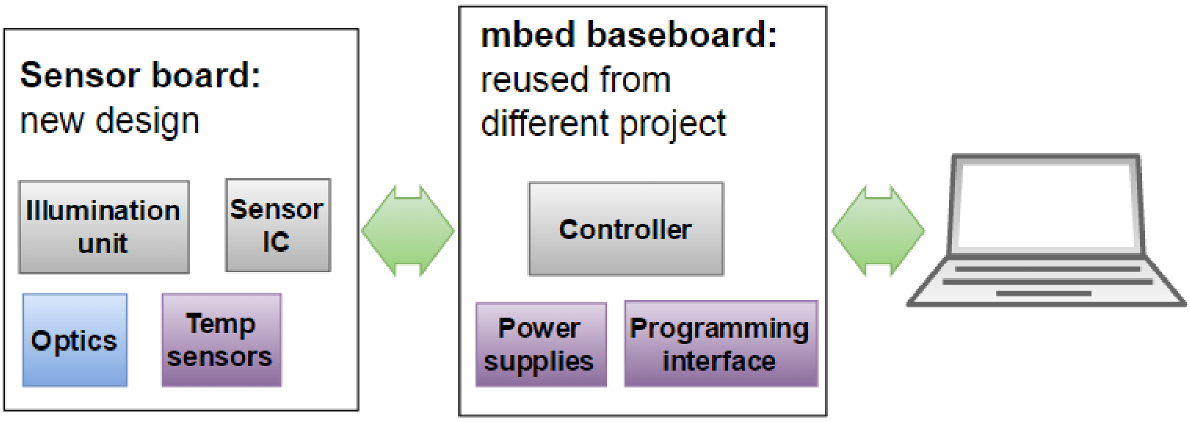

Figure 4 Diagram of the different boards

5.1 Base board

As mentioned in the concept section, the mbed base board is a PCB with a header to hold an mbed LPC1768 development board with some peripheral parts. The complete board consists of the following parts:

- 3.3V linear voltage regulator (capable of 12V input)
- USB connector
- GPIOs, SPI and I²C pin connectors
- LPC1768 development board

The pin and USB connectors connect to the relevant pins from the development board [3]. The important features of the development board are as follows:

- 96MHz ARM LPC1768 microcontroller
- 32KB RAM, 512KB FLASH
- USB Host/Device, SPI, I²C, GPIO
- Built-in USB drag 'n' drop FLASH programmer
- Online C/C++ programming environment and compiler

The 32kBytes of RAM memory gives a sufficient margin for the 19kByte trace buffer, the FLASH capacity is also large enough to contain both the firmware and patches for the MLX75320 IC. The USB host/device is used as device and is connected to the USB connector on the base board, the other pins for SPI, I²C and GPIO are passed to the sensor board via connectors and finally there is an onboard USB connector used for loading files on the FLASH memory. This USB connector can also serve as virtual serial port for debugging.

The mbed platform uses an online programming environment and includes many pre-existing libraries to speed up development. Figure 5 shows a picture of the base board on its own.

Figure 5 Picture of base board with mbed development board mounted

5.2 Sensor board

In contrast to the base board, the sensor board is a completely new design. It takes power and interface signals from the base board and uses them to do distance measurements. As seen in the earlier figure 4, the sensor board has four components: The illumination unit, sensor IC, temperature sensors and the optics. Of these, the sensor IC and temperature sensors are single ICs with some additional components specified in the datasheets. Since the information in the datasheets are sufficient to utilize them, the thesis only explains the illumination unit and the optics.

5.2.1 Illumination unit

The illumination contains LEDs, a capacitor bank, two MOSFETs and two MOSFET drivers. Figure 6 shows the circuit. The following sequence describes how the circuit works:

1. PWM signal from the MLX75320 on the high side MOSFET driver switches MOSFET Q1;
2. As Q1 switches a current flows to the capacitor bank to charge it up and increase the voltage;
3. With the desired voltage on the capacitor bank, the MLX75320 sends a pulse to fire the LEDs;
4. The pulse is detected by the low side MOSFET driver and the MOSFET Q2 closes the circuit;
5. The capacitor bank discharges through the LEDs with a very large peak current;
6. The high peak current corresponds with a strong light pulse;
7. The cycle repeats for each over sample and accumulation.

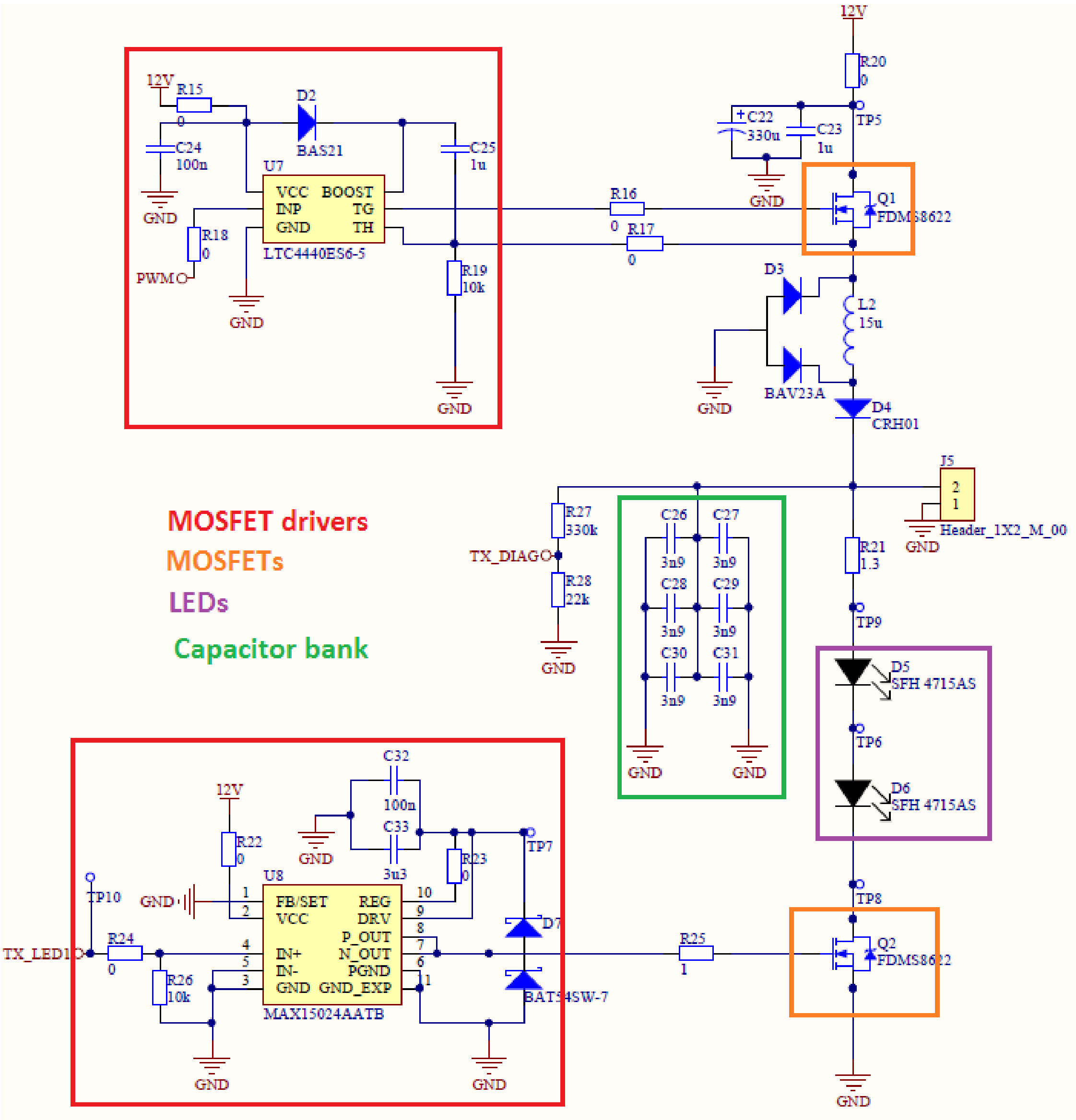

Figure 6 Illumination unit circuit

The whole illumination circuit uses a 12V supply. This is also the maximum voltage of the capacitor bank. By varying the amount of PWM pulses in the charge cycle, the voltage can vary between the forward voltage of the LEDs and 12V. This current, and therefore also the light strength, is linear with this voltage. The inductor L2 smooths the peaks of the charge current.

5.2.2 Optics

Two sets of optics are used on the sensor board. First there are the receiving optics over the MLX75320 sensor product which bend light coming from various directions to corresponding photodiodes. Secondly there are the emitting optics to shape the beam from the illumination unit so it matches the angle of view of the receiving optics.

The receiving optics decide the angle of view of the sensor product. The desired angle of view for this evaluation kit is between 40 and 50 degrees. The dimensions of the photo diodes array are 11.1mm x 0.7mm [2]. The next formula is used to calculate the angle of view for a given sensor size and focal length.

$$angle\ of\ view = 2 \cdot \tan^{-1}\frac{d}{2 \cdot f}$$

$angle\ of\ view\ in\ rad \mid d{:}\ length\ of\ dimension\ in\ mm \mid f{:}\ focal\ length\ in\ mm$

A focal length of 15mm translates to an angle of view of 41° x 9°, which is ideal. C-mount compatible lenses of this focal length are readily available. To limit costs a single aspheric lens with a diameter and effective focal length of 15mm is chosen. A C-mount is also fitted over the MLX75320 to screw in the lens.

Now that we know the angle of view of the receiving optics, the emitting light pulse has to be shaped to match this angle. Elliptical lenses are chosen to bend the symmetric 90° degree full angle of the LEDs to a slightly larger angle of about 50° x 10°. Such elliptical lenses for the LEDs in the evaluation kit are readily available from suppliers like Khatod [4].

5.2.3 Sensor board PCB layout

Below is the PCB layout of the top and bottom layer of the sensor board. The 3.3V and GND planes are not visible. Both planes have their own layer. Two large circular mounts can be seen around the LEDs and a large rectangular C-mount around the MLX7530 IC.

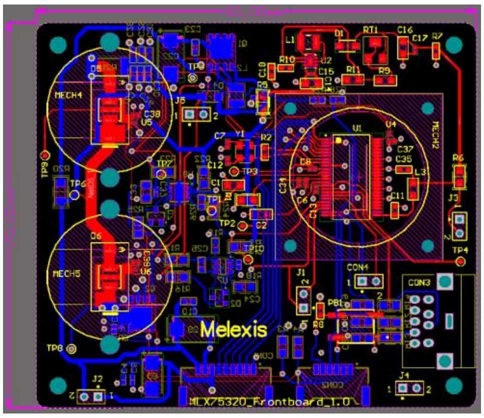

Figure 7 PCB layout of sensor board: Top layer in red, bottom layer in blue

5.3 Concluding

This concludes the hardware design phase. With the hardware of the evaluation kit complete the next steps can begin: Writing the software and validating the design. Figure 8 shows a picture of the complete evaluation kit: Base board PCB on the bottom, sensor board on top.

Figure 8 Evaluation kit hardware

6 Firmware / API

Firmware refers to the lower level software running on the microcontroller which controls the MLX75320. Since it was decided to run the application software on a separate PC the entire microcontroller functions solely as interface between the sensor product and the host PC. To achieve this, the microcontroller has to implement both SPI to the MLX75320 and a USB interface to the host PC.

6.1 SPI protocol

The base for the SPI link is the standard SPI library from the mbed platform [5]. A new library was written which utilizes mbed's SPI class and implements the SPI protocol defined in the MLX75320 specification requirement on top of it. All functionality is encapsulated in the class LidarSpi. The protocol consists of short or long data packets [6]. The short packets are four words long while long packets go from three to 150 words. Each word is 16 bits. Of each packet the first word is the header which tells what type of packet it is, the payload size and in some cases a sequence number. The final word is always the CRC (Cyclic Redundancy Check) code to verify whether the packet was sent correctly. The packet structures are displayed in figure 9.

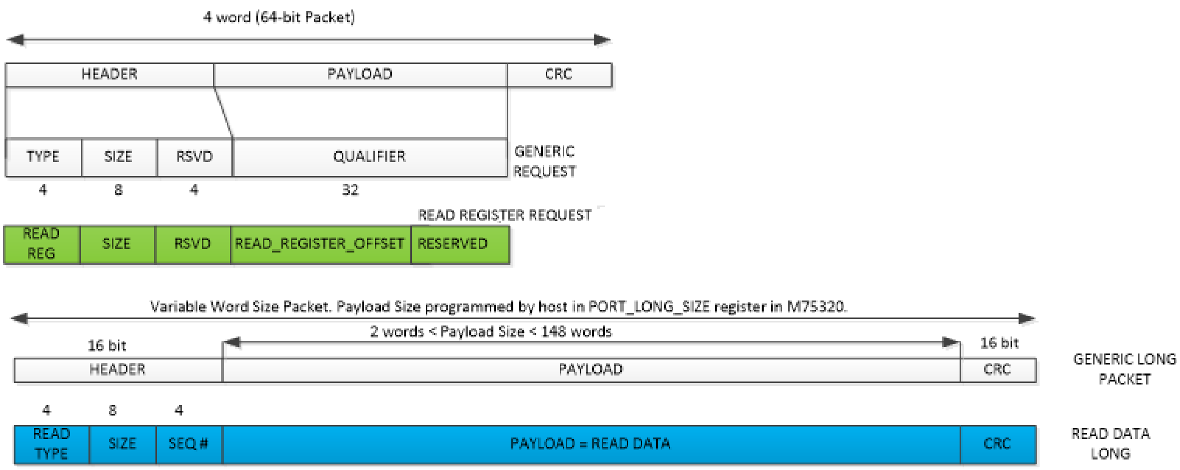

Figure 9 Contents of short and long packets with a short packet example in green and a long packet example in blue

To execute a function like reading the value in a register multiple packets have to be sent and received. With each packet that is sent by the microcontroller, the sensor product will also send a packet back at the same time. The first packet sent by the microcontroller is always a request packet to initiate a certain function; for example reading the contents of register 0x14c. At the same time the MLX75320 sends a status packet to indicate whether anything went wrong previously. The next packets depend on what the request was, usually either a data or a status packet. Figure 10 shows an image with the packet sequences to read and write to registers and to read a complete measurement.

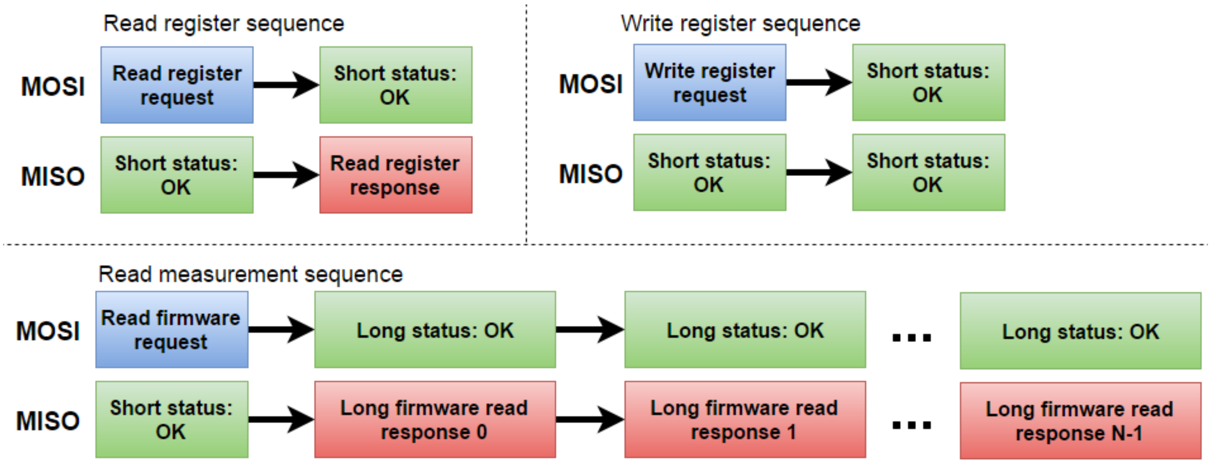

Figure 10 Sequences of SPI packets to perform read/write register commands and to read a measurement

If something goes wrong during the SPI communication a status packet is send with an error code and the communication is aborted. It is therefore vital to decode and verify each packet as it is received. Figure 11 displays a flowchart of a single SPI exchange.

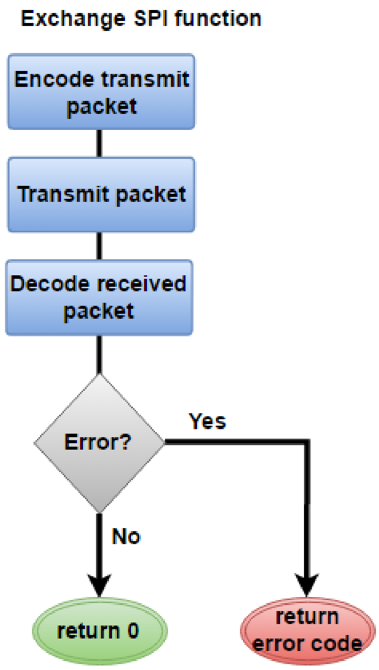

Figure 11 Flowchart of a single SPI packet transfer between microcontroller and MLX75320

These single packet transfers are chained together in methods to create the desired functionality. The main methods in the LidarSpi class are:

- Read a register value;
- write a register value;
- execute a measurement and read the raw values;
- execute a measurement and read the processed data;
- load a patch file saved on the flash of the microcontroller to the MLX75320.

Together, all functionality of the sensor product can be accessed.

6.2 USB interface

The USB interface is the main loop of the firmware. It receives commands from a host PC, executes the required methods from the LidarSpi class and sends a response back to the host PC. Sending and receiving messages is handled by the USB libraries provided by mbed. Two USB protocols have been tried:

- USBSerial, a simple virtual serial port compatible with STD functions like 'printf' and 'getc';
- USBHID, polls packets of up to 64 bytes at up to 1000Hz via an interrupt routine

The final version uses USBHID due to the higher available bandwidth. Figure 12 shows the flow of the main loop.

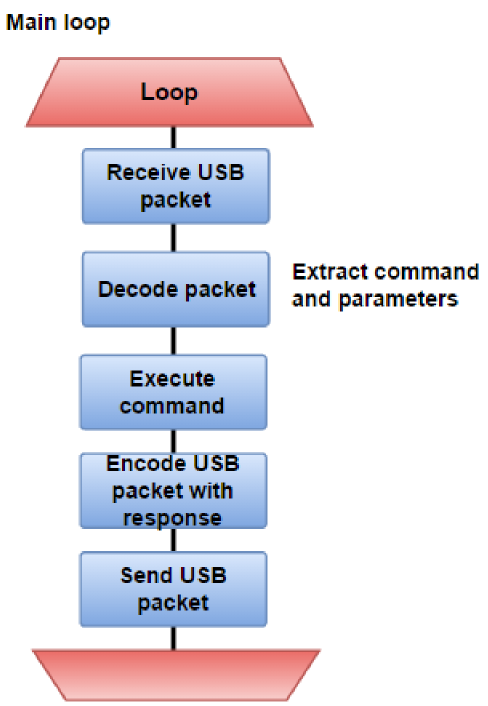

Figure 12 Flowchart of USB interface

The microcontroller expects just a string inside the received USB with the first word representing the command followed by the parameters in hexadecimal form separated by spaces. Here are a few examples:

- "RREG 14C " : Read the value at register 0x14C and return the value;
- "WREG 14C FFF " : Write the value 0xFFF to register 0x14C;
- "TRCE 0 " : Setup the registers for a measurement, do the measurement and return the raw trace buffer.

The reason strings are used is because these can be easily decoded in one line with the C function 'sscanf' and the commands themselves do not have to provide much data. In contrast to the receiving packet a response can contain much more data; up to 18kByte. For this reason the response is not encoded as a string but as binary data. The response starts with a five byte header and is followed by the data. The first byte is the status byte which is zero if no error occurs while the next four represent the number of bytes the number of data bytes which will follow. The number of data bytes can far exceed the limit of 64 bytes for a USBHID packet, in this case the data is spread over multiple packets but only the first packet has a header. A generic packet and three specific response examples are shown in figure 13. Notable is the "write register response" which does not have any data and the "processed measurement response" of which the data is split over seven packets.

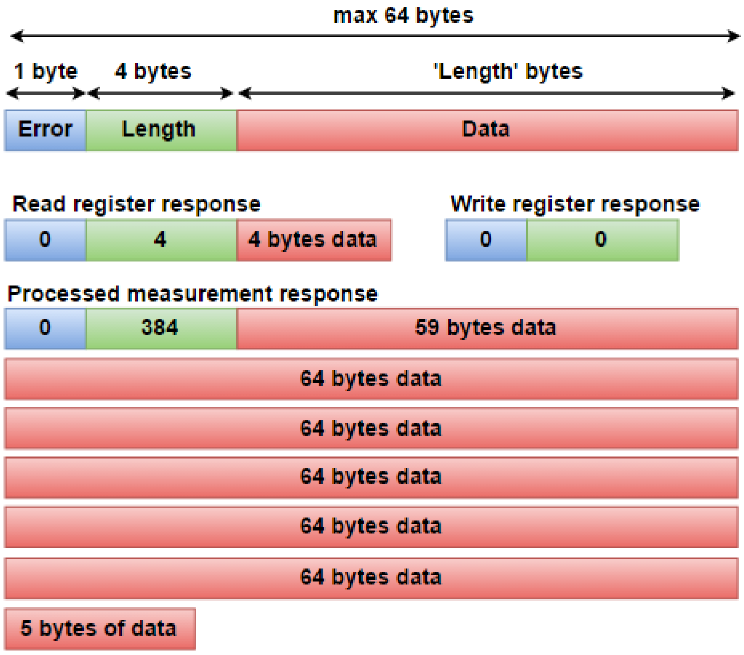

Figure 13 USB response packet examples.

None of the USB packets have any CRC code for error checking. This functionality is already included in the low level USB protocol. Should an error occur in the transmission the used USB library resends the packet automatically.

This concludes the firmware. Utilising the existing mbed libraries a simple to use USB to MLX7530 API has been created. Due to the standard USBHID packets, a host application can interface with the sensor product in any high level programming language with existing libraries for USBHID (C++, Java, Python, C# ...) without any special drivers [7].

7 Software

This part of the thesis expands on the application software which runs on the host PC. For this project Python was chosen as the programming language due to prior knowledge, many pre-existing libraries and compatibility with the test scripts used by the Melexis Nashua development team. The software consists of a class which handles the interface to the evaluation kit and the actual application software. Communication between the host PC software and the microcontroller firmware is always initialized by the host.

7.1 Host side API

The host side API has the reverse organization of the USB interface as seen in the firmware. Instead of decoding a string for a command and encoding data in binary form, this API creates the string and decodes the binary data. All the API commands are implemented as methods in a Python class which utilize the Python module 'pywinusb' to send and receive USBHID packets. Figure 14 shows a flowchart of the implementation of a generic API command. At first, a command is sent to the microcontroller as a text string. The method then waits until the microcontroller sends a response back. This response is received by a callback function which checks the packet for errors and to see if more packets have to be received before the API command can continue. Once all packets are received the API command can fit the received data in the required data structure.

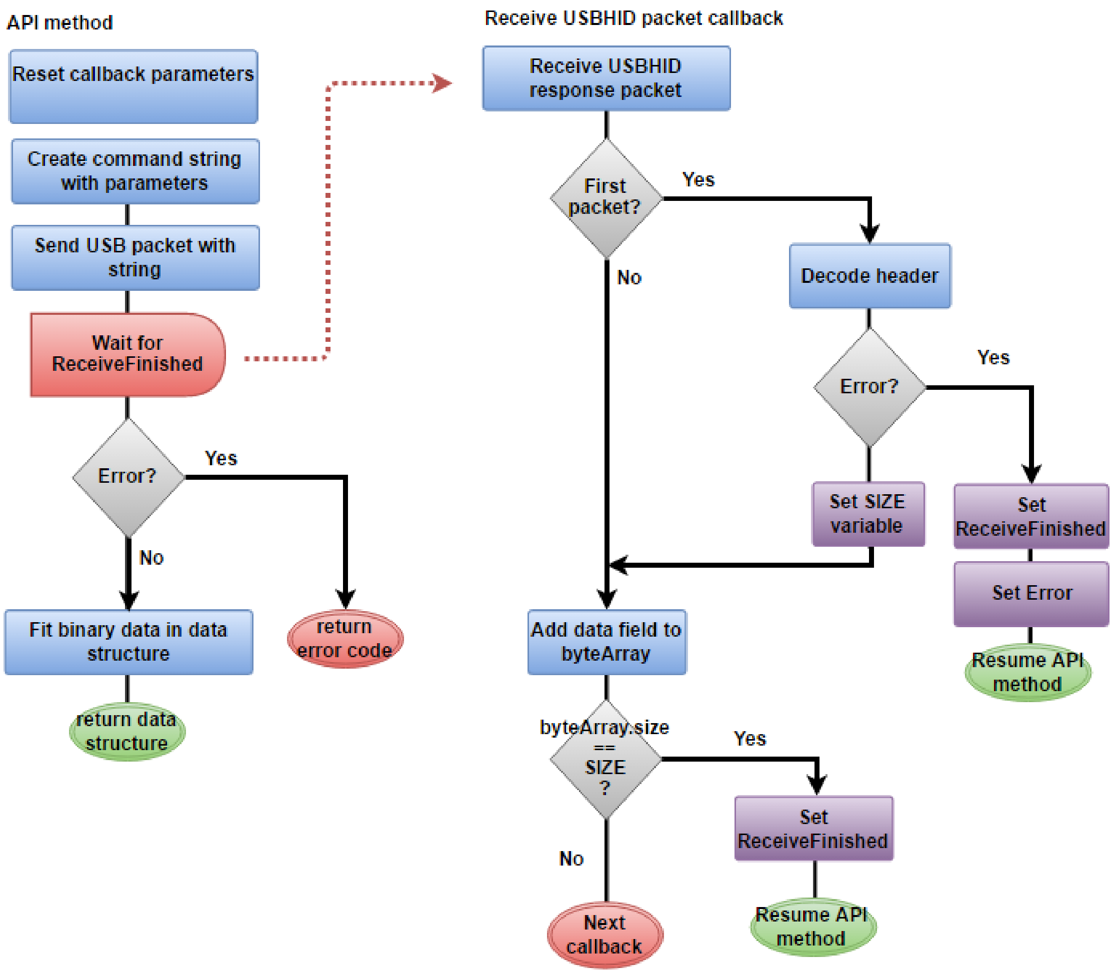

Figure 14 Flowchart of python API command

7.2 Application

The purpose of the application software is to continuously read the raw trace buffer of the sensor product, split the trace buffer in its 16 channels and display all those 16 channels in a plot. Before this loop is executed all the necessary registers are set and an offset cancellation routine is called while the sensor product is obscured. The offset cancellation removes the periodic ripple at the start if each measurement which is caused by firing the LEDs. The flowchart of this code is displayed in figure 15.

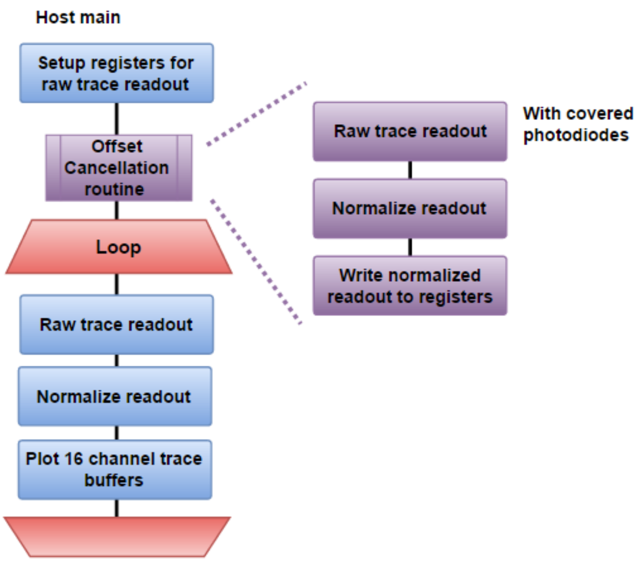

Figure 15 Flowchart of host PC application

The application script provides the perfect environment to validate the functionality of the MLX75320 evaluation kit. Because a raw trace buffer shows the light pulse reflections as detected by the photodiodes with little further processing, problems caused by the peak detection algorithm can be excluded when validating the general functionality. Figure 16 shows a few plots of measurements with an object at different distances

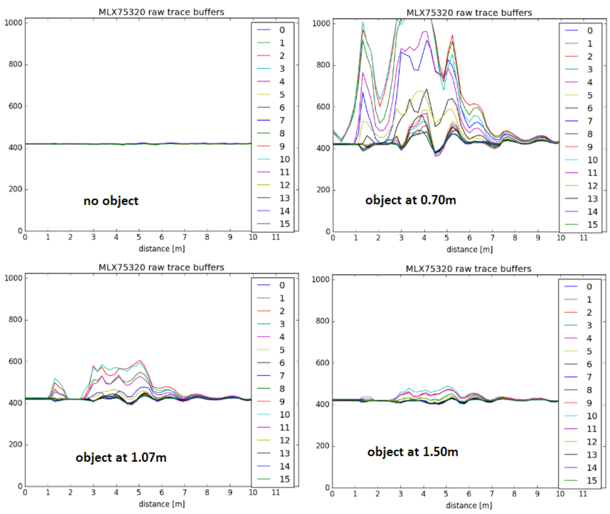

Figure 16 Plots drawn by application of object at multiple distances

8 Validation

This final section describes the validation of the evaluation kit and hardware.

The first step of the validation starts right after completing the hardware. The entire PCB is checked for short circuits, faulty connections and similar issues. The PCB layout simplifies this by allowing entire components to be shutdown by not fitting zero ohm resistors between each component. This way each part can be disconnected from the power supply and other parts.

Once the hardware is validated, the same can be done for software and functionality. The process here goes a bit back and forth: Software is required to enable certain functions and the software can only be properly tested when these functions work as expected. Figure 17 shows the work flow in a block diagram.

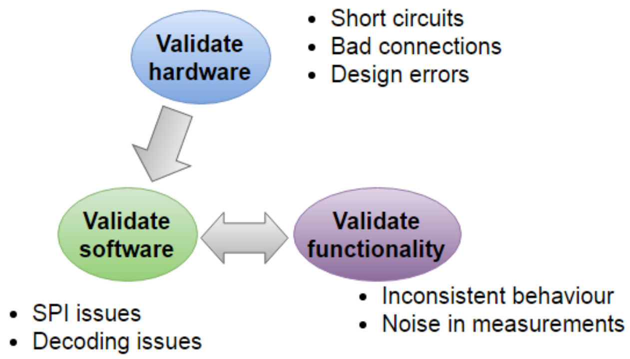

Figure 17 Validation flow

During this process a list of hardware, software and functional issues is compiled along with their root cause and possible solutions.

8.1 Validation results

Table 1, 2 and 3 give an overview of the various issues and how they were solved. Table 1 is about the hardware issues. The solutions here are workarounds until a second iteration of the PCB design solves these. Table 2 deals with the software issues and were mainly problems with the SPI interface. Table 3 gives a view of the functional issues. The biggest issues here are innate in the current chip iteration of the MLX75320, workarounds lessen the impact it has to enable better measurements.

Table 1 Hardware issues and solutions

Observed issue	Root cause	Workaround / solution
MLX75320 resets every X milliseconds	Thermal safety feature of 3.3V LDO regulator on base board	Disable LDO and supply 3.3V from an external source
High side MOSFET driver not switching MOSFET	Wrong pin assignments of footprint in PCB layout	Cut and patch traces on PCB to make correct connections
Bias voltage's boost converter always outputs 8V instead of variable voltage	Wrong pin assignments of footprint in PCB layout	Disable boos converter and attach external bias voltage
Temperature sensors too small to solder manually	4 pin BGA footprint	Leave unpopulated.
LED not firing during measurement	MOSFET driver threshold voltage higher than specified in datasheet	Order various variants of the driver to find a 3.3V compatible variant and patch PCB

Table 2 Software issues and solutions

Observed issue	Root cause	Workaround / solution
MLX75320 enters functional safety lockdown when loading patch in memory	Loading the patch in pieces	Load the patch all at once
USBSerial too slow for raw measurements	Limitation of mbed library	Use USBHID instead
Received SPI data is shifted by one bit	Mbed SPI library too slow for 24MHz	Lower SPI frequency to 16MHz
Raw measurement data appears random over its entire 16 bit range	Most and least significant byte are swapped inside the MLX75320	Swap MSB and LSB in microcontroller software

Table 3 Functionality issues and solutions

Observed issue	Root cause	Workaround / solution
Capacitor bank peak voltage drops after X accumulations	Remaining charge of previous LED fire interferes with the functionality of MOSFET driver	Balance capacitor bank size, LED pulse width and LED current to completely discharge capacitor bank after each fire.
Ripple in raw trace	Internal noise in MLX75320 from sampling	Change sampling delay
Ripple in raw trace	Internal noise from PWM charge signals	Offset cancelation
Ripple in raw trace	Internal noise from firing the LED	Offset cancelation
Pulse position in raw trace not linear with distance of object	Limitation of current IC design: Nonlinear behavior at low distances	

Very short detectable distance (<5 meter)	Interference ripple larger than signal at >5 meter	Boost LED power

9 Conclusion

In this thesis an evaluation kit was developed from concept to first iteration. Here is a summary of the achievements:

- First iteration evaluation kit hardware combining new and pre-existing PCBs;
- Firmware to convert low level SPI to a high level USBHID interface;
- PC software which utilises the high level USBHID interface to perform measurements and display the results in a graph.

Unfortunately the raw measurements are sensitive to interference. This interference comes from both limitations in the current MLX75320 IC iteration and from problems in the electrical and optical designs. These issues are too severe to do accurate measurements and have to be solved in the next iteration. The final few months of this thesis were spend trying to reduce these issues instead of expanding the functionality of the application software.

In the future a new PCB iteration with stricter design rules will reduce electrical inference. The optical design will also be renewed to reduce reflections coming from the PCB itself. At the same time the features of the application software are expanded to improve the visualisation and to allow users to change various settings while the application is running.

Since the end of May 2016 the redesign is being done by the Melexis engineer Jelko Huysmans and a new thesis student, Vianney Payelle, is using my API to create the improved graphical user interface. Figure 18 shows a screenshot of a new user interface which displays the contents of the trace buffer. The radius of the graph corresponds with the distance of an obstacle, the angle with the angular position and the colour corresponds with the amplitude of the received light.

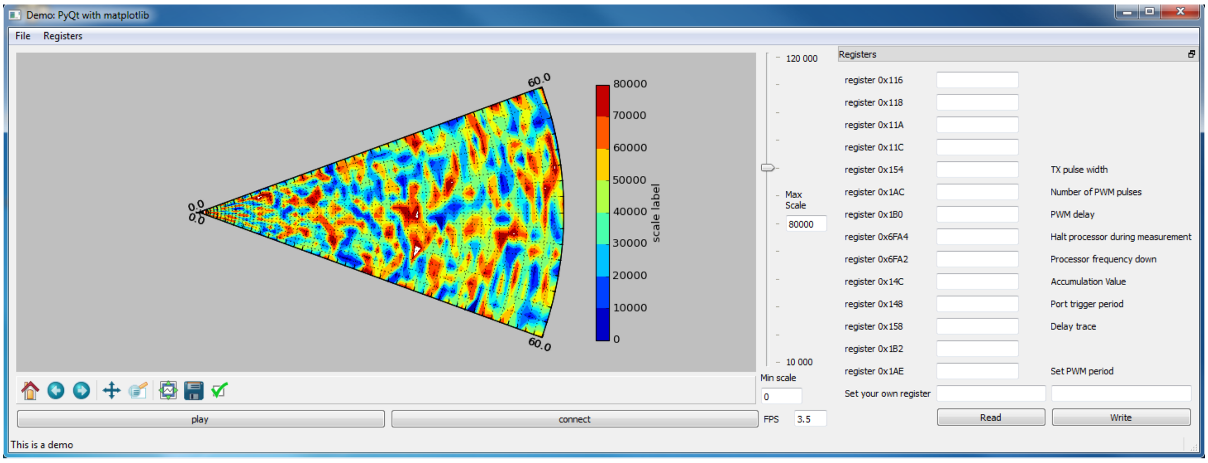

Figure 18 Python user interface by Vianney Payelle [9]

References

[1] “Melexis company profile,” Melexis, [Online]. Available: http://www.melexis.com/company.aspx. [Accessed November 2015].

[2] GOE, “Picture of IC package,” Internal Melexis picture, 2016.

[3] MIZ et al, “MLX75320_ASIC_Specification_Rev1.8,” Internal Melexis document, Nashua (NH), 2015.

[4] “mbed LPC1768,” June 2016. [Online]. Available: https://developer.mbed.org/platforms/mbed-LPC1768/.

[5] [Online]. Available: http://www.khatod.com/Khatod/home.aspx.

[6] “mbed handbook SPI,” June 2016. [Online]. Available: https://developer.mbed.org/handbook/SPI.

[7] BSI, “75320_SPI_AppNote,” Internal Melexis document, Nashua, NH, 2016.

[8] June 2016. [Online]. Available: https://developer.mbed.org/cookbook/USBHID-bindings-.

[9] VPE, “MLX75320 GUI progress screencap,” Internal Melexis picture, 2016.

[10] T. Dunigan, “LPC1768 Maximum frequency SPI,” June 2016. [Online]. Available: https://developer.mbed.org/forum/helloworld/topic/2498/.

[11] “Calculator for Field of View,” June 2016. [Online]. Available: http://www.scantips.com/lights/fieldofview.html.

[12] “Understanding Focal Length and Field of View,” June 2016. [Online]. Available: http://www.edmundoptics.com/resources/application-notes/imaging/understanding-focal-length-and-field-of-view/.

[13] “mbed handbook SerialPC,” June 2016. [Online]. Available: https://developer.mbed.org/handbook/SerialPC.

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Real-time optical range sensing application development

Richting: **master in de industriële wetenschappen: elektronica-ICT**
Jaar: **2016**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Nulens, Tom

Datum: **4/06/2016**