BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

SYSTEM FOR AUTONOMOUS DATA COLLECTION FROM WEIGHTING SENSORS

SYSTÉM PRE AUTONÓMNÝ ZBER DÁT Z VÁH

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTOR PRÁCE

AUTHOR AUTOR PRÁCE	ADAM LUČANSKÝ
SUPERVISOR VEDOUCÍ PRÁCE	Ing. VÁCLAV ŠIMEK

BRNO 2016

Bachelor Project Specification/18973/2015/xlucan01

Brno University of Technology - Faculty of Information Technology

Department of Computer Systems　　　　　　　　　　Academic year 2015/2016

Bachelor Project Specification

For: **Lučanský Adam**
Branch of study: Information Technology
Title: **System for Autonomous Data Collection from Weighting Sensors**
Category: Embedded Systems

Instructions for project work:

1. Perform in-depth study of sensoric technologies suitable for weight acquisition of the objects under observation.
2. Propose a solution for monitoring weight of beehives and export of the collected data via selected network technology.
3. Design the architecture of a system for data collection and their visualization that allows to monitor the the weight of beehives.
4. Implement the proposed system and carefully verify its functionality.
5. Evaluate the achieved results and try to discuss possible extensions.

Basic references:

- According to the instructions of supervisor.

Requirements for the first semester:
Fulfillment of the items 1 to 3 of the assignment.

Detailed formal specifications can be found at http://www.fit.vutbr.cz/info/szz/

The Bachelor Thesis must define its purpose, describe a current state of the art, introduce the theoretical and technical background relevant to the problems solved, and specify what parts have been used from earlier projects or have been taken over from other sources.

Each student will hand-in printed as well as electronic versions of the technical report, an electronic version of the complete program documentation, program source files, and a functional hardware prototype sample if desired. The information in electronic form will be stored on a standard non-rewritable medium (CD-R, DVD-R, etc.) in formats common at the FIT. In order to allow regular handling, the medium will be securely attached to the printed report.

Supervisor: **Šimek Václav, Ing.**, DCSY FIT BUT
Beginning of work: November 1, 2015
Date of delivery: May 18, 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
612 66 Brno, Božetěchova 2

L.S.

Zdeněk Kotásek
Associate Professor and Head of Department

Abstract

Goal of this bachelor thesis is to design and create necessary electronics with plastic box enclosure and web-interface for low-cost beehive weight measurement. By measuring beehive weight in long term, health of colony can be determinted. In first part of the thesis, generic concepts of weight measuring, necessary electronics and means of wireless data transmission are presented. Later chapters are discussing proposed concept including firmware. Final chapter is presenting achieved results. First built unit will be given to the local hobby beekeeper.

Abstrakt

Cílem bakalářske práce je navrhnout a vytvořit potřebnou elektroniku s plastovým pouzdrem, webovým rozhranim pro nízkonákladové bezdrátové měření váhy včelích úlu. Sledovaním váhy úlu je možné dlohodobě sledovat zdraví včelstva. V první časti práce je vysvětlen obecný koncept měření váhy, potřebná elektronika a způsob bezdrátové komunikace. Následně je detailně popsán předložený návrh včetně firmwaru. V závěru se pojednáva o dosažených výsledcích práce a možnosti vylepšení. První vyrobena jednotka je darována lokálnimu včelaři.

Keywords

wireless, weighting, embedded systems, MSP432, BoosterPack, HX711, beehive, IoT

Klíčová slova

bezdrátove technogie, vážení, vestavěné systémy, MSP432, BoosterPack, HX711, včelí úly, IoT

Reference

LUČANSKÝ, Adam. *System for Autonomous Data Collection from Weighting Sensors.* Brno, 2016. Bachelor’s thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Šimek Václav.

System for Autonomous Data Collection from Weighting Sensors

Declaration

Hereby I declare that this bachelor's thesis was prepared as an original author's work under the supervision of Ing. Vaclav Šimek and Prof. Regine Bolter (during stay abroad). All the relevant information sources, which were used during preparation of this thesis, are properly cited and included in the list of references.

. . . .
Adam Lučanský
July 29, 2016

Acknowledgements

I would like to thank this way my supervisors Ing. Václav Šimek and Prof. Regine Bolter for help during writing of this thesis, as well as Prof. Reinhard Schneider and Prof. Robert Amann for making PCB prototyping possible during my Erasmus stay abroad.

Contents

List of Figures

Chapter 1

Introduction

For each every-day problem there comes an idea of automating otherwise tedious task done by human. This particular thesis is solving problem of regular collection of sensor data. Subject of measurement are beehives located in the garden.

Beehive weighting provides useful information, such as daily gain, health of colony.

Result of this project is a low-cost wireless solution for a hobby beekeeper, including weight measuring instrument with necessary electronics enclosed in a box. First developed device will be given to a local beekeeper for testing purpose.

In first two chapters, basics of project are described. Later, the proposed concept is introduced and all its parts are described in detail. Description of firmware is done in separate chapter, as well as cost analysis and testing.

Main motivation of this thesis is an experience gained with product design from idea to working unit.

Chapter 2

System overview

Overall system consists of three main parts (Fig. 2.1), which are further described in following sections including requirements. Whole system is intended to be as simple as possible.

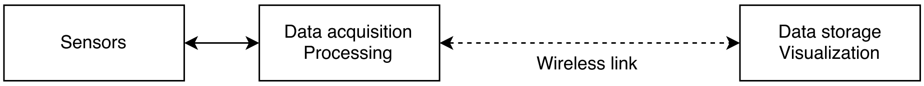


Fig. 2.1: General schema of system.

2.1 Sensors

Beehives are placed on chassis (Fig. 2.2) in between which is the load cell measuring weight (with analog output). Cable from load cell is fed to the central unit for further processing. Load cells and strain gauges are described in chapter 3.

2.2 Data acquisition

Load cell provides analog data, which needs to be converted to digital prior to further processing. Central unit contains MCU[1] which is sampling data out of weight sensors is reporting data on regular basis through wireless module to webserver for storage.

2.3 Wireless data transmission

For the purpose of logging data remotely, offsite data transfer is necessary. Data should be available at any time with no need to visit beehives physically and extract these data from storage media (SD Card). Several wireless technologies are available.

For ranges up to 50 meters to nearest network device (beekeeping in garden), such as router, WiFi provides easy connection without need of specialized component on receiving side. For long ranges, GSM/GPRS modem may be more appropriate, but requires better power management (due to 2A peak current[9, p. 2] of widely used GSM modem module SIM900).

[1] microcontroller unit

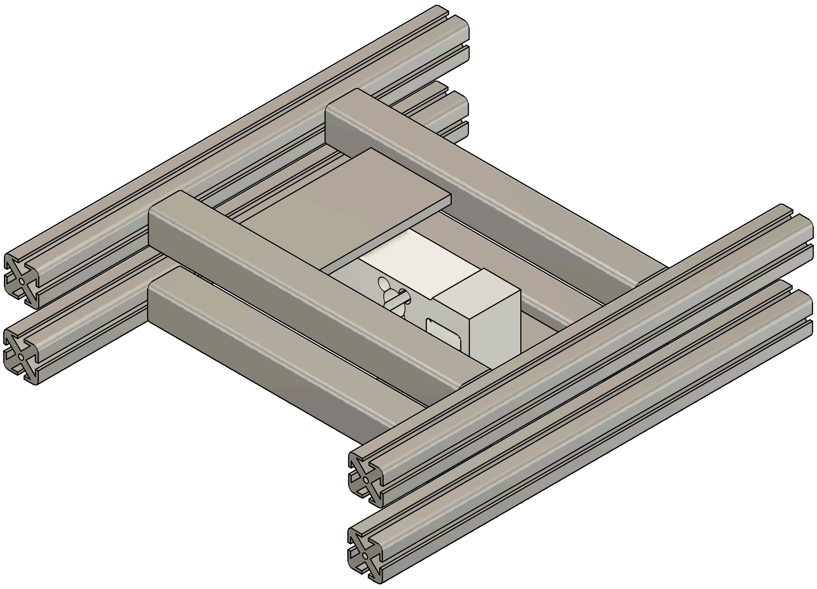


Fig. 2.2: 3D model of chassis with mounted load cell.

2.4 Data storage / visualization

Data acquired from load cells are sent through wireless network to a server, either as on LAN or as a VPS[2], handling data storage for further viewing. In case of webserver running on LAN, low-cost devices like Raspberry Pi can be used.

User accesses web service containing administration, charts for every load cell with history data and overall statistics.

[2] virtual private server

Chapter 3

Weighting principles

The most common strain measurement sensor is a strain gauge. The strain gauge transduction principle is based on the relationship between the change in length and its resulting change in the resistance of a conductor [6, p. 373].

A Strain gauge is a sensor whose resistance varies with applied force and in commonly used for load, weight, and force detection..It is basically a foil resistor, whose line resistance is proportional to the length and inversely to the area of the cross section. It consists of a small diameter wire, that is attached to a backing material (usually made of plastic). The wire is looped back and forth several times to create an effectively longer wire. The longer the wire, the larger the resistance, and the larger the change in resistance with. However, the change of the resistance is very small, so we need a good amplifier and measurement principle to detect such small differences. It is one of the most important tools of the electrical measurement technique applied to the measurement of mechanical quantities.

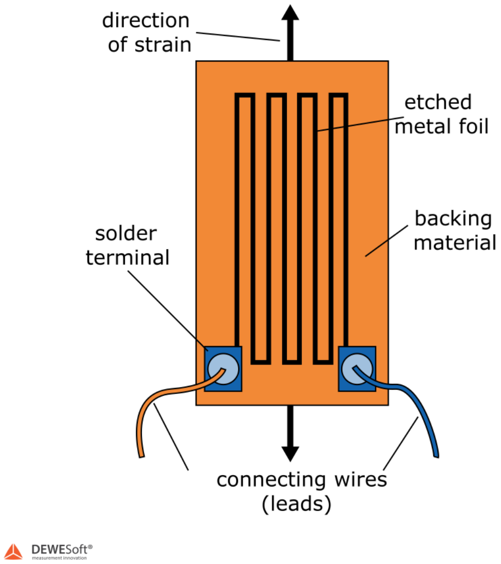


Fig. 3.1: Strain gauge structure [7].

The Wheatstone Bridge circuit is nothing else than two simple series-parallel arrangements of resistors connected between a voltage supply terminal and the ground producing zero voltage difference, when the two parallel resistor legs are balanced. It has two input terminals and two output terminals consisting of four resistors configured in a diamond. You can see the typical drawing of Wheatstone bridge in the picture below. It is suitable for measuring small changes in resistance, making it good for strain gauges. [7]

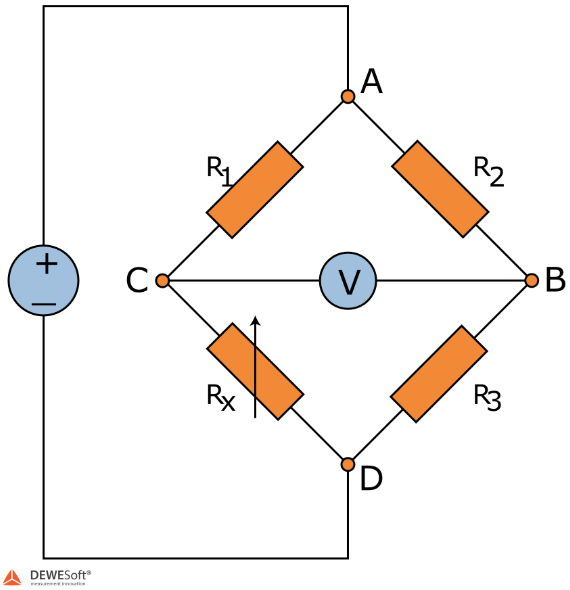


Fig. 3.2: Wheatstone bridge [7].

Typical force/torque sensors are called load cells. [6, p. 373].

Beehives are to be placed on a chassis with necessary measuring instrument (load cell in this case), see Fig. 3.3. Each beehive stack weights from 30kg to 90kg – depending on the time of year, therefore chassis and load cell shall sustain weights up to 150kg.

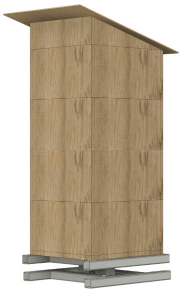


Fig. 3.3: Beehive placed on a weighting chassis with load cell.

Chapter 4

Existing solutions

This chapter summarizes similar available commercial systems similar to the unit created as a result of this thesis. Provided functionality may vary.

4.1 ALYA s.r.o

Company ALYA s.r.o. offers product with name VILKO (Fig. 4.1), which contains electronics necessary for measuring beehive weight. According to manufacturer, weight is placed only under reference beehive which is located near the rest. Device is transmitting data through GSM network. Whole set includes weight and costs 660€[1, p. 1].

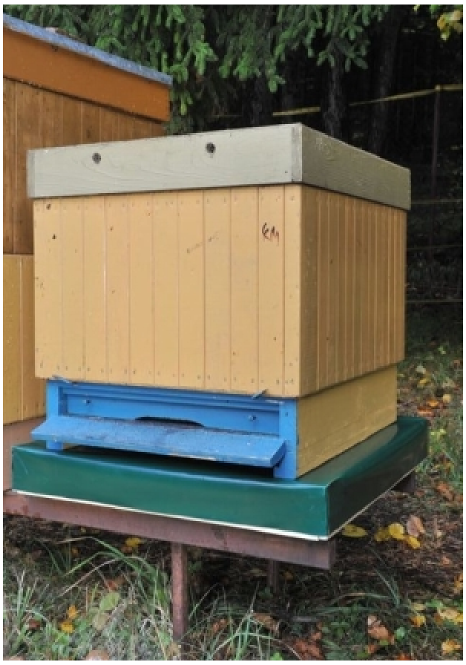


Fig. 4.1: Beehive placed on a weighting system of company ALYA s.r.o. [2].

4.2 Beewise

Company Beewise® offers similar system. This system is using reporting by SMS messages and can be also powered by solar panel. Interesting feature is availability of theft protection – stolen unit cannot be used because PIN code is hardcoded in device[5].

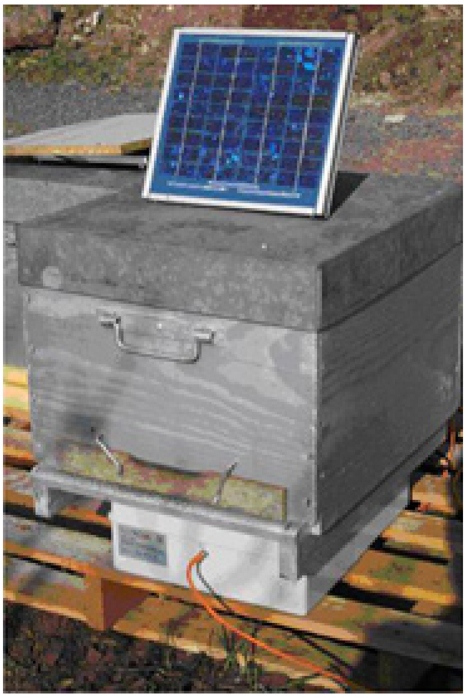


Fig. 4.2: Beehive placed on a weighting system Beewise [5].

4.3 Arnia

Company Arnia offers wireless weighting system, where under each beehive, separate wireless module is transmitting data to central module, which aggregates data and sends them through 2G GPRS network. Units are battery powered without solar panels. Data are sent to shared cloud service [3].

Chapter 5

Proposed concept

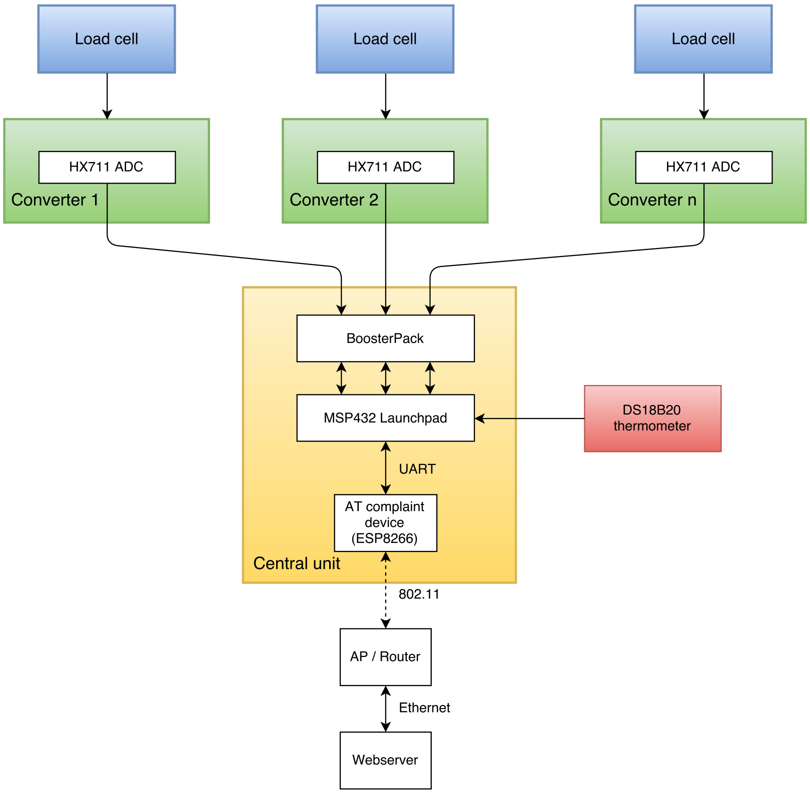


Fig. 5.1: Detailed schema of system

Central unit enclosed in 3D printed box (Fig. 5.3) contains Texas Instruments MSP432 LaunchPad [12], battery pack as a power source, WiFi module and manufactured BoosterPack. BoosterPack is an expansion board with connectors placed at predefined position on a PCB fitting together with master module (see Fig. 5.2). For each load cell, ADC converter with amplifier on breakout board is provided in separate box (Fig. 5.1).

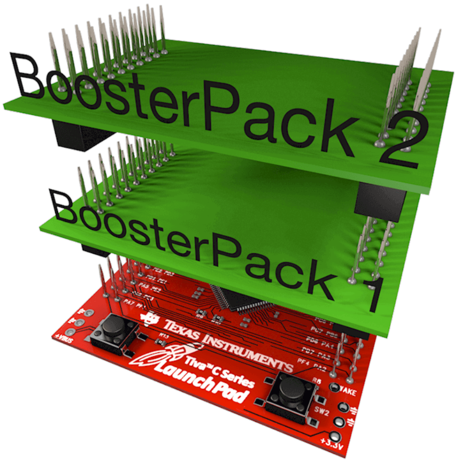


Fig. 5.2: BoosterPack's ability to stack [12].

First development iteration contained converters on BoosterPack, but due to size requirements, separate modules for converters were created.

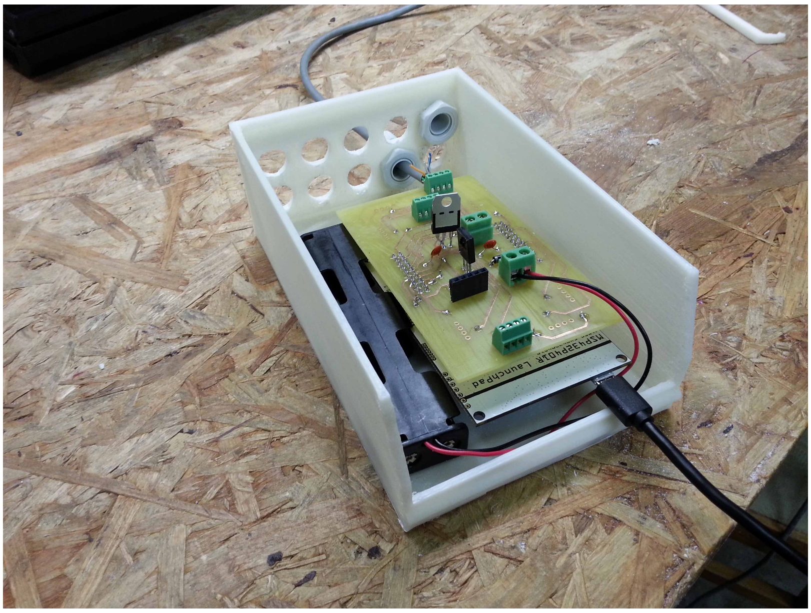


Fig. 5.3: Central unit in box, without lid

5.1 BoosterPack

Chosen platform TI MSP432 enables developers creation of customized add-on boards, similar to shields for Arduino. One such BoosterPack is designed and manufactured to enable easy connection of converter modules and batteries with central unit (Fig. 5.3).

BoosterPack includes connectors for 8 weighting converters, WiFi module, battery, solar panel (reverse protected by diode), connector for temperature sensor DS18B20[1]. Sleep mode on WiFi module can be enabled on-demand through PNP transistor.

[1] https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

5.1.1 PCB

Design of BoosterPack started with defintion of necessary peripherials, mostly connectors for weighting sensors. PCB is designed in tool Upverter[2] (Fig. 5.4), which provides easy user interface for prototyping boards.

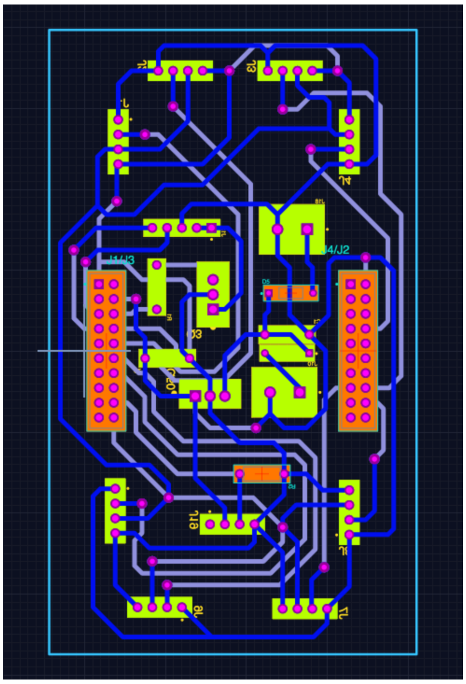


Fig. 5.4: PCB layout of BoosterPack. Designed in utility upverter.com.

5.1.2 Power

BoosterPack is powered directly from 3.3V rail provided by LaunchPad. Device shall be able to be powered by external energy (solar panel, battery), therefore has to have suitable linear regulator. Low quiescent linear regulator LF33CV is used, which has maximum input voltage of 40V [11]. During autonomous measuring, system is primarily powered by 4x AA batteries. Solar panel is connected for reduction of power consumption during the day.

[2] https://upverter.com

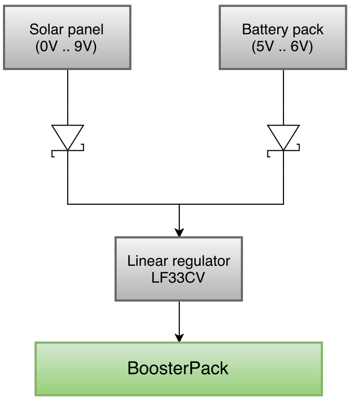


Fig. 5.5: Block schema of power management.

5.1.3 Connectors

4 pin screw terminals with pitch 2.54mm are used between central unit and converter – in future, may be replaced with RJ45. 2 pin terminals with pitch 5.08mm are used for connecting external power. Header connector is used for connecting WiFi module.

5.1.4 Manufacturing

PCB prototype was then manufactured on CNC milling machine. After functional validation, more robust PCB (Fig. 5.9) was made by etching.

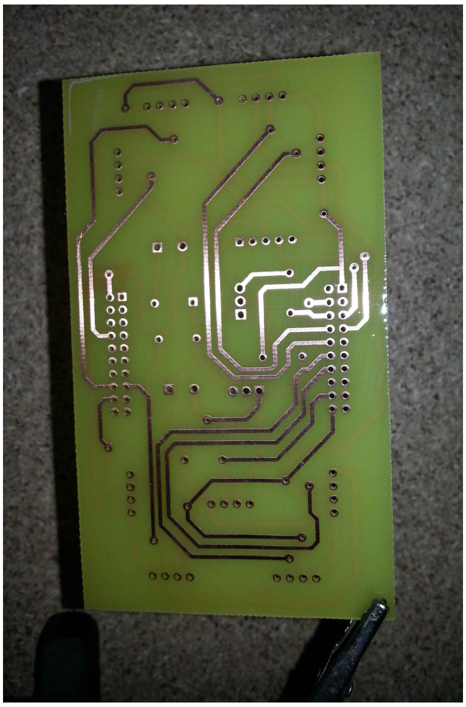


Fig. 5.6: Manufactured BoosterPack PCB

5.2 Load cell

Consulting with local distributor of weights and weighting systems LESAK s.r.o., load cell ZEMIC L7G (Fig. 5.7) was suggested by them to be suitable for this purpose. This particular load cell has maximum capacity of 300kg. Cost of one unit is 101€(as of May 2016)[3], although price may be more favorable in greater quantities.

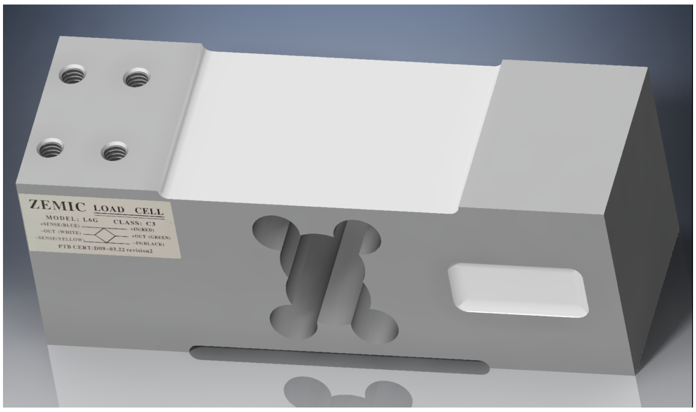


Fig. 5.7: 3D model of load cell ZEMIC L6G

5.3 Weight converter

Weight of object placed on chassis with load cell is acquired by integrated circuit **HX711** which contains amplifier (gain 128x) and 24-bit ADC[4] .

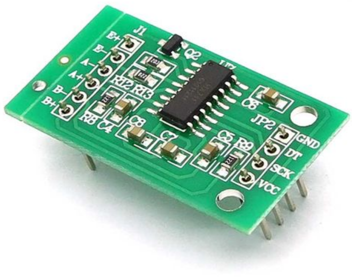


Fig. 5.8: HX711 on breakout board, source: [10]

Integrated circuit is soldered on an inexpensive breakout board for easy prototyping.

Whole system needs to modular up to 8 beehives.

HX711 on breakout board is soldered to designed PCB5.9 making it possible to connect external thermometer. In current design, data from external thermometers are not gathered.

[3] http://www.profivahy.cz/profi-vahy/eshop/0/3/5/772-ZEMIC-L6G-300kg-IP-65-hlinik

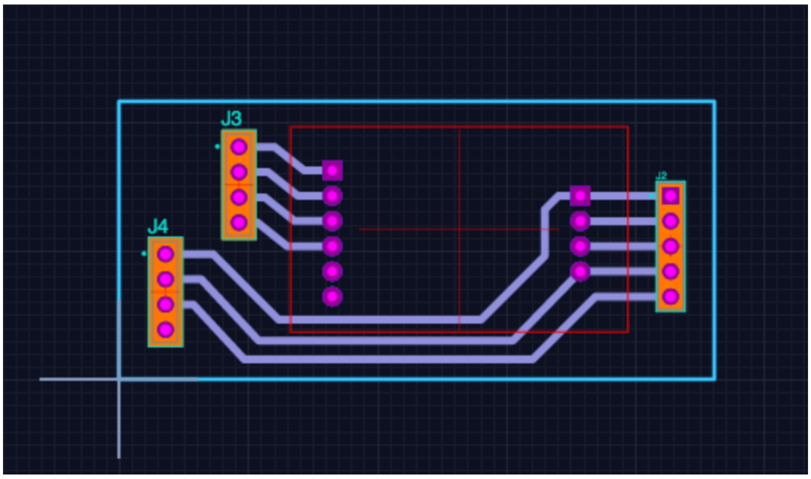


Fig. 5.9: PCB design of BoosterPack.

Breakout board connected to PCB is enclosed in 3D printed box (5.10) for protection against short circuiting.

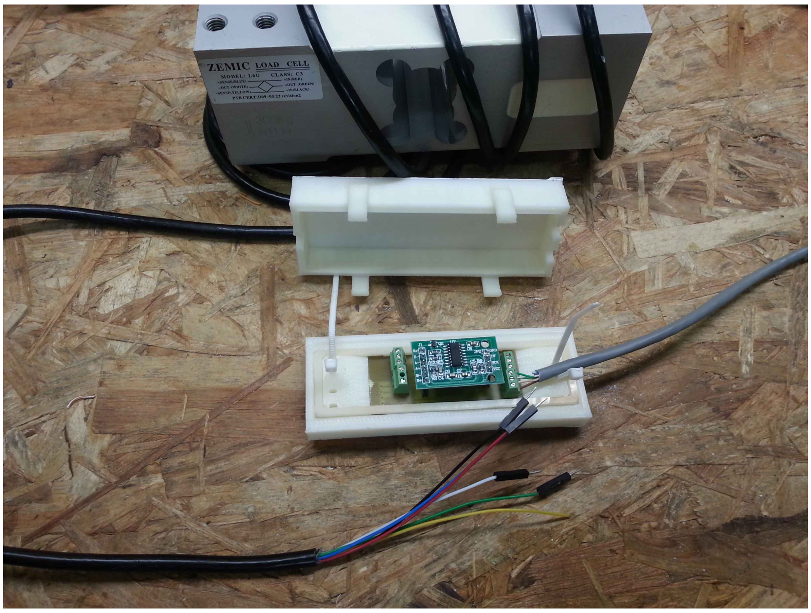


Fig. 5.10: Converter mounted on PCB with 3D printed box.

5.3.1 Protocol

ADC converter HX711 uses simple bit-banding protocol described in [4]. Protocol is similar to SPI – by sending pulses to `PD_SCK` pin, one bit of 24 bit number is shifted out to `DOUT`. In one reading cycle, up to 27 pulses may be sent to `PD_SCK` – first 24 bit are for data, three more are for setting gain of amplifier in next measuring.

5.4 Enclosure

Due to outdoor placement of box with electronics, sufficient cover protecting PCB inside needs to be created. Simple solution for small amount of manufactured units is 3D printing such box. Such box can be seen in Fig. 5.11.

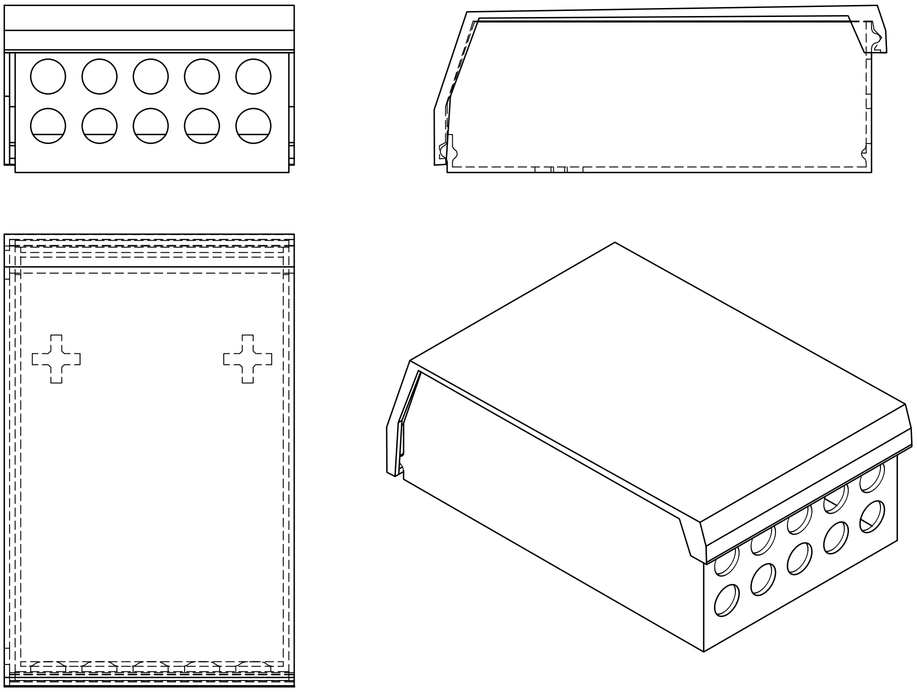


Fig. 5.11: Central unit assembly model.

3D printing of bottom part takes up to 4 hours.

5.5 Data transmission

Central unit contains WiFi module ESP8266, which acts as client. Module contains compiled firmware OpenMCU with Lua language interpreter[4]. This module is interfaced through UART from MSP432 and receives commands. Module also may run in both client and AP mode, which allows overall configuration through web browser. For the purpose of this thesis, SSID and password is predefined in a file.

5.6 Information system, service

Information system has four database tables (Fig. 5.12). System also differentiates sensor types – currently only weight.

[4] https://www.lua.org/start.html

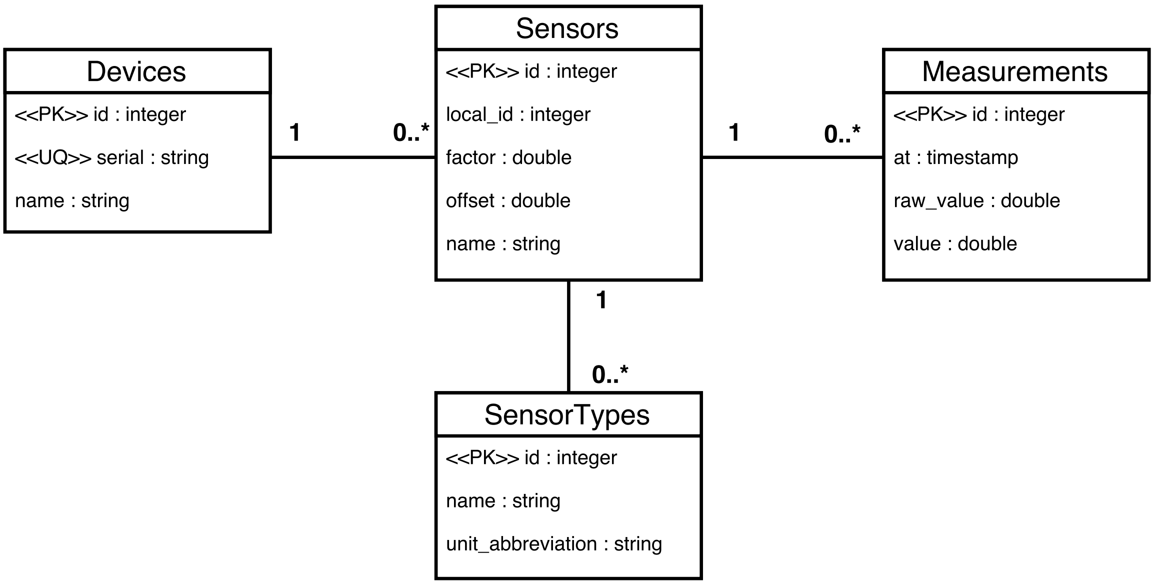


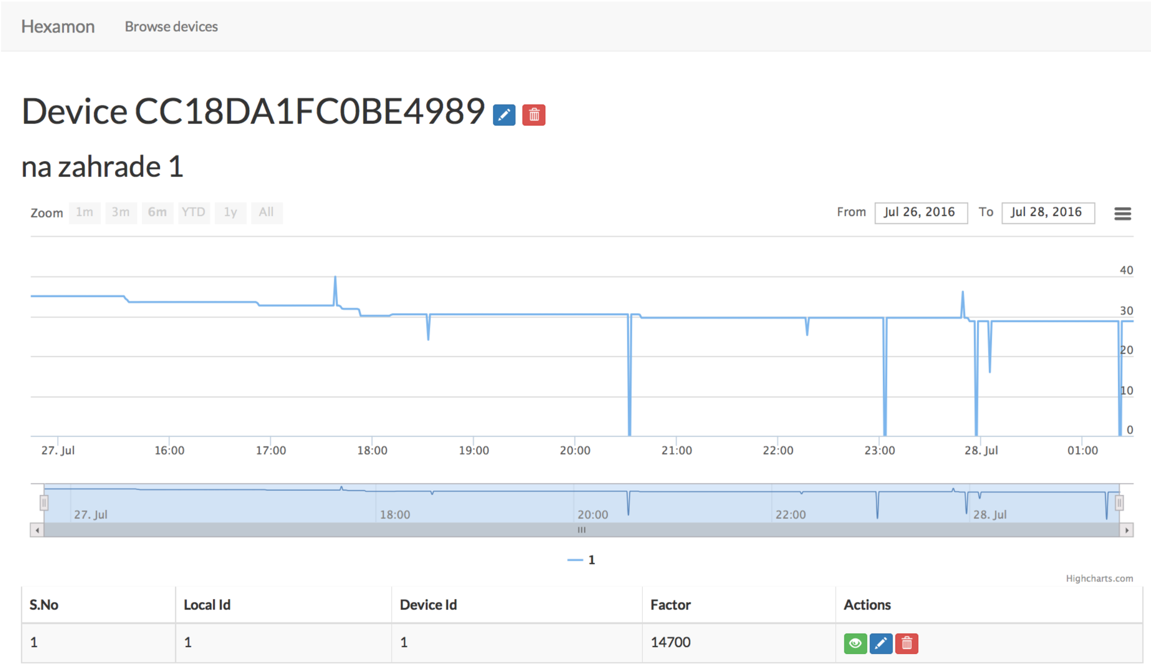
Fig. 5.12: ER diagram of information system.

HTTP requests sent from ESP8266 are received by a virtual server running LAMP[5] stack. Information system is running on PHP (Laravel framework) interfacing MySQL database.

Each central unit has 64 bit serial number stored in ESP8266, which is used for authentication when submitting data in form of hexadecimal representation.

User needs to know serial number except for last 3 bytes for access to statistics. In order to submit new data, full serial number must be known. This method prevents database poisoning. Method is not bullet-proof, but serves purpose well and prevents necessity of more complex authentication methods like username and password.

[5] Linux, Apache, MySQL, and PHP



Device CC18DA1FC0BE4989

na zahrade 1

S.No	Local Id	Device Id	Factor	Actions
1	1	1	14700	👁 ✎ 🗑

Fig. 5.13: Web interface with device detail, showing charts. For debugging purpose, full serial number is shown.

Example charts can be seen on URL: http://hexamon.io/devices/CC18DA1FC0 .

5.6.1 API

Data are reported in HTTP GET request parameters. System is made to be as simple as possible, therefore central unit needs to send only one type of API call:

```
GET /devices/{SERIAL}/sensors/{LOCAL_ID}/{SENSOR_TYPE}/{RAW_VALUE}
```

where:

- SERIAL – 64 bit serial number in hexadecimal format, uppercase.
- LOCAL_ID – integer which identifier sensor in scope of central unit (usually 0 or more).
- SENSOR_TYPE – denoted type of sensor. Possible values are weight, temp, humidity.
- RAW_VALUE – raw value acquired by sensor. Tare and multiplying is done on server-side.

Chapter 6

Modules description and firmware

This chapter presents details of proposed modules in previous chapter and describes used firmware.

6.1 ESP8266

Espressif ESP8266 is an inexpensive WiFi module containing 80MHz/160MHz SoC chip with architecture Xtensa LX106[8]. Module can be bought as a PCB soldered by user to breakout board (Fig. 6.1).

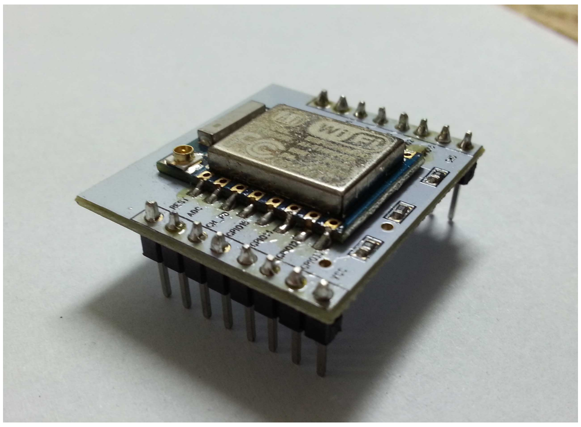


Fig. 6.1: ESP8266 module soldered on breakout board.

Main advantage of this module is that its possible to flash to it customized firmware. For this project, NodeMCU was chosen due to low complexity. Firmware contains filesystem, which makes it possible to store and edit scripts without full reflash. Last but not least benefit may be ability to redownload scripts periodically from webserver, making it possible to upgrade API remotely.

6.2 MSP432

MSP432 is an architecture introduced in 2015 by Texas Instruments, based on previous architecture MSP430, providing backwards compatibility [13]. As of Q2 2016, standalone

MCU is available only on a LaunchPad or as a sample. For this particular project, programming language Processing was chosen for easier development.

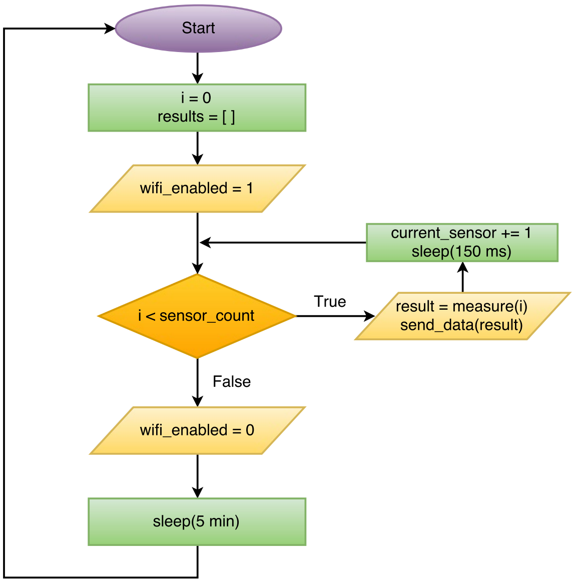


Fig. 6.2: Flowchart of program in MSP432. Parallelogram denotes interfacing with I/O.

Program running on MCU consist of main loop (6.2), which is regularly collecting data from sensors and sending them to webserver through WiFi module. Between each measurements cycle through all sensors, MCU sleeps for some time – usually 5 minutes or 1 hour, depending on need.

Chapter 7

Calibration and testing

This chapter is briefly describing calibration process and determining error of continuous measuring of permanent load.

7.1 Determining scaling factor

Calibration of load cell is done by placing object of known weight on load cell. By measuring raw value from converter without any load and with reference weight, parameters of linear function describing curve mapping raw value to weight can be estimated.

$$offset = raw\_value\_no\_load$$
$$scaling\_factor = \frac{raw\_value\_loaded - raw\_value\_no\_load}{object\_weight}$$

7.2 Error of continuous measurements

By placing constant weight on load, 800 measurement samples were done with period of 5 seconds to see how weight changes in time. For calibration and testing like this, wooden platform (7.1) was created on which items can be placed. After 800 samples with 5 second period, conclusion was made that standard deviation is 4.99 grams under constant temperature. Full list of measured values can be found on attached DVD.

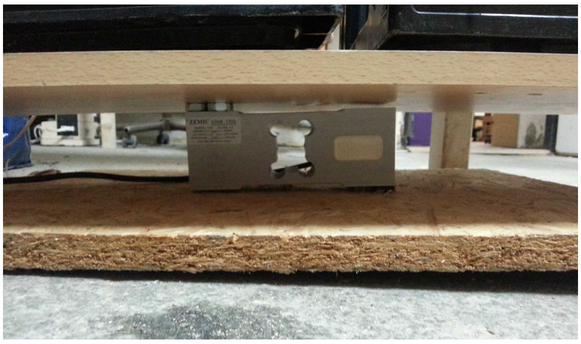


Fig. 7.1: Reference platform consisting of two wood sheets for calibration

Further improvements can be made by shielding the converter. For this particular purpose of weighting, given error is sufficient.

Chapter 8

Conclusion

First prototype unit was successfully made and ready for release for long term testing. Developed system fullifies requirements, but is still not mature enough to compete with commercial solutions. Development of this project will continue after completion of this thesis. Vision is to integrate MCU on one board with all connectors, with possibility to chain converters on one bus, LiPo battery charger circuit which will make it possible to charge batteries from solar panel and GSM modem for remote data gathering.

First unit will be installed on 8 beehives of a local beekeeper for long-term testing and collection of feedback of the device. BoosterPack is probably not the best solution for commercial applications, but may provide alternative to expensive commercial products. Interest to such systems can be seed throughout online forums and websites like Instructables [10], containing simple methods to achieve beehive weighting. Current state of the project is by no means user-friendly in terms of installation and usage, but provides a good starting point of development of such devices. Webservice is current running on domain http://hexamon.io.

Bibliography

[1] ALYA, s.r.o.: Pricelist of beehive weights. http://www.alya.sk/alya/userfiles/documents/vcely/Cennik_vcelarskych_vah_2-2015.pdf. 2015, [cite. 2016-05-12].

[2] ALYA, s.r.o.: VILKO - multifunkčné vážiace zariadenie (MVZ) pre včelárov. http://www.alya.sk/vcelarska-vaha-vilko. 2015, [cite. 2016-05-12].

[3] Arnia: Arnia – remote hive monitoring, System Overview. http://www.arnia.co.uk/how-it-works/. 2011, [cite. 2016-05-13].

[4] AVIA Semiconductor: 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales. https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf. 2009, [cite. 2016-05-10].

[5] Beewise: BeeWISE, product information. http://www.beewise.eu/bee-hive/gms-sms-scale.php. 2005, [cite. 2016-04-13].

[6] Cetinkunt, S.: *Mechatronics With Experiments, 2nd Edition*. Wiley. 2015. iSBN 9781118802465.

[7] Dewesoft: Strain measurement. http://www.dewesoft.com/pro/course/strain-measurement-1. 2016 [cite. 2016-01-27].

[8] Espressif Systems IOT Team: ESP8266EX Datasheet. http://download.arduino.org/products/UNOWIFI/0A-ESP8266-Datasheet-EN-v4.3.pdf. 2015, [cite. 2016-04-03].

[9] Shanghai SIMCom Wireless Solutions Ltd.: SIM900 Reference Design Guide. http://simcom.ee/documents/SIM900/AN_SIM900%20Reference%20Design%20Guide_V1.02.pdf. 2010, [cite. 2016-05-10].

[10] SohamG: Make your weighing scale hack using arduino and hx711. http://www.instructables.com/id/Make-your-weighing-scale-hack-using-arduino/?ALLSTEPS. 2015, [cite. 2016-01-27].

[11] STMicroelectronics: Datasheet - Very low drop voltage regulator with inhibit function. www.st.com/resource/en/datasheet/cd00000546.pdf. 2015, [cite. 2016-05-10].

[12] Texas Instruments Incorporated.: MSP-EXP432P401R. https://www.ti.com/ww/en/launchpad/launchpads-msp430-msp-exp432p401r.html. 2015, [cite. 2016-05-10].

[13] Texas Instruments, Incorporated.: MSP432P401R – Low-Power and High Performance 32-Bit ARM Cortex M4F With 256KB Flash and 64KB RAM. http://www.ti.com/product/msp432p401r. 2015, [cite. 2016-05-12].

Appendices

List of Appendices

Appendix A

Contents of DVD

`src/esp/esp-open-sdk` - ESP8266 toolchain
`src/esp/nodemcu-firmware` - NodeMCU firmware with integrated Lua language interpreter
`src/esp/esptool` - Utility for flashing firmware
`src/esp/luatool` - Utility for transferring scripts to ESP8266
`src/msp432` - firmware of MSP432 with libraries
`src/web` - Laravel website containing information system
`BP_xlucan01.pdf` - this thesis in digital form

Appendix B

Compiling and flashing firmware

This chapter discusses firmware compilation and flashing procedure.

B.1 ESP8266

B.1.1 ESP8266 toolchain

In order to compile NodeMCU, toolchain needs to be built to proceed. Depending on used platform, installation manual[1] needs to be followed. DVD contains precompiled OpenSDK for x64 Fedora flavour Linux distribution.

B.1.2 NodeMCU

Before compiling NodeMCU firmware, file `BuildShell` needs to modified to suit the path of resulting binary generated in previous section. Default directory is `/share`. By issuing command `make`, firmware is built. Resulting firmware is located in directory `bin` with names `0x00000.bin` and `0x10000.bin`.

B.1.3 Flashing NodeMCU

Upon connecting ESP8266 board to any USB-to-serial converter (must be 3,3V complaint) and connecting `GPIO0` to ground pin. By issuing command `./esptool.py -b 921600 write_flash -fm dio 0x00000 bin/0x00000.bin 0x10000 bin/0x10000.bin` firmware is flashed. By pulling GPIO0 to 3.3V and restarting the device, NodeMCU is started. By reconnecting to serial port at baud rate 115200, terminal is shown with Lua interpreter prompt.

B.1.4 Uploading scripts

Serial port cannot be occupied during script uploading due to collisions. In file weighting.lua, default SSID, password, IP address of webserver can be changed. By issuing command `make`, necessary scripts are copied to file system. By reissuing make command, all existing files are rewritten. Automated updating of scripts from webserver upon startup is not implemented in current version.

[1] https://github.com/esp8266/esp8266-wiki/wiki/Toolchain

B.2 MSP432

To compile and upload firmware to MSP432, program Energia[2] needs to be downloaded and installed. After installation, project can to be opened (weighting.ino). Before flashing through the program, board needs to be set in **Tools - Board - LaunchPad w/ msp432 EMT (48MHz)**

B.3 Web installation

Directory `/public` needs to be set as a root for given VirtualHost. To set database config, file `.env.example` needs to be copied to file `.env` and modified appropriately. Package manager Composer[3] needs to be installed. After installation, commands `composer install`, `php artisan migrate` need to be issued.

[2] http://energia.nu/

[3] https://getcomposer.org/

Appendix C

BoosterPack and converter PCB

Both BoosterPack and converter were designed in online service Upverter. Previous outdated KiCad designs are not published.

BoosterPack schematic with design:
https://upverter.com/Adluc/0fc215ac3ab9a2c0/40-Pin-Launchpad-BoosterPack-HX711/

ADC converter PCB:
https://upverter.com/Adluc/47b601ea2e495fe4/HX711-ADC-board/