Jacob Honer

A comprehensive comparison between the NXP LPC1125 and the NXP KL25Z128 Microcontrollers from NXP Semiconductors regarding Analog and Serial Communication, as well as Suitability for a Wearable Health Monitoring Application

ECE 331 Research Project 3

December 15th, 2020

Introduction

This paper will provide a surface level analysis of the analog and serial communication modules of the NXP LPC1125 Microcontroller (MC), as well as offer comparisons between the NXP LPC1125 MC and the NXP KL25Z128 MC with respect to both their analog and serial communication functionality. Following said analyses and comparisons, this paper will juxtapose both MCs and compare their utility for being the MCs of a wearable health monitoring system. This conclusive analysis will include a discussion of each MCs memory, GPIO, Timers, analog, and serial communications. All information for this report pertaining to the NXP LPC1125 and the NXP KL25Z128 were gathered from their respective User Manuals, taken directly from the NXP website [1], [2].

Analog Module

A. Overview of analog functionality

Regarding the analog peripherals of the NXP LPC1125 MC, this MC only comes with one analog peripheral in an analog-to-digital Converter (ADC). This MC has no built in digital-to-analog converter (DAC) or separate comparator; the ADC does have a grouping of 4 threshold registers that allow for comparison, inducing two Compare Low Threshold Registers and two Compare High Threshold Registers. The ADC on this MC is a 12-bit successive approximation ADC, and its features include input multiplexing among 8 pins, optional automatic high/low threshold comparison and "zero crossing" detection, an ADC measurement range from a preset negative reference voltage to a positive reference voltage, which typically cannot exceed the set analog supply voltage, which is typically 3 volts, and lastly, a burst conversion mode that can be used for single or multiple inputs. The built in ADC of the NXP LPC1125 has a 12-bit conversion rate of 2 MHz and two configurable conversion sequences with independent triggers that will be discussed later. There are a total of 12 ADC channels, 8 of which are pin connected, and 4 other miscellaneous channels, totaling 12 ADC total channels.

To configure the ADC of the NXP LPC1125, first the clock of the ADC must be enabled in the system clock control register on the MC by setting the corresponding bit to high. The ADC block can create up to four interrupts, a threshold crossing interrupt, and an overrun interrupt, and two end-of-sequence interrupts for each respective conversion sequence. The ADC analog inputs are selected as such on the IOCON block of the NXP LPC1125 and totals up to 8 different inputs; as summarized in previous reports, these IOCON registers would be the pin control registers (PCR) of the NXP KL25Z128 MC and the MUX selections bits of the PCR would parallel the selection bits of the IOCON registers. The ADC block diagram can be found in figure 1. Also notable is that the power of the ADC on the NXP LPC1125 is also configurable, and calibration of the ADC is required after every reset which includes power-up or wake-up from Deep power-down mode. Also, a fully accurate conversion requires 25 ADC clocks, which can of course be configured similarly to any clock configuration discussed in previous reports.

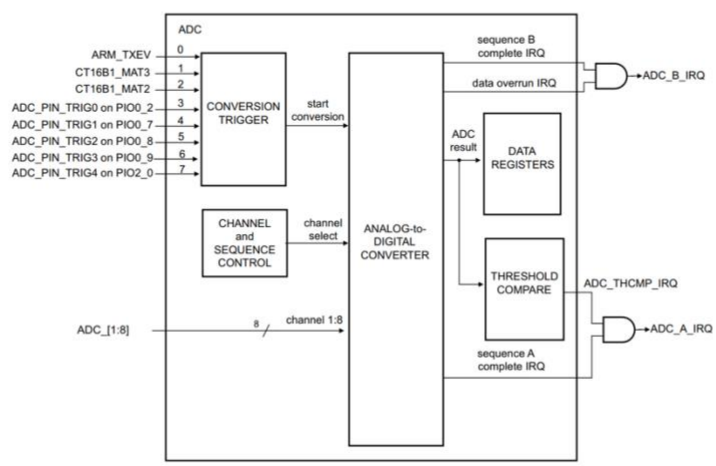

Figure 1 – The ADC block diagram for the NXP LPC1125 MC [1].

ADC can be initiated by a hardware trigger by 5 of the different pins on the MC. When triggered, the ADC conversion is started, and the results are stored in each respective channel's register. The conversion sequences registers can be a helpful tool as they allow for passing through multiple conversions from multiple channels to one global register. This process occurs when the sequence is triggered, and does so in a sequential order, starting with the lowest ordered channel. These conversion sequences can also be helpful, as they have a configurable single-step mode, which allows for advancing through each channel, one at a time, on each successive occurrence of a trigger.

The last interesting faucet to discuss of the ADC of the NXP LPC1125 is its built-in, self-calibration mode, which is required after every chip reset and allows for the ADC to achieve its specified accuracy. It is recommended to perform calibration on a periodic basis if possible. A calibration tends to take about 290 µs to complete, and while calibrating, there is basically a freeze on normal ADC conversions, and the ADC Control Register must not be written-to.

There are more details regarding the ADC of the NXP LPC1125 MC, but they will not be discussed further as they are outside of the scope of this paper. More details can be found in the NXP LPC1125 user manual [1].

B. Comparison of the NXP LPC1125 to the NXP KL25Z128 regarding Analog

Regarding analog functionality, the NXP KL25Z128 is almost certainly more diverse and utilitarian. The NXP KL25Z128 comes with a built in 16-bit successive-approximation ADC module, a single comparator that can trigger ADC acquisitions, timer/pulse width modulation updates, or CPU interrupts, and then two ADCs, one 6-bit and one 12-bit. Clearly the NXP KL25Z128 comes more equipped, as the NXP LPC1125 doesn't have a separate comparator or any ADC.

With that said though, the ADC of the NXP LPC1125 does come with its threshold compare register, built into the ADC, hence, the NXP LPC1125 is not left with no comparative functionality. The comparator of the NXP KL25Z128 compares an external analog voltage on a pin to either another external analog voltage on a pin, or an internal reference voltage, producing a high or low result. The ability to compare analog voltages is convenient; to do the same using the NXP LPC1125 would require an ADC conversion and then a comparison, which presumably is a much slower process than immediately comparing analog to analog. Both the NXP LPC1125 ADC and the NXP KL25Z128 comparator take in 8 selectable inputs, but the NXP KL25Z128 comparator allows for inverting and non-inverting inputs, as well as two internal reference voltages, compared to the NXP LPC1125 ADC's one reference voltage and inability to invert inputs.

Comparing the ADCs of each respective MC, the NXP LPC1125 has up to 12 input channels, compared to the NXP KL25Z128's 7 to 14 channels. Both can generate multiple hardware and software interrupts to be expected. The NXP KL25Z128 comes with a single-ended compare option, which compares to the low voltage reference, similar to the compare methods of the NXP LPC1125, as well as a differential option, which outputs a result of the difference of two inputs. The latter cannot be found on the NXP LPC1125 and could only be achieved with software, being less efficient.

In the end, regarding analog functionality, it can be concluded that almost without a doubt the NXP KL25Z128 beats out the NXP LPC1125 in terms of utility, diversity of choice, and functionality.

Serial Communications Modules

A. Overview of serial communication functionality

The NXP LPC1125 comes equipped with three universal asynchronous receiver-transmitters (UARTs) with fractional baud rate generation, and internal first in first out (FIFO), and RS-485 support; one of the UARTs (UART0) is equipped with modem control. The NXP LPC1125 also has two Synchronous Serial Ports (SSPs) with FIFO and other multi-protocol capabilities, and lastly, and I2C-bus interface that supports a full I2C-bus specification, as well as a fast-mode plus that has a date rate of 1 megabits per second with multiple address recognition and monitoring.

This paper will first explore notable features of the NXP LPC1125 MCs UARTs. Given an in-depth analysis of the UARTs on the NXP LPC1125, there does not appear to be anything too special or unique about them. The UARTs do support interrupts, as any UART should, and the configuration of the registers for transmitting and receiving data match that of the NXP KL25Z128. The UARTs of the NXP LPC1125 feature 16-byte receive and transmit FIFOs, as well as RS-485/EIA-485 9-bit mode support with output enable and RTS/CTS flow control, among other modem control signals. The UARTs support both synchronous and asynchronous processing, as well as parity generation and checking.

For the SSPs of the NXP LPC1125, it is good to note that although the SSPs are different from the SPIs that are on the NXP KL25Z128, they have a similar functionality and are compatible with Motorola SPI, 4-wire TI SSI,

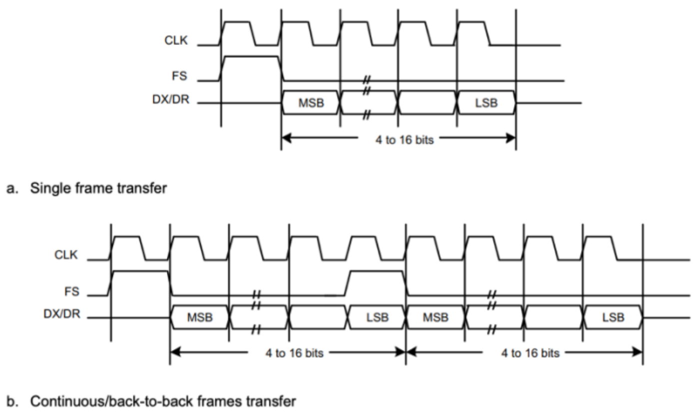

Figure 2 – Two examples of the Texas Instruments Synchronous Frame Format data transfer used in the SSI module of the NXP LPC1125 MC [1].

and National Semiconductor Microwire Buses, meaning that although the MC configuration for each will differ, the overall functionality remains somewhat consistent. The SSP peripheral, as with the SPI peripheral of the NXP KL25Z128, only supports synchronous serial communication, where its clock is set in its respective configuration register. Each of the two SSPs on the NXP LPC1125 can act as either a master or a slave. The SSPs are also equipped with eight-frame FIFOs for both transmitting and receiving data and have a specifiable frame size between 4 to 16 bits. Each SSP has 4 connection lines: its clock, slave select, Master Output/Slave Input, and Master Input/Slave Output. With these 4 connection lines, the SPI module of the NXP LPC1125 follows the 4-wire Texas Instruments Synchronous Serial Frame Format and an example of both a single frame transfer and a continuous/back-to-back frames transfer from the Texas Instruments Synchronous Frame Format can be found in figure 2; for more information regarding this format, see the NXP LPC1125 manual.

Transitioning to the standard I2C-compliant bus interfaces, they can be configured to Master, Slave, or Master/Slave. Data transfer is bidirectional between the masters and slaves, and the built-in programmable clock allows for the I2C transfer rates to be adjusted. The I2C also comes with serial clock synchronization, allowing different devices with different bit rates to communicate via the same bus, as well as suspending and resuming serial transfer, since this synchronization acts as a handshake mechanism. The I2C of the NXP LPC1125 supports an optional recognition of up to four distinct slave addresses. It also has a monitor mode that allows observing all I2C-bus traffic, regardless of slave address, which may be useful for debugging, as well as some other practical applications. What makes this I2C perhaps stand out is that it supports Fast-mode Plus, which is a module provided by NXP that allows rates above 400 kHz and up to 1 MHz to be selected as a clock rate. This mode is only available through pins 4 and 5 on port 0 of the NXP LPC1125. When the rates of Fast-mode Plus (400 kHz - 1 MHz) are compared to the rates of Fast-mode (400 kHz) or standard mode (100kHz), it is clear why having Fast-mode Plus can be extremely useful. The I2C is traditionally much slower than an SSP, but having Fast-mode Plus might alter some design decisions when selecting between which communication module to use for a given task. The I2C module also has a comparator built in that requests an interrupt when an equality is found, which would be another alternative to the NXP LPC1125 not having its own unique comparator. The built in I2C module is apparently very useful in interfacing to external I2C standard parts, such as serial RAMs, LCDs, tone generators, other MCs, etc.

B. Comparison of the NXP LPC1125 to the NXP KL25Z128 regarding Serial Communications

Comparing the two MCs in terms of serial communication, the NXP KL25Z128 has one low power UART and two standard UARTs; the NXP LPC1125 on the other hand has three standard UARTs. In terms of functionality of the respective UART modules for each MC, both are quite similar. They both share 16-byte receive and transmit FIFOs and the ability to set the baud rate, as well as hardware parity generation and checking. The main difference would be in the special case UARTS; if a low power UART is preferred, then the NXP KL25Z128 has an edge, however, if a UART with a modem control is desired, then the NXP LPC1125, is better.

Having a look at the NXP LPC1125's SSP compared to the NXP KL25Z128's SPI, the SSP being the newer technology might make it more future driven, however, the drawback of perhaps having to learn the new configuration might not be worth it if somebody is familiar with an SPI. Ultimately, they are extremely similar in terms of functionality though, and both would serve the same purpose rather well, hence, neither MC is given a real edge in terms of SSP for the NXP LPC1125 or SPI for the NXP KL25Z128.

This leaves the comparison between the I2C modules of the respective MCs. The NXP KL25Z128 comes with two I2C modules compared to the NXP LPC1125 single I2C module. Each I2C module of the NXP KL25Z128 is designed to operate at a speed of 100 kilobits per second, which is equivalent in speed to the NXP LPC1125 standard mode but is significantly less than the Fast-mode and Fast-mode Plus which operates at 400 kilobits and 1000 kilobits per second respectively. The NXP KL25Z128 I2C module can operate at frequencies up to 2.4MHz (clock/20), however, so this rate might in fact be faster; a further exploration into the performances at higher frequencies would be necessary to know which I2C is in fact better in that regard. Both MCs have some form of programmable glitch input filter, which can be useful in catching errors. With the functionality of each taken into account, the functionalities are rather similar, hence, the fact that the NXP KL25Z128 has two modules might give it the edge over the NXP LPC1125, which only has the one module.

To conclude the comparison of the two MCs, the NXP KL25Z128 also has a USB OTG controller with a built-in FS/LS transceiver; the NXP LPC1125 lacks this completely. Therefore, if this is a desirable communication module for a certain task, the NXP KL25Z128 would be the only realistic option.

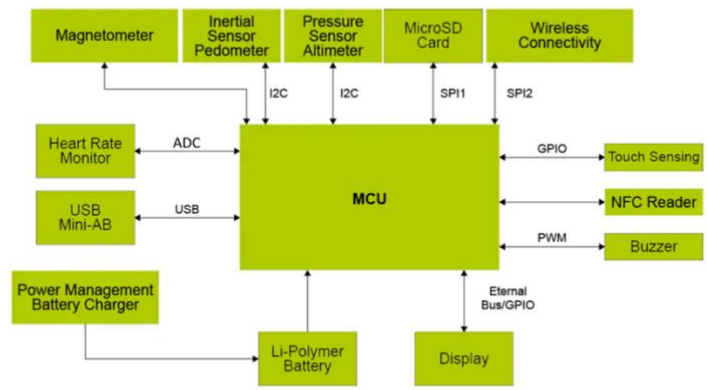

Figure 3 – The schematic for wearable health monitoring system of used for comparing the two MCs [3].

Application Analysis for a Wearable Health Monitoring System

A. Suitability of the NXP LPC1125 MC for the Target Application

From the beginning it is clear that the NXP LPC1125 would not be suitable for the target application without some sort of major design overhaul, and this is because it simply does not support the number or ports required for the assigned wearable health monitoring system. Given the schematic that can be found in figure 3, the monitoring system requires two I2C ports, 2 SPI ports, an ADC port, a pulse width modulator (PWM), a USB connection, and at least two GPIO connections. The NXP LPC1125 would without a doubt be sufficient regarding GPIO, PWM, and ADC. The SSP of the SPI should provide similar, if not identical functionality, and there are 2 of them on the NXP LPC1125, meaning it would have just enough for the monitoring system. There may be a learning curve in transitioning from one to the other, as they have different transmission logic, as well as different configurations. The bigger problems arise, however, with the I2C connections of the NXP LPC1125. As the NXP LPC1125 only has one I2C port, it would be impossible to implement the monitoring system without some sort of design overhaul. This design overall could be as simple as switching an I2C connection to another serial connection, or as complex as having the Inertial Sensor Pedometer and the Pressure Sensor Altimeter share an I2C port, which might be possible using the two different sequences of the NXP LPC1125 and toggling transmissions, or something of the sort. Both of these changes might hinder overall performance though, depending largely on the rate of data transmission required for each module, and this must be seriously considered. The NXP LPC1125 also lacks a USB connection, meaning that this is another clear design issue that would have to be addressed.

At the end of the day though, if the lack of an I2C port and USB connection is addressed, the NXP LPC1125 should be adequate in all other areas for the health monitoring system. Given the functionality of the NXP KL25Z128 that is being utilized, the NXP LPC1125 has extremely similar performance, and even has some attractive features such as its sleep and deep sleep mode features which might be useful for certain improvements to the monitoring system. The NXP KL25Z128 is noted for being an excellent low power MC as well, and since the monitoring system is battery powered, this could be especially attractive if in fact its low power features prove to be superior.

B. Comparison to NXP KL25Z128 for target application

This section will provide a conclusive comparison between the NXP KL25Z128 and the NXP LPC1125 regarding each's ability to be the MC behind a wearable health monitoring system. As stated in the previous section, the NXP LPC1125 lacks an I2C port and USB connection, hence, as it stands, it cannot be realistic option, however, we will assume that some work around is available and this section will compare the two MCs on other modules.

Starting with memory, the NXP KL25Z128 has more Flash and SRAM than the NXP LPC1125; the relevancy of this largely depends on how much of each memory the monitoring system requires. The NXP KL25Z128 has more GPIO pins than the NXP LPC1125 (66 to 38), but both are more than sufficient for the monitoring system and are extremely similar in functionality, so this will not really influence the decision of one MC over the other. With respect to each MCs Timer modules, the NXP KL25Z128 has a more diverse array of Timers to choose from, including a SysTick Timer, a 2-channel periodic interrupt timer, a low-power timer, a real time clock, and then three PWM timer modules, compared to the NXP LPC1125's SysTick Timer, two general purpose 16-bit counter/timers, and two general purpose 32-bit counter/timers. The only real edge that the NXP LPC1125 might have in this area is that it has 32-bit timers; the NXP KL25Z128 has only up to 16-bit timers. Both have more than enough Timers for what is required for the monitoring system, however, and many of the different Timer modules would be more than sufficient for operating a simple buzzer, which is the only Timer dependent peripheral of the health monitoring system. This leaves only the comparison between the analog and serial communication, which has been outlined above in this paper. To resume, the NXP KL25Z128 is undoubtably superior regarding serial communication diversity, unless an SSP is desired over an SPI for whatever reason, or modem control specific functionality is desired. For analog functions, the NXP KL25Z128 is also better, as it has more options in terms of ADC, DAC, and comparator options. The NXP LPC1125 not having a DAC might be particularly alarming if that is a desired module, but DACs as MC modules are not too commonly used anyways, and that's most likely why the NXP LPC1125 omits one; it is usually just as easy to have an external DAC.

Both MCs run the Thumb instruction set and are supplied by NXP, so there is likely to be much overlap in terms of programming and design, hence, the learning curve, or work required to transition from one to the other, should be rather stressless. The last important thing to acknowledge is that the NXP LPC1125 is a much cheaper MC than the NXP KL25Z128, and it seems appropriately so as it has much less peripherals. Since both entry level microcontrollers are rather cheap to begin with, however, this does not influence the overall decision of what MC to use for the monitoring system too much.

With all things considered, it appears that the NXP KL25Z128 is the better MC for the wearable health monitoring system. It provides much more flexibility in many modules compared to the NXP LPC1125 and includes all of the peripherals that would be required for the monitoring system without required a design overhaul. The only argument for the NXP LPC1125 would be that it is the cheaper MC of the two, which on a singular unit would make a small difference in price but could make a significant difference if a large number of units were sold. It can also be stated that the NXP KL25Z128 is in fact a bit excessive, as it has many Timers, pins, etc. that are unused in the current monitoring system, but I would argue that this allows for further expansion to the wearable health monitoring system at a minimal increase in price, solidifying the NXP KL25Z128 to be the better option over the NXP LPC1125.

References:

[1] Nxp.com. 2020. *Lpcxpresso1125 Board | NXP*. [online] Available at: <https://www.nxp.com/design/microcontrollers-developer-resources/lpc-microcontroller-utilities/lpcxpresso1125-board:OM13080> [Accessed 25 October 2020].

[2] Nxp.com. 2020. *FRDM-KL25Z|Freedom Development Platform|Kinetis® MCU | NXP*. [online] Available at: <https://www.nxp.com/design/development-boards/freedom-development-boards/mcu-boards/freedom-development-platform-for-kinetis-kl14-kl15-kl24-kl25-mcus:FRDM-KL25Z> [Accessed 25 October 2020].

[3] Mason, A., 2020. Research Project Guide. [online] D2l.msu.edu. Available at: <https://d2l.msu.edu/d2l/le/content/1201172/viewContent/9321593/View> [Accessed 15 December 2020].

** all information regarding the NXP LPC1125 were obtained from the first citation, and all information pertaining to the NXP KL25Z128 were obtained from the second citation. Both are citations for each MCs respective manuals obtained from the NXP website.