Embedded System Development for Autonomous Underwater Chemical Sensors

CSE-293 MS Project Report: Spring 2013.

Germán E. Alfaro
Department of Computer Science and Engineering
University of California, San Diego
La Jolla CA.
gealfaro@ucsd.edu

***Abstract*--In this paper I discuss the development of a new Embedded System to replace the aging system inside the SeapHOx autonomous chemical sensor system from Scripps Institution of Oceanography. This autonomous instrument measures important variables for the Marine Chemistry. The new design is both from Hardware and Software perspective. A new microcontroller is implemented based on ARM Cortex M4F (LM4F232H5QD), also the signal acquisition subsystem is improved with newer and better performance ADC.**

***Index Terms*—Embedded Systems, autonomous sensors.**

I. Introduction

We live in a time were constant change is the common denominator. The human being is changing the planet at a faster rate than ever before. It is of vital importance to be able to measure the environmental variables to understand the implications of this change. The progress on instrumentation and computation, particularly Embedded Systems, are allowing this measurements in situ for extended periods of time running autonomously.

Much of the ocean is sampled a few times [1] in periods as large as a decade because this measurements rely on ship that collect samples on ship or return to lab to analyze. Those sampling rates are inadequate to understand, some process that are in a more high frequency time scale like: daily, semi-diurnal or event episodic events like storms. This is called the sampling problem. (Figure 1)

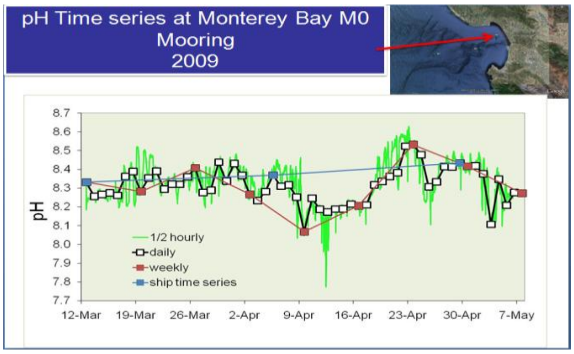

Fig. 1. Different frequency sampling (courtesy T. Martz)

In figure 2 we can see some of the routes of research oceanographic cruises that take year every year. As we can see is easy to notice, that not only is vast ocean surface not covered, but also the recurrence of those are not enough to measure high frequency phenomena, this can be overcome by using autonomous instruments.

Chemical sensors technology has matured enough in the last 20 years to the point of being able to get measurements of sufficient precision closely enough to lab precision, the resulting time series can be used to research the ocean chemistry. Autonomous chemical sensors in the ocean can be deployed: in moorings, profiling floats, or manually deployed in the button of shadow waters. [1]

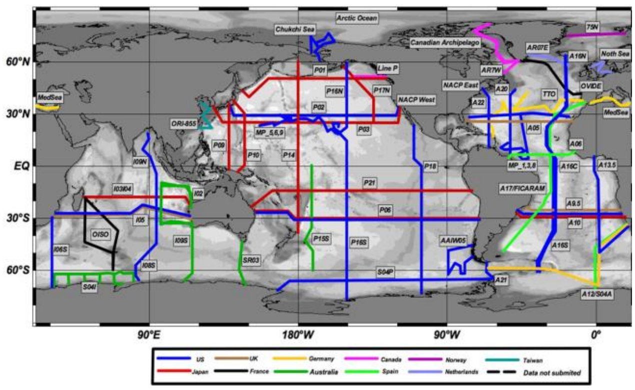

Fig. 2. Routes of Oceanographic cruises with frequency of one year.

The sensor system SeapHOx from Martz Lab at Scripps Institution of Oceanography (SIO) aim to acquire data of the ocean Inorganic carbon cycle, to understand the long-term variation in the mass of anthropogenic carbon dioxide in the ocean --the flux of CO_2 between sea-ocean, and the biological processes.[1]

In section II we are going to describe briefly the sensors that the SeapHOx system integrates, section III and IV the details of this M.S. project new Embedded System design will be addressed.

The complete characterization of the Inorganic Carbon cycle requires the measurement of two independent carbon dioxide parameters and understanding of the thermodynamic equilibria for acid/base of the solution. These parameters are: partial pressure of carbon dioxide in solution ($_pCO_2$), pH, total inorganic carbon (TCO_2), and titration alkalinity.

SeapHOx measures pH, pressure, salinity, temperature, in order to calculate the CO_2 content of sea water.

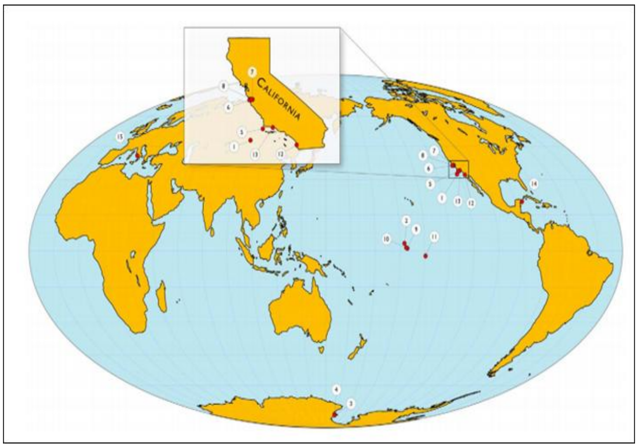

Fig. 3. Deployment locations of SeapHOx and earlier generation SeaFET. (More details at: Hofmann et al "High-Frequency dynamics of Ocean pH").

The SeapHOx has been deployed in very different ecosystems and locations around the earth from northern California to Antarctica, from Italy to Palmyra in the middle of Pacific passing from Yucatán, México. SeapHOx is deployed in moorings and also by just attaching it to the sea floor at shallow locations. Figure 4 shows the variability of Ecosystems the SeapHOx onboard pH sensor is capable to record. Currently SeapHOx is limited to shallow deployments less than 60 meters depth, because the pH sensor based on a ISFET chips fails at higher depth, there is a joint effort between: SIO, Monterey Bay Research Aquarium (MBARI) and Honeywell (the manufacturer of the pH sensor) to increase the depth operating range to more than thousand meters and being able to incorporate to profiling floats.

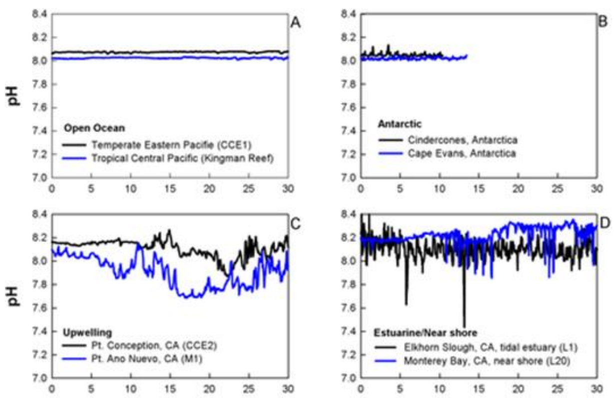

Fig. 4. pH measurements at different ecosystems by SeapHOx.[2]

The limitations of Autonomous underwater sensors without an order of importance are:

- Biofouling.
- Sensor drift.
- Communication.
- Power.
- Cost.

The Embedded System developed described in this paper tries to address the last three, having a significant success in the last one --cost around 20X improvement, power/performance ratio is incremented, and the communication peripherals are also increment in number and variety, these details will be further discussed in the design sections III-IV.

II. SENSORS

A. Honeywell Durafet pH sensor

The measuring technology behind the Honeywell Durafet is a Ion Sensitive Field Effect Transistor (ISFET). The ISFET is a Metal Oxide Semiconductor Field Effect Transistor (MOSFET) without a metal gate electrode over the conduction channel. [3] A conventional reference electrode is used instead the removed gate. ISFET's are commonly operated by applying a constant drain-source voltage and using a feedback circuit to hold drain-source current constant.

The analytical pH signal is proportional to the reference electrode to source voltage. The Durafet chip modification described at Martz[3] , Durafet seawater evaluation is the one used at the SeapHOx with a chloride ion selective electrode (CI-ISE) reference electrode.

The Durafet has been tested successfully on deployments of more than 6 months, with less drifting compared to a glass pH electrode. The Durafet sensor has better stability than 0.005 pH for periods of months of deployment at open ocean.

Fig. 5. Left Durafet attached to open SeapHOx, right a Durafet pH sensor from Honeywell. (source Martz Lab and Honeywell)

B MicroCat CTD

SeapHOx includes a Sea-Bird MicroCat 37S1 CTD to measure temperature and salinity. The MicroCat measures salinity by internal conductivity cell, temperature is measure by an aged thermistor [4] resistance to a high precision reference resistor to a 24-Bit ADC.

The MicroCat internal microcontroller communicates by a serial (UART) link to the main microcontroller onboard, the MicroCat is attached to the main waterproof case and connected to a waterproof cable connection.

Fig. 6. MicroCat attached to a SeapHOx.

C. Optode

The Optode is a optical sensor for measuring oxygen, according to the operating manual: [5] a fluorescence indicator embedded in a gas permeable foil that is exposed to the surrounding water, a black optical isolator protects the complex indicator from sunlight and water particles, the sensing foil is pushed against a sapphire window by a screw securing plate providing access to the optical LED blue light from the inside, the phase of a returned red light is measure compensated by temperature gives the absolute O_2- concentration.

The Optode is very stable in long term and communicates with main controller by serial (RS232).

Fig. 7. The Andeera Optode 3830, oxygen sensor (source Andeera)

D Pressure sensor MLH series by Honeywell.

The Honeywell MLH series sensor measures pressure by a metal diaphragm connected to a Application Specific Integrated Circuit (ASIC) [6] this sensor is connected to one of the ADC onboard. Having a pressure sensor onboard helps to calculate the depth of the deployment of the instrument and to calculate the characterization of the CO_2 on the seawater.

Fig. 8. MLH series sensor by Honeywell.

III EMBEDDED ELECTRONIC DESIGN

A. ADC

SeapHOx by the nature of the measurements it performs, needs to be able to have a design with attention to precision and avoiding noise, the original Analog-Digital Converter (ADC) CS5534 from Cirrus Logic, had 4 channels and 24-Bit resolution Sigma-Delta ADC, with a maximum sampling rate of 3.8 kSPS, --the maximum resolution drops proportional to the sampling rate. In our system design we got to the conclusion that was a good time to update the ADC onboard with newer and better performance chip, the ADC selected was the Texas Instruments (TI) ADS1248 [10], with the same resolution (24-Bit) but with more channels 7 in total (3 more) and also a faster sampling rate 20 kSPS this means almost 4X improvement in sampling rate, this allows to be able to average more samples, and also not only improving the resolution but also minimizing the time the electronics be powered and hence improvement in the overall power efficiency of the system.

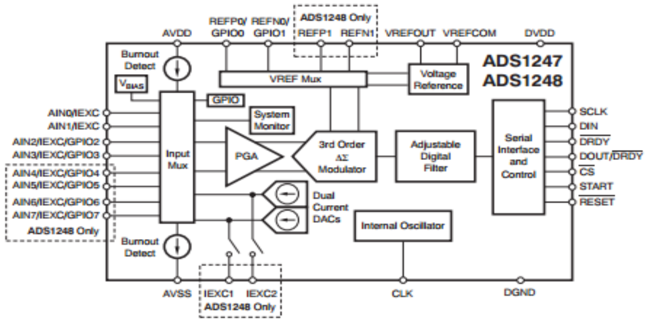

Fig. 9. ADS1248 Block diagram (source datasheet)

It is important to note that the new system has two ADS1248 onboard, for development we used an evaluation board from Texas Instruments: ADS1248EVM connected to the instruments and using SPI as a communication peripheral to the ARM Cortex M4F.

One of the greatest challenges of the whole project was to make the ADC circuit work with the Durafet pH sensor, since the signal from the Durafet is in the order of micro-Volts (uV), the pH sensor operates on negative voltage below 7 pH scale, that means we supplying the ADC with positive and negative voltage $^{+}$-2.5 V, the chip has an internal voltage reference. The constant value of the V int-ref is 2.048 V and also a Programmable Gain (PGA) from 1 to 128 on each channel, we use a gain of 1 since higher gain affects the resolution.

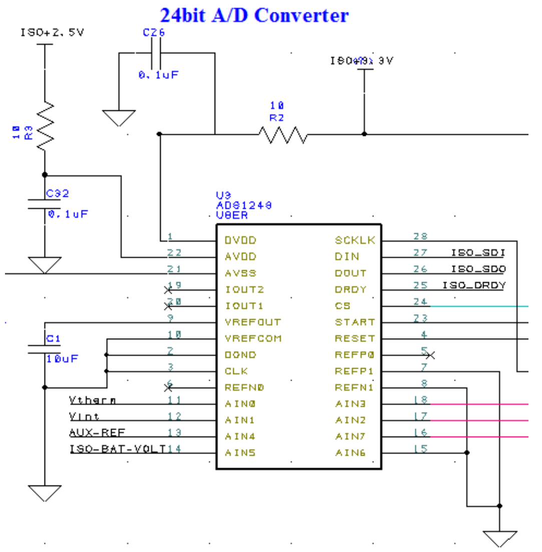

Fig. 10. ADS1248 Schematics.

The ADC is connected to the main controller by Serial Peripheral Interface (SPI), TI designates SSI it's version of SPI, but for practical reasons they are the same. For the SPI driver development was very helpful an internal temperature on chip that the ADS1248 has, since we were able to read the temperature and compare with the lab temperature, being able to test the voltage supply to the chip and the communication link.

In figure 10 we can see how the wires from the sensor are connected to the ADC to the Durafet since we have two ADS1248 we are going to call this ADS-A, ; Vtherm for the internal thermistor inside the Durafet, Vint for the voltage and ISO-BAT-VOLT, for the battery voltage.

The ADS-B connections are: ISO-BAT-VOLT (again) pressure from MLH sensor (Figure 8).

B. PCB and miscellaneous components.

Adding new components also represented adding new power supply chip, a new MicroSD card mounting fixture, another MAX3222 for another RS232 port, and also jumper connector for 4 on chip ARM pins with double function: GPIO and ADC of 12-Bits at 1 mSPS.

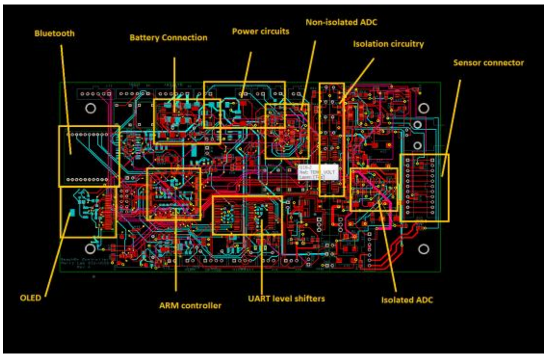

Fig. 11. New PCB

The main additions to the old design are: CAD parts of the described components and lay-out of ADS1248 and ARM Cortex LM4F232HQD, SD-card fixture, and tens of passive components. The software used for PCB design is: PCB Artist is a freeware from Advanced Circuits (Tempe Arizona). PCB Artist advantage is that you can summit the design from the same software directly to Advance Circuits manufacturing plant, the drawback is the lack of complete parts libraries, basically most of the time you end drawing each part, which is time consuming, but for prototypes the final PCB is good quality, fast turnaround and reasonable price.

TABLE 1. ADC comparison.

ADC	ADS1248 (New)	CS5534 (Old)
Resolution	24-Bit	24-Bit
Sampling rate	20 kSPS	2 kSPS
Channels	7	4

IV EMBEDDED COMPUTER DESIGN

A. Choosing a Microcontroller

The old SeapHOx included as a main controller a Persistor CF2, which is an embedded system well known to the oceanographic community, it uses two microcontrollers master-slave configuration, a Motorola MC68LK332 (32-bit) as a main controller, and TI MSP430 as slave. The MSP430 receives the name of Time Processing Unit (TPU) and it's in charge of turning of the system during hibernation, this embedded systems is mostly used as a data-logger for extended periods of time, having this configuration the CF2 is capable of being very power efficient. CF2 utilizes a proprietary operating system and development libraries that allows relatively easy programming, but the problem is not easy to "hack" new components since the system is predesigned, to utilize their own expansion cards for peripherals, the system is also constrained in the number of peripherals as is, without buying additional expansion units, this makes the platform already expensive ($395) even more expensive; price was the first motivation of replacing this embedded system from the SeapHOx, this price per unit is fair for a prototype or even a few number of parts, but for scaling to a price that researchers around the world may be able to use it, is when the situation change and worth the effort of investing development time in creating a newer economical system.

Table 2. Microcontroller comparison

Price	Price	Performance	Peripherals	Software support
LM4F232	Low	Excellent	Excellent	Excellent
PIC32	Low	Low	Good	Medium
LPC1700	Low	Excellent	Excellent	Good

Basically three microcontroller where considered and tested with evaluation boards, the first to be considered was the PIC32 since is an interesting chip powered by a MIPS core, it has also good variety of peripherals, but after testing the evaluation with some of sample code, it was easy to find that not many reference designs are out there, and also a faulty debugger/programmer we received make us realize that the support was not good from Microchip the company that manufactures the chip, other than that the chip runs at a good 80 Mhz, it seems to be powerful enough for our applications, but the MIPS technology even that is excellent lacks the actual presence of the ARM family of microcontrollers, that take us to our second device tested: the LPC1769 ARM Cortex M3 family from NXP semiconductor [7], this microcontroller is very high performance oriented with a clock rate of 120 Mhz (more than we need) and excellent range of peripherals even Ethernet included on chip, it is a very impressive device, the development IDE is based on Eclipse and came preloaded with tons of sample code, that actually runs at first time, definitively an excellent contender for our project. The third MCU evaluated was the LM4F232H5QD [9], it is a ARM Cortex M4F from TI, the "F" in M4F means floating point unit, this chip also has a DSP unit, that is basically the difference from a Cortex M3 to a M4, the actual core inside is the same, but they have great differences from one part to another, from one vendor to other in peripherals. Even that our project does not need to do signal processing computations it is good to have for free that capability since future sensors may require it. Other important aspect to mention is that all of the evaluated microcontrollers have a price under $10.00.

The selected part was LM4F232H5QD, not really because was the most higher performance since the SeapHOx requirements are about low power, but because was the one with higher number of serial ports (UART and SPI) and the best documentation from the 3 contenders, actually I would like to add that the online service of TI is excellent, they have an online blog where you can post questions and most of the time they are answered in matter of hours, and also other users are happy to help, in that sense was an excellent decision taken to choose the LM4F232H5QD.

Also the ARM family of microcontrollers have currently the most diverse family of MCU's, from the small Cortex M0 targeted at the 8-bit MCU market to other end of high end Multicore 64-Bit and the 95% of the world market of smarthphones [8].

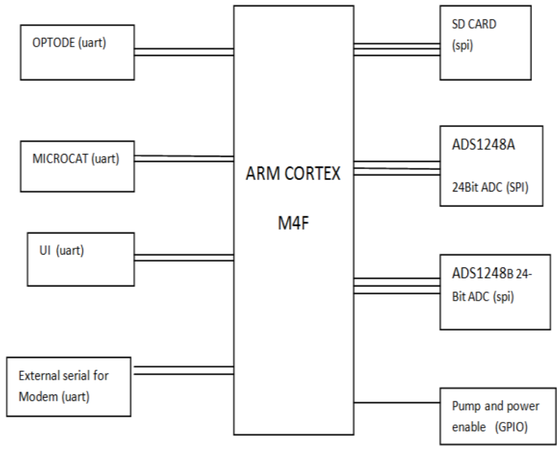

Fig. 11. Block diagram of new SeapHox Embedded System.

B. Software design

The first task after selecting the new microcontroller and also the new ADC (ADS1248) was to start developing the device drivers of the peripherals (Fig. 11). The development environment for LM4F232H5QD is the Code Composer Studio (CCS) from TI, which is based on Eclipse IDE, the compiler used is GCC for ARM, and the LM4F232HQD Evaluation board, which have a wide range of peripherals and soldering holes to connect to the GPIO's on the chip, among the peripherals has a SDcard holder/connector, this evaluation board has a programmer/debugger attached on the same development board.

The driver interfaces are modeled to resemble the ioctl from UNIX, the drivers developed are: SPI which is used for the ADC, the SDcard driver was provided by TI, but still we need to implement a method for writing to SDcard our main buffer that is fill with the data that came from polling the Microcat, Optode (both UART) and the two ADS1248, also a command line interface (CLI) to talk to the user in a friendly manner was developed, the CLI is used to retrieve the data stored after deployment, also has an option for testing each of the instruments on board, and also to store to EEPROM memory parameters that are used to the correct functioning on the board instruments, the user also can set the: hour, date and set the timer for sensing which in this application is normally 30 minutes. The system asks when setting up for the GMT time.

The instruments more used were: voltmeter, BusBee logic analyzer, and Oscilloscope from the hardware side. From the software side printing to console was the most useful resource, even more than using GDB on Eclipse. Having a logic analyzer at hand made much straightforward to develop a more advance protocol driver as SPI.

C SOFTWARE FUNCTIONS

Test: This function is made to be able to test-use the instrument from the command line interface, the user can select each of the instruments, then data is polled from the microcontroller where it runs the signal acquisition, the user is abstracted from the nature of signal acquisition in other words the communication layer is hidden.

This function is very helpful for sensor development and testing since the user doesn't need to rely in other signal acquisition systems.

Deployment: The deployment loop is the one that runs when the actual system is in the ocean, sensing, is basically a data logger, but the important point here is that it needs to be very power efficient, basically the system spends the time in deep hibernation. The M4F options to saving power are basically two: sleep and hibernation. The hibernation state also has another three functioning options: Real Time Clock (RTC) not enable, RTC enable and RTC enable with auxiliary battery, we need to be able to use the RTC clock since having the clock working is the only way to use a timer, in this state everything on the MCU is turned off, the only thing that remains is the actual clock. Our design uses two modes of waking up from hibernation: one is the clock timer (user input), and the other is a wake pin that is connected to a Reed switch thought a NPN transistor, this manual wake is used in field situations when the user needs to wake the instrument and work at the CLI.

The system basically goes to hibernation wakes at the predetermined time and starts a sensing routine, that basically polls the actual sensors one by one, and stores the data in a buffer, at the end of the routine this buffer is written to the SDCard. SDcard is not a limitation since most of the time series to date from the SeapHOx are under 1 MB.

To saving power some coding methods were used like pipelining instrument initialization and polling to save time and then power.

Table 3. LM4F232 specification and resources used.

	LM4F232HQ5D	Resources used
12C	6	0
CPU	ARM Cortex-M4	NA
Pin/Package	144LQFP	NA
Max Speed (Mhz)	80	48
Operating temperature	-40 to 85 °C	NA
Memory Protection Unit	Yes	NA
Watchdog timers	2	1
SSI (SPI)	4	3
ADC units	2	1
ADC resolution	12-bit	NA
UART	8	4
GPIO's	105	61(FREE)
SRAM (KB)	32	NA
Flash (KB)	256	NA

Configuration: This function as the name implies is used to configure the operation parameters of the system, this function ask for setting the time, and also there is operational parameters of the Durafet pH sensor used on the time of sampling that needs to be set on to the system, this parameters are stored on the onboard EEPROM memory.

V CONCLUSION

The PCB was received from the factory a couple of weeks ago, this is the first revision, fortunately the ARM was able to survive a short circuit made by wrong pin lay-out that was corrected and the ARM was able to boot and interact with the CLI from the PC. Also it is receiving data from: UARTS (Microcat and Optode) and also from SPI, failures had been corrected by directly wiring through the PCB, this is only for testing, we are working on having the issues corrected for the second week of June/2013 and send the second revision to manufacture.

It has been a great experience be able to work on a multidisciplinary environment in this project. Embedded Systems by nature can cross domains of knowledge this also add complexity to the projects but at the same time make the process more interesting and rewarding.

REFERENCES

[1] Kenneth S. Johnson, Joseph A. Needoba, Stephen C. Riser and William J. Showers "Chemical Sensor Networks for the Aquatic environment", Chem, Rev., 107, 623-640, 2007.

[2] Gretchen E. Hoffmann, et al, "High-Frequency Dynamics of Ocean pH: A multi-ecosystem comparison". PLoS ONE 6(12) e28983. doi:10.1371/journal.pone.0028983

[3] Todd R. Martz, James G. Connery and Kenneth S. Johnson, " Testing the Honeywell Durafet for pH applications", Limnology and Oceanography: Methods 8, 2010, 172-184.

[4] SBE 37-SI MicroCat Manual.

[5] Andeera 3830 Manual.

[6] Honeywell MLH series pressure sensor Datasheet.

[7] NXP LPC1769/68/67/66/65/64/63 Product data sheet.

[8] MICROCHIP PIC32MX5XX/6XX/7XX Product data sheet.

[9] Tiva™ TM4C123GH6PGE Microcontroller data sheet (identical to LM4F232H5QD TI changed names)

[10] Texas Instruments ADS1246-48 ADC data sheet

[11] Elicia White, "Making Embedded Systems" O'REILLY 2012.