ETH Zürich
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computer Engineering and Networks Laboratory (TIK)
Distributed Computing Group (DISCO)
Prof. Dr. R. Wattenhofer

Semester Thesis
Spring Semester 2010

Design Of An Energy Harvesting Network Node for Wind Measurement

Author:
Oliver Schultes

Date of delivery: July/21/2010

Supervisor: Johannes Schneider
Philipp Sommer

Abstract

Sensor networks are many small devices connected to each other. Ideally any of such a device is a autonomous system with its own power supply. One possible task of such a network is a grid of sensor nodes for wind strength measuring. If a node is able to generate power out of wind, a sensor node's power supply can be supported. And increasing its life time heavily.

The topic of this report is to develop a node which is able to harvest the power from the value under observation. In this case wind. A system for a sensor node has been designed and partially evaluated. The basics of wind energy harvesting have been worked out. And a set of requirements have been defined for most of the components of the system. But the focus is the energy harvesting part and the evaluation of the single components.

The processor module including the processor and a radio transceiver has been evaluated for power consumption and timing behavior running tinyOS operating system. The energy storage of the node has been investigated under certain conditions. The energy harvesting consisting of a generator and a power manger integrated circuit has been measured under various conditions.

A proposal for a prototype is made and ready for rebuild.

Contents

1 Chapter 1. Introduction

1.1. Motivation

Weather-observation is one of the oldest activities of mankind, but it is still an important field of science and research. Today's predictions are mostly based on mathematical models. The input data for those models consist of many measuring points all over the world. For example, the standard mesh of the MeteoSwiss is about 141 measuring stations for the whole country of about 41'285 km^2 [1, 2]. This is good for giving large scale predictions but not enough to explain air flow within an area of some square kilometers. To observe such small areas and record the fluid dynamics it is necessary to increase the density of sensor stations heavily.

Why is this of interest? A case in Wasserauen (Switzerland) from 1984 which repeated in 2007 showed dramatically the local influence of wind turbulences in a valley. Wind gusts overthrew a whole train from its rail Fig. 1.1. Up to date it is not possible to simulate the air streams which are responsible for such heavy wind gusts in that area. One problem is that only two measurement stations exist, one on top of the mountain and one in the valley. Therefore it is of interest to have a closer mesh of points especially for gust measuring. Such a network of tiny sensors could be helpful to get input data for further calculations and modeling.

A further application for such a sensor-network is on airports or on cable cable-cars. Airplanes are often confronted with difficult wind situations during the touch down. Crosswinds are known since the beginning of flying and are well treated as long as the wind stream is continuous. The airplane can fly a small angle into the wind. This compensates for the wind trying to push the airplane sideways. In that maneuver the real problem are gusts. If the pilot were able to know when such a heavy gust will hit the plan he could fly a touch-an-go.

Figure 1.1.: Foto: Kantonspolizei Appenzell Innerrhoden.

Another scenario are cable-cars. In ski resorts they are often mounted in exposed positions for wind gusts. It happened more than once that the cable skipped out of the wheels due to wind gusts. With a warning network the car could be stopped before the gust arrives. A stationary cable seems more resistant against gusts.

A network which can deliver such information consists of many nodes and a base station. The nodes are ideally fully autonomous systems related to the power. A node should be able to harvest its own power. Therefore the nodes have a longer lifetime compared to a exclusively battery driven system. The captured information is transfered via a radio link to the base station. There the information from many nodes are collected and interpreted.

Having set a few boundaries for the desired devices a view to the market is made. What kind of sensors are currently available to the market? There are a lot of tools to measure wind velocity. But not every device is suitable for this project. One of the goals of the project is to build autonomous devices. For a larger number of nodes it is meaningful to have autonomous device which are independent of a power source. In an ideal case once a node is placed nobody has to care about it up to that date the network is being removed. Therefore all methods which are in principle not able to harvest energy are not suitable. The different available systems will be explained later.

So, what makes wind energy better than solar energy? The task is to measure wind therefore it is obviously a challenge to receive the power for the device from that parameter which is under observation. A solar-cell is just a additional part which must be included into the tooling and so should be avoided. Contrary to other forms of energy harvesting is wind one of the uncommon method for micro power applications. Because the smaller the windmill the smaller the effectiveness due to increasing losses.

1.2. Theoretical Background

1.2.1. Wind

Beside the special behavior at the noted valley, wind is strongly dependent on its environment. In this particular case it is of interest to get information about the wind gusts. Therefore a little research about gusts was made. According to Hau [3] gusts are usually treated as one dimensional fluctuations of wind speed, otherwise the calculations are too complicated. Those events are a few to some tens of seconds in duration. It turned out that a standard meteorological measuring is averaged over 3 seconds. Times about 3,5, and 15 seconds are used by wind-loading engineers to describe wind loading events [4]. Thus it is not possible to recognize events shorter than that time with standard meteorological tools.

The gust factor G is introduced to describe such events. It is defined as the ratio of the maximum wind speed to the mean wind speed. The values for G vary between 1 and infinity theoretically. But practically between 1 and 3. This means that the maximum speed is three times as strong as the underlying average wind. As explained by Hau [3] the gust factor is a function of the duration and also depends on the frequency of occurrence.

The diagram in Fig. 1.2 shows the gust-factor as a function of the mean wind speed. For the case described in the previous chapter there must have been mean wind speeds of above 10m/s to overthrow a train. For example, looking at the second curve, the gust factor remains below a factor of two. Another diagram Fig. 1.3 shows the gust factor versus gust duration. Factors of less than two can appear in time spans of 100 Milli seconds. Therefore it should be possible to recognize events with less than a second duration. It must be pointed out, that the diagrams used in this chapter are valid for the North German coastal area and not for mountain terrain, but in

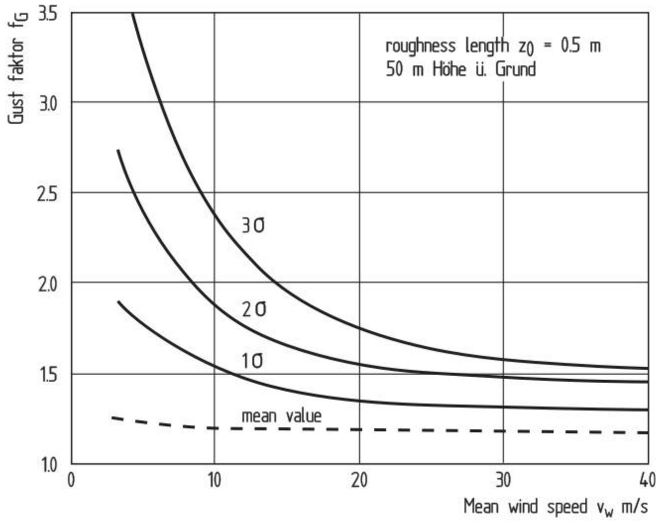


Figure 1.2.: Mean wind speed vs. Gust factor [3].

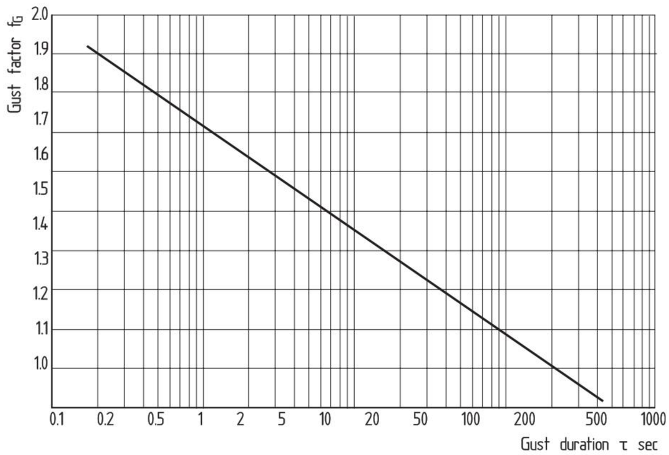


Figure 1.3.: Gust duration vs. Gust factor [3].

absence of real data it is a good assumption to get an impression of the expected values. Different sources state that some conditions hold for wind gusts:

- A wind gust does not appear out of a average wind of zero.
- Expected effects appear at speed higher than 10m/s.
- The wind direction is of interest too.

Data provided by the German-Weather-Service (DWD) show the maximum wind speed per day. Data is available for every day of a year [5]. The year 2009 is selected for reference. The speed is given in terms of the Beaufort scale. This means a degree of 5 is approximately 10m/s. The data shows a duration of 164 days of less or equal to 10m/s. The biggest difference between two days with less than 10m/s is found as 16 days.

Tool	Gauging principles				Application	
	mech.	sonic	thermal	others	aver.	turb.
Cup anemometer (Schalenstern-)	x				x	
Vane anemometer (Propeller-)	x				x	(x)
Hot wire anemometer (Hitzdraht-)			x			x
Ultra sonic anemometer (Ultraschall-)		x			x	x
Laser anemometer				x	(x)	X

Table 1.1.: Deployment scenario for wind sensors [3].

1.2.2. Measurement

Wind measurement is done in many different ways. The basic instruments of anemometry are listed in Tab. 1.1 which was published in [3]. It shows the different measurement methods. The oldest is the mechanical method, based on the drag of the wind stream.

A very common one is the Cup and Vane anemometer its mode of operation is based on counting the number of rotations per time interval. Cup anemometers have a well known and widely discussed issue which is called over-speeding [6, 7, 8]. Its response to rising gradients is faster than to falling thus showing a slightly higher average wind speed than there really is. This is the main reason why such anemometers can't be used for turbulence measurement.

The Hot wire anemometer is a heating element which is cooled by the bypassing wind. The amount of thermal energy loss can be measured as a variation in the resistance respectively the current and is so far proportional to the wind speed. A advantage is its good response to turbulences and its sensitivity for low speed streams. A disadvantage is its calibrating procedure due to the dependence of the environment temperature [9].

Ultra sonic anemometer and laser anemometer have good properties in turbulence measuring and sampling rate [10]. One utilizes the fact that sonic waves are influenced by wind and thus delaying or pushing the acoustic velocity [11]. The other uses particles in the air to excite a Doppler-shift. But both of them need a more complex electronic system to evaluate the results. Additionally, ultra sonic anemometry is dependent on temperature and humidity.

Every of the given methods has its advantage for different applications. But there are only two applicable for the application under discussion. Cup and Vane anemometers. Both anemometers available on the market work with the same principle. The electronics inside simply has to count pulses per unit time to calculate the speed. This is done either with inductive sensing or optical sensing. The shaft can run more or less friction free. But in order not to disturb the measurement it is necessary to reduce friction of the shaft as much as possible.

Now using the anemometer to harvest energy the rotational power must be converted to electrical power. An alternator must be attached to the shaft. Attaching one has two effects. First it is an additional mass. This induces a moment of inertia and thus giving the system a larger low pass behavior. And second, the torque against the rotation is heavily influenced by the electrical load attached to the alternator. For a good measurement the load at the alternator has to be constant to make the influence to the uncertainty in the speed of the shaft predictable. But this point will be worked out later. Summing up, it is not possible to use a existing system which is available on the market.

1.2.3. Energy Harvesting

The principles on wind energy harvesting are based on a single equation which gives the relation between the wind velocity and the force excited by the wind. The maximum power available in an free air stream is given by:

$$P_0 = \frac{1}{2}\varrho A v^3 \tag{1.1}$$

In general one important parameter for turbines is its efficiency. The ability to convert the available power of a fluid-stream to rotational power. Unfortunately all turbines are limited in efficiency.

The Betz limit for turbines is the value of maximum efficiency. For a horizontal-axis the Betz limit for turbine efficiency is 59% [3] but even good turbines work below this threshold.

There are some methods to convert the wind stream to mechanical energy. One method is the drag type, used in cup anemometers or some vertical axis windmills like Fig. 1.4. Such a device uses aerodynamic drag instead of aerodynamic lift, like a airplane does. A drag device has a reasonable large surface in the air stream. To this area a mechanical drag is excited by the stream. The air impinges to the surface A and generates a force to the area:

$$F = \frac{1}{2}\varrho A c_w v^2 \tag{1.2}$$

This force is a power at the shaft according to:

$$P = M\omega = F v_r = \frac{1}{2}\varrho A c_w (v_w - v_r)^2 v_r \tag{1.3}$$

It turned out that two speeds are involved. v_w the wind velocity and v_r the rotational velocity of the shaft. The velocity generating the torque is the difference of the wind and rotational velocity. Thus the power delivered by the shaft is determined by the mass of the liquid (ϱ), the area (A), the drag coefficient (c_w) and two velocities (v_w, v_r). The drag coefficient only depends on the geometries of the cup. For a hemisphere c_w can reach a value up to 1.33.

The Betz limit for a ideal vertical axis turbine is roughly 30% [12] for a drag coefficient of 2.0 [13]. This is lower than for a horizontal axis turbine. The ratio between maximum power and available power is the variable c_p. It depends on the ratio of the Wind speed and rotational speed:

$$c_p = \frac{P}{P_0} = c_w \frac{v_r}{v_w^3}(v_w - v_r)^2 = c_w \left(\left(\frac{v_r}{v_w}\right) - 2\left(\frac{v_r}{v_w}\right)^2 + \left(\frac{v_r}{v_w}\right)^3 \right) \tag{1.4}$$

Fig. 1.5 shows the velocity dependent part of the equation, without the c_w factor. From the graph it can be seen that the ratio of $\frac{v_r}{v_w}$ between 0 and 100% , reaches its maximum at relative wind speed of 33%. Multiplied by the drag coefficient it is the ratio c_p. This is a harvesting ratio not an efficiency ratio. In numbers for the cup anemometer it is $1.33 \cdot 0.15 = 0.2$. Thus the harvested power ratio is about 20% maximum.

But it must be pointed out that this calculation is for frictionless ideal application only. It is assumed that the real power coefficient is much less than 20% because of the friction of the shaft, moment of inertia and the generator load itself the voltage regulator etc.

A example calculation shows the expected amount of mechanical energy from a small drag device. Roughly speaking this is the power at the shaft. A anemometer shovel of $5cm$ diameter is assumed. The density of air is $1.2kg/m^3$ according to a reference chart [14].

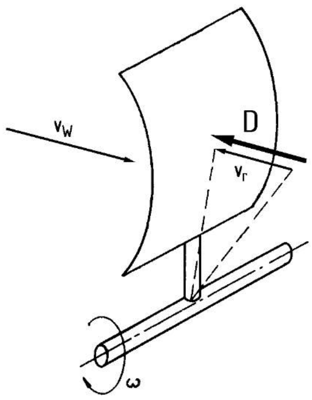


Figure 1.4.: Drag device.

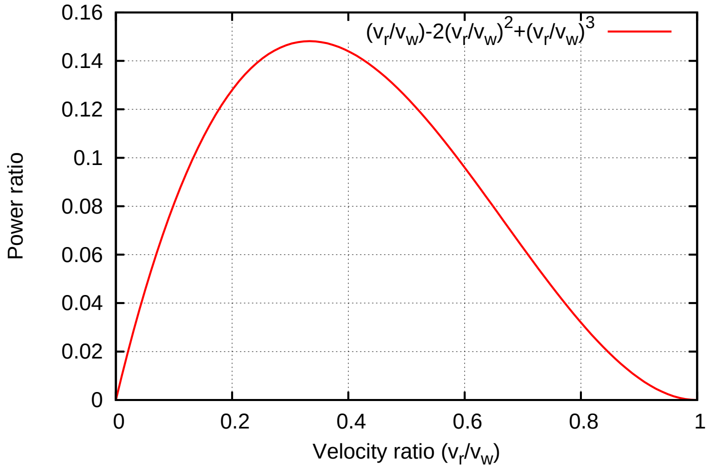


Figure 1.5.: Power-Ratio to determine the effenciency of a turbine.

Assuming a wind speed of $5m/s$ which is equal to a Level of 3 at the Beaufort scale.

$$P = P_0 \cdot c_p \cdot \eta_a = \frac{1}{2}\varrho A v^3 \cdot c_p \cdot \eta_a = \frac{1}{2}1.2kg/m^3\pi(2.5cm)^2(5m/s)^3 \cdot 0.2 = 29mW \tag{1.5}$$

Compared to a Level of 5 equal to 8m/s:

$$P = \frac{1}{2}1.2kg/m^3\pi(2.5cm)^2(8m/s)^3 \cdot 0.2 = 120mW \tag{1.6}$$

But it must be explicitly said that this is a harvesting ratio, this does not include any efficiencies of the system.

1.3. Related Work

Scientists had a lot of ideas about energy harvesting for many different applications. The goal is always to design truly autonomous systems without external connections for power supplies. This ranges from micro-watts for watches, harvested by the heat of the body to some few milliwatts produced by a oscillating weight and micro-generators. Inductive or piezoelectric elements in shoes for on-body applications to a few hundreds of milliwatts. A famous method of harvesting is solar energy. It covers a huge range from a few milliwatts to some hundred watts or even kilowatts in a power-plant. For wind energy this is the same range as well. It is just a question of size and efficiency. Only two scientific papers have been found about energy harvesting with windmills.

A paper by the University of Singapore from 2007 [15] describes to some degree exactly what is relevant for designing such a autonomous wind speed sensor application. A system has been developed, consisting of a wind turbine generator, a three phase rectifier, a voltage converter, power storage and the wireless transmitter. The rectifier is necessary because of the output of the turbine delivers AC power signal. The turbine, a vane, has 6 blades each with a radius of 17cm which is not very small but allows a reasonable handling. The system continuously sends data via the transceiver and is designed to measure the wind speed over long periods of time. Due to the relatively large blades of the turbine and its cut-in speed of 2.2 m/s it is possible to harvest energy and measure velocity down to speed of about 3 m/s.

Another interesting topic mentioned in the paper is that they use a self-made feed-forward buck converter, motivated by the statement that the efficiency of DC-DC converter IC chips for light load is rather low.

A second interesting paper by the university of Colorado [16] gives an example how to utilize a cup anemometer with a compact self-made alternator. Their goal is to provide a system for continuous wind speed measurement. It is supplied by a battery and increases its lifetime by additional power harvesting.

This paper shows the first realistic scenario with real values. Their alternator connected to a load resistance of $1k\Omega$ is producing power between $20\mu W$ and $1.02mW$. According to the paper the output voltages are approximately between 0.1V and 0.3V including a rectifier for the AC alternator. Obviously based primarily on the wind speed.

An interesting topic is the influence due to the attached alternator to the measured wind speed. They state that the alternator induces an error of approx 10% and is roughly linear. As a final result they present a graph where the Output Power vs. Wind Speed is given. The output includes the alternator, the rectifier and the regulator. It is given in Fig. 1.6.

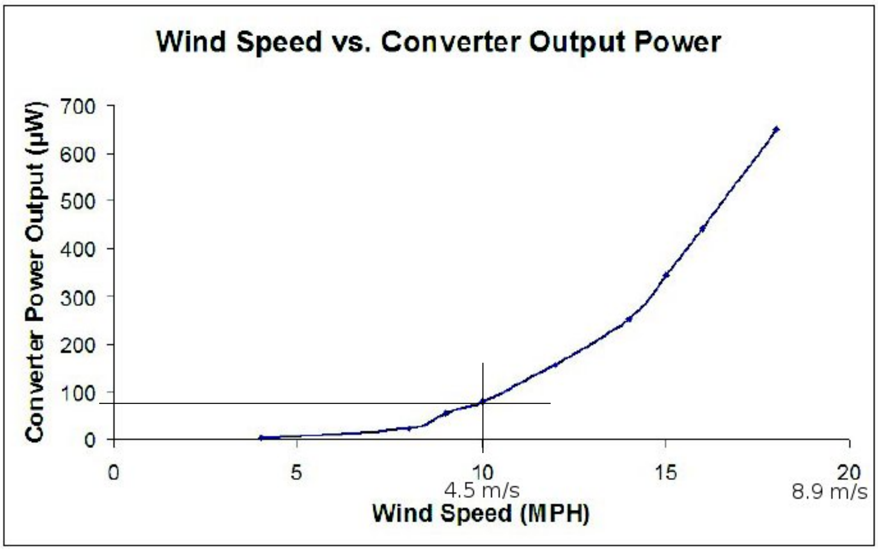


Figure 1.6.: Buck-Boost-Converter Output Power vs. Wind Speed (in [16]).

2 Chapter 2. Component evaluation

The system must consist of a generator, a rectifier, a voltage converter (regulator), a buffer capacitance and the Processor and RF module. The quantity to be measured is either the frequency and/or the Voltage of the generator. This is decided during the evaluation process. In this chapter the properties of different components are evaluated and the best components are selected for the prototype.

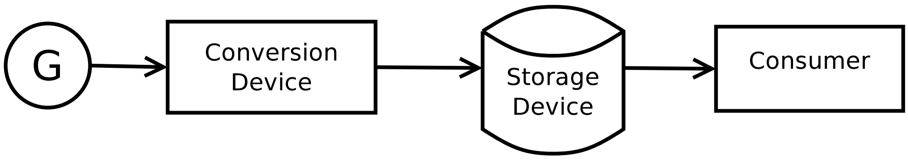


Figure 2.1.: Blockdiagram of components.

2.1. ZigBit Modul

Starting with the processor and the RF module because their power consumption is responsible for the dimensioning of the preceding elements. The focus is on Atmel's ZigBit module. The ZigBit module represents the consumer in the chain. It is a small sized fixed system consisting of a microprocessor, crystal, and RF-chip inside a shielded case. It is able to run with a minimum of external components like a antenna, a balun and a few blocking capacitors. According to the data sheet it should be possible to drive the module to a low-power mode with a current consumption of $6\mu A$. To test this feature, the module was used on the Pixie platform. The components mounted on the platform are only those described as the minimum configuration. Therefore this module is applicable for low-power testing.

By inspection of the default configuration used inside the programming environment (Eclipse with YETI / avr-dude) the configuration can be found. It turned out that the AVR has the following configuration: The key features are:

- System running on $8MHz$ internal RC Oscillator.
- RTC Crystal mounted ($32,768kHz$).
- RTC has Prescaler of 32 and generates Overflow Interrupts with $4Hz$.
- Radio uses its own crystal with $16MHz$.

The fuse Bits are: lfuse reads as C2, hfuse reads as 99, efuse reads as FF. This is a configuration as following: The efuse Byte is configured as default thus according to the data sheet. Important to notice, the Brown-Out-Detector is disabled. This configuration is given in Tab. 2.1 in detail. For further description of the fuse bits see the data sheet [17] of the micro-controller.

Register	Bit	Value	Description	Register	Bit	Value	Description
lfuse	7	1	Divide clock by 8 (disabled)	hfuse	7	1	Enable OCD (disabled)
	6	0	Clock output (disabled)		6	0	Enable JTAG (enabled)
	5	1	Select start-up time (maximum)		5	0	Enable Serial Downloading (enabled)
	4	0	Select start-up time		4	1	Watchdog always on (disabled)
	3	0	Select Clock Source ($8MHz$ internal RC)		3	1	Preserve E2 from Chip Erase (disabled)
	2	0	Select Clock Source		2	0	Select Boot size
	1	1	Select Clock Source		1	0	Select Boot size
	0	0	Select Clock Source		0	1	Select Reset Vector

Table 2.1.: AVR fuse bit configuration.

2.1.1. tinyOS

The ZigBit module is used with a software operating system called tinyOS. This OS has some features to rapidly run a system with some few lines of code. tinyOS has a lot of drivers for different platforms including those required for the ZigBit. For a test environment a software was build starting up tinyOS, initializing the radio, sending a few packets via radio and enter the sleep mode. This short test software has some traps if one does not know how tinyOS works internally.

Bootup timing

The first issue appeared at boot time. tinyOS in the standard configuration for a ATmega128 controller requires about 1 second plus X to startup. This is indeed incredibly long. The time is consumed by a module called ”MeasureClockC”. It starts the RTC-Timer (Timer2-Async mode) waits as many clock cycles as 1 second has to stabilize the $32KHz$ crystal. In general this is not wrong, because the RTC crystal really needs that time to settle. After settling, the function measures the system clock cycles ($8MHz$) per RTC clockcycle. This value is stored but never used. This is checked by searching the output of the NES-C compiler. So the result of this research allows to remove the calibration function. Never the less after removing this calibration routine the boot time can be compressed. With a test software running the boot-up , initializing the radio and sending three packets the measured time is approximately $110ms$.

But it must be pointed out that this only holds for the given configuration. For example the UART or the ADC employ this calculated calibration value. But in general, this does not mean that Timer2 is not used anymore. In deed it is but later in the boot process it is initialized correctly.

Powermanagment

The power management of tinyOS is mainly described in TEP112 [18]. But in the given simple configuration it is much easier to understand. TinyOS has a scheduler which manages queued tasks. A task may be posted by an interrupt or any other function. As soon as there is no more task in the queue the scheduler calls the function `McuSleepC__McuSleep__sleep()`. This function first checks which part of the hardware is used. And calls every component which provides a interface called `McuPowerOverride`. Both return a suggestion for a power state. After collecting all states tinyOS decides which the highest (less efficient) state is. Observing that all required functions still work.

External Clock disable

The third item to save power is to disable the external clock pin of the radio chip. The radio offers a output with its clock signal derived from the crystal. This output can be configured with a prescaler or turned off. Depending on the application this output is not required. To disable the output the RF-Driver must be set to the right configuration. The setting is located in `RadioConfig.h`. By changing the definition of a constant called `RF212_TRX_CTRL_0_VALUE` the content of the `TRX_CTRL_0` (Adr. 0x03) can be set.

To avoid changes in the repository of tinyOS one can change system files in a easy way. According to the man-page of the NES-C compiler it is reasonable to copy a file to the local `src` folder to prefer it. `RadioConfig.h` is located in:

`./tos/platforms/meshbean900/chips/rf212/RadioConfig.h`

By copying it to the local folder it is preferred by the compiler.

2.1.2. ZigBit Power consumption

A research about the ZigBit's power consumption under different conditions was made. To capture the current consumed by the module the test-software is adapted to hold the system into a defined state such that it consumes equally current over time. During this period an averaged current can be measured with a multimeter. The results are given in Tab. 2.2. Research shows that the RTC is initialized by the RF driver but not used in the current configuration. Therefore this is an option for power saving. The RTC is deinitialized directly after the boot procedure is finished. It is build into the `boot.done()` function. For accessing the RTC configuration it is necessary to wire the component `HplAtm1281Timer2AsyncC` into the code. A very important behavior is the explicit reset of the timer. Since every call from the power-manager asks the RTC driver for the remaining clocks to overflow. Its answer depends on the counter. Therefore in the worst case the system waits for a stopped counter and may never power down. One mode in row five shows a strange behavior but was not observed further. [1]

Descr.	Radio Stack	RTC	Lowest State	Current	Power @2.5V
Reset	-	-	-	1.5mA	3.75mW
TX	Transmitting	Ena	-	14.5mA	36.25mW
Idle	not Stopped	Ena	ATM128_POWER_IDLE	11.5mA	28.75mW
Idle	Stopped	Dis	ATM128_POWER_IDLE	2.34mA	5.85mW
Idle	Stopped	Ena	ATM128_POWER_DOWN	91μA/150μA	227μW/375μW
Power-Down	Stopped	Dis	ATM128_POWER_DOWN	2.7μA	6.75μW

Table 2.2.: ZigBit states vs. current consumption ("-" = don't care).

An additional evaluation of a wake-up event has been made on the ZigBit module. Fig. 2.2 shows the current vs. time. This diagram is explicit made for observation of the timing, not for accurate current measurement. [2] The event is triggered by an external interrupt. The controller runs the following procedure.

1. Wake up
2. Initialize the radio
3. Send three radio messages
4. De-initialize the radio
5. Return to power down mode

This sequence requires about 53ms. The three transmissions or radio-chip activities can be seen as raise in the current consumption for about 5ms. But obviously the length of the transmission depends on its payload length.

[1] During measurement an effect in a mode was observed which could not be fully explained. The mode in the fourth row starts with about 91μA current consumption and changed to 150μA after random time. This effect was recognized but ignored.

[2] The curve has a outstanding behavior which is due to measurement errors. The current seems like negative in the lowest power mode. This happens due to internal connections of the power supply and leakage currents.

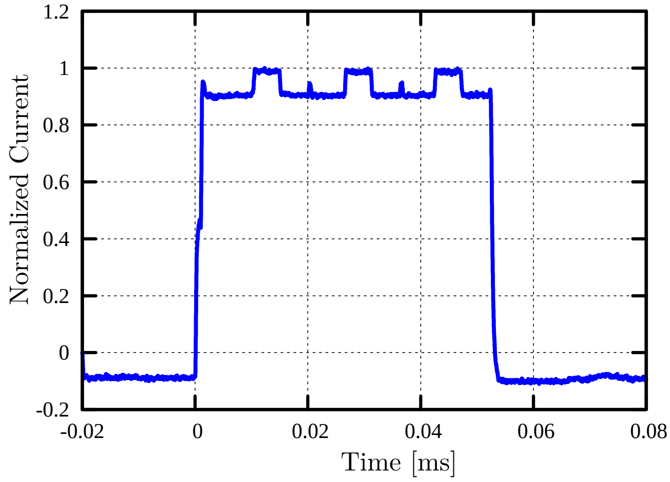


Figure 2.2.: ZigBit Current consumption at external interrupt wake-up.

2.2. Storage Device

The task of the storage device is to backup the power supply in case of source loss or to buffer the power for a period of activity. The storage device is treated in general independent of the other components to show what dimensions are required.

Assuming a situation where the node must survive in times of power loss. This could happens if no wind or not enough wind is available. If the assumption from Sec. 1.2.1 is used as reference, a period of 16 days without enough wind can occur.

Assuming that the node has to send a alive message or measures data every 2 seconds. The node's wake up frequency would be $2Hz$. Therefore a average current consumption can be calculated by

$$\begin{aligned} \overline{I} &= I_{pwrdown} + \overline{I}_{active} * t_{active} * f_{active} \\ \overline{I} &= 6\mu A + 14.5mA * 53ms * 0.5Hz = 388\mu A \end{aligned} \tag{2.1}$$

This is the current in the deepest sleep mode plus the average of the transmitting state as given in Tab. 2.2. Assuming a duration of 53 ms for sending three packages given by the previous chapter. Further it is assumed that this current must be provided continuously over the given period of 16 days. Ending up with a averaged current of approximately $388\mu A$.

In general there are two choices for storage devices. Either a electrolyte capacitor or a accumulator. For the given power consumption it turned out that only the accumulator is a reasonable solution. Eqn. 2.2 shows that a capacitor for backup would exceed the range of a real capacitor. The biggest capacitors available are approximately one or two Farad. Whereas the accumulator in Eqn. 2.3 is deep within the boundaries of available products. Cheap accumulators have at least $2500mAh$. Assume that $2000mAh$ can be used. This is a lifetime of roughly two thirds of a year. This is more than enough to survive. But if an accumulator is used the properties of a fully autonomous system is lost. This can be avoid by the choice of a proper power manager.

$$\text{Capacitor: } C = \frac{I * t}{\Delta U} \approx 388\mu A * 3600 * 24 * 16/0.5V = 1072F \tag{2.2}$$

$$\text{Accumulator: } C \approx 388\mu A * 24 * 16 = 149mAh \tag{2.3}$$

$$\text{Accumulator: } t \approx 2000mAh/(388\mu A * 24) = 214days \tag{2.4}$$

2.3. Voltage Regulator

2.3.1. Basics

Regulators are usually required to set a defined voltage level and provide a spurious free supply to the consumer. Another job is to change levels either from a higher to a lower or vice versa. Therefore the regulator must transform the unregulated generator output to a regulated voltage which is high enough to supply all components with stable conditions so that all components work reliably.

In general there are three types of switching regulators available, buck, boost and buck-boost. Their principle structure is shown in Fig. 2.3. The buck converter is able to regulate an output voltage across the capacitor to a level smaller than the input V1. Because with closed switch the capacitor is maximum loaded with the input voltage. The voltage is controlled by the pulse width of the switch. The boost converter is able to produce an output which is higher than the input. If switch S2 is open the output is equal to the input and the coil L2 is loaded. By opening the switch the energy stored in L2 is transfered to the capacitor which increases its voltage due to the additional charge. Therefore the output will be higher than the input voltage. The third converter is a combination of both. During the closed phase of the switch, L3 is loaded. Since the switch opens, the energy is transfered to C3. But the big disadvantage of this converter is its inverting character. If the Source V3 and the switch S3 is interchanged the voltage at the output node is negative compared to the input. Thus this converter requires a floating source.

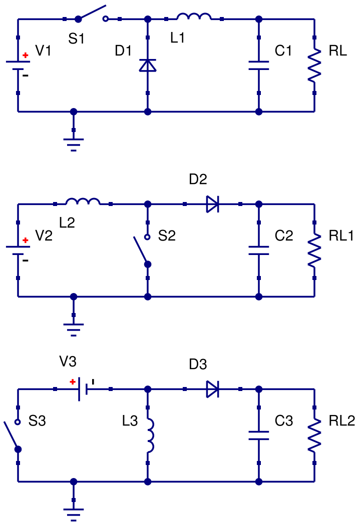


Figure 2.3.: Typical Switching Regulators.

The seeking for a proper regulator is difficult because of the very small input voltage below 0.7V. The most industrial types of switching-regulators have a minimal input power of approximately 1 Volt. A discussion about regulators for very small power applications was made by Paing and Zane [19] their goal is a regulator which is able to work from a low power source and provide good efficiency.

Unfortunately all switching regulators for very low voltage input must have a backup supply to start them working. The part of the regulator represented by the switch requires a supply to start working. The cause for this is in the semiconductor physics. A semiconductor needs a minimum amount of voltage to work. And integrated circuits with many transistors in series need a multiple of this. Therefore a backup capacitor or accumulator is required which is supported by the harvested energy. Otherwise the regulator is not able to power-up.

To work around this issue another type of integrated regulator is required. Linear Semiconductor offers a chip for micro power harvesting applications. The LTC3108 is a Power Manager for extremely low input voltages. Its main feature is a resonant step-up converter which allows to boost very low voltages to reasonable values. Further trials will show how it works and under which circumstances.

2.3.2. LTC - Exerimental Results

The power manager is made for sensor nodes with periodically power consumption so that there is enough time between the activities to reload the power tank. For example a node has a active period of 100ms and wake up every two seconds. So there is a period of two seconds to reload the tank to start a new power consuming period.

The device has two tanks shown in Fig. 2.4. One for backup of the active period of the load, it is C_{out}. And the other to backup the system for a long period in the absence of source power. C_{out} must be a capacitor so that it can be reloaded fast. The long term buffer could either be a capacitor or a accumulator. But as shown in previous calculations it will be a accumulator.

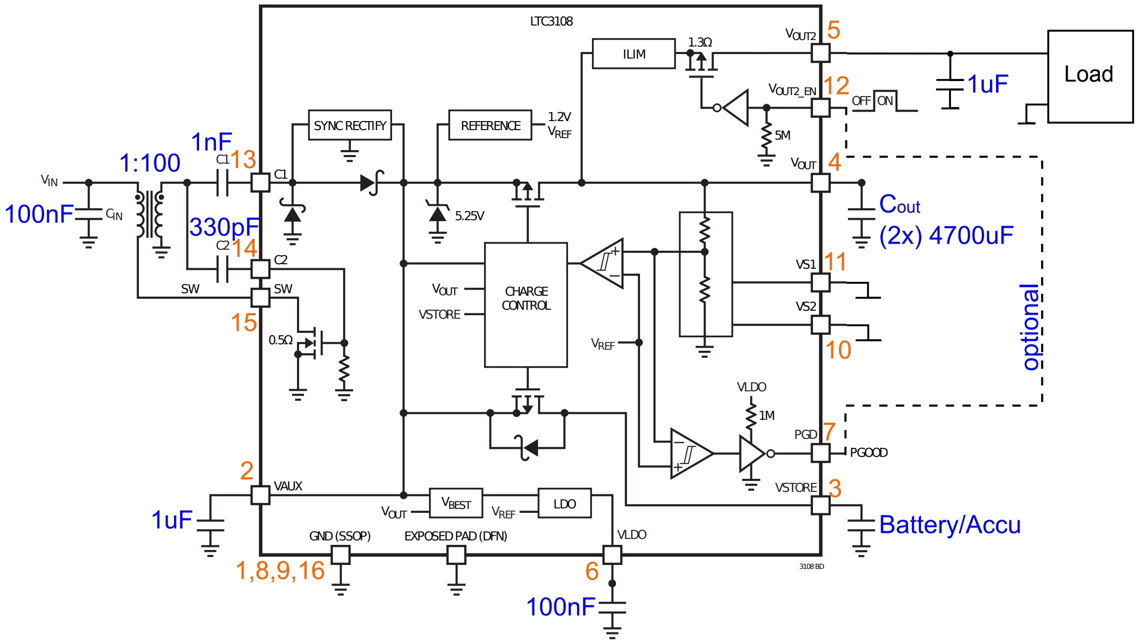


Figure 2.4.: LTC3108 evaluation schematic.

For the first prototype design with the IC a peripheral setup like Fig. 2.4 is chosen. The transformer is a 1:100, to push the voltage as much as possible. C2 and L of the secondary winding of the transformer form a parallel resonator with a resonance frequency of

$$f_{res} = \frac{1}{2\pi\sqrt{LC}} \tag{2.5}$$

This results in a frequency of approx. $32kHz$ [3]. C1 is the coupling between the transformer and the rectifier. It pushes the AC power into the rectifier from where it is distributed in the chip. VLDO is a fixed stabilized voltage. It is important to some voltage levels caused by pull resistors to that level. But it is not used externally. Pin 10 and 11 set the output voltage (V_{out}) to predefined values. Both connected to ground is equal to 2.35V for V_{out}. The long term buffer is not connected for the first trials because a constant source is available provided by a power supply. The consumer is emulated with a simple resistor equal to 150Ω sinking a current of $15mA$ (see Sec. 2.1.2). The short period buffer is selected to be

$$C \geq \frac{I \cdot t}{\Delta U} \tag{2.6}$$

with $I_{max} = 15mA$, $t_{PwrUp} = 110ms$ from a previous chapter. A allowed voltage drop of 211mV, derived by page 3 of [20] with a output of $2.35V$ and a drop of 9%. It turned out that C must be $7.8mF$. Two $4700uF$ capacitors are available and thus be used for C_{Out}.

Trial #1

Searching for the smallest possible input voltage turned out to be 30mV. With this minimal value all following test are run because more voltage means better conditions. The first trial has been made to verify an idea to automatically power up the system. If one connect the Power-Good-Output with the enable of V_{out2} the system could be automatically powered if enough power is available. Fig. 2.5 show the resulting curves.

[3] Indeed it is measured to be $46kHz$. Additionally it shows a little dependence on the available input power.

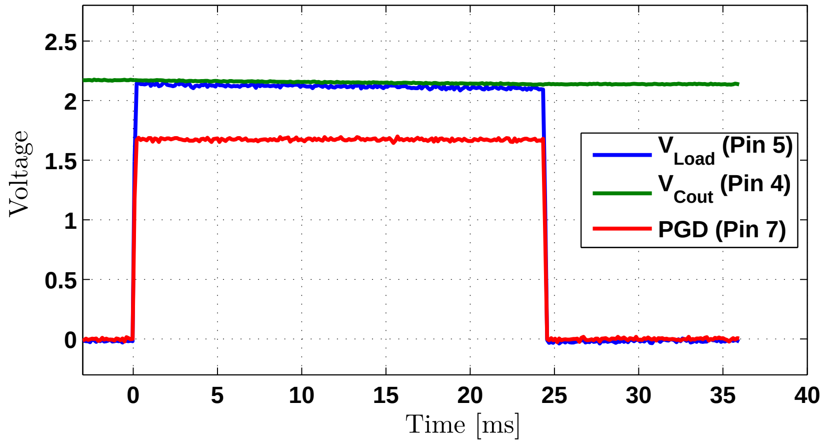


Figure 2.5.: Result Trial #1.

Unfortunately this mode is a bad idea for one reason. Assume that the Power-Good pins is activated at a threshold of -7,5% at rising voltage V_{out}. And deactivated at -9% of falling voltage. Referring to the nominal voltage of $2.35V$. That is a difference of $35mV$ which is available for activity of the load switched by V_{out2en}. Thus the system can use a drop of $35mV@15mA$ expecting a calculated (see Eqn. 2.6) duration of $\approx 22ms$. Compared to Fig. 2.5 with $24ms$ a valid assumption. The same holds for the load period. The input of the power manager is connected to a power supply showing $30mV$ and $1mA$ output. This is about $30\mu W$. This power is pumped to $2,35V$ resulting in a available current of $12,8\mu A$. It is assumed that this current is fully available for loading the buffer capacitor. With Eqn. 2.6 a recovery time of 25 seconds is expected. This value is not far away from reality. The measured time is 23 seconds.

Due to the short available period this is not a useful mode. It is proposed to use the two signals, PGOOD and V_{out2EN} as signals for the AVR or a additional control circuit.

Trial #2

Trial number two is about the enabling of the consumer after a long period of energy storage. The tank is fully loaded and the consumer is switched off. Fig. 2.6 shows this. V_{out2en} is switched manually. So the consumer represented by a resistor as mentioned before starts sinking current. So a fully loaded tank is able to run the system for approximately $150ms$ with the drop of $211mV$ as mentioned before.

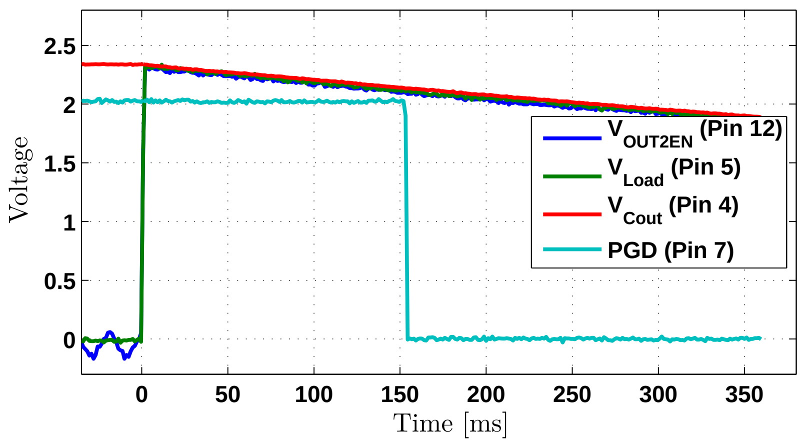


Figure 2.6.: Result Trial #2.

Trial #3

The third trial is a simple test how large a voltage source with 100Ω source resistance must be to power up the device. It turned out that $400mV$ is enough to start oscillating. But it is far away from efficiently loading the backup capacitance.

Some assumptions are made for further trials. The higher the available source power the faster the reload of the tank and therefore a higher activity of the consumer is possible. And secondly pay attention to the long term tank. If an accumulator is connected to the chip and the frequency of activity of the consumer is high. Than the energy will mostly come out of the long term tank and. But all in all the results of the tests look promising. The power-manager will be employed in further test environments.

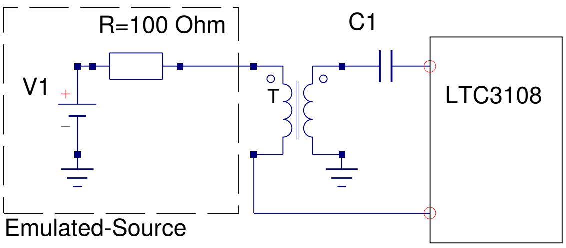


Figure 2.7.: Block diagram of emulated source.

2.4. Generator

The selection of the generator out of a limited amount of motors is made by trial and a few theoretical considerations. The induced voltage of a generator is mainly defined by the induction law.

$$u_{ind} = -\frac{d\Phi}{dt} = -\frac{d(\vec{B} \times \vec{A})}{dt} \tag{2.7}$$

Therefore there are three parameters in the equation. The frequency, area and field. It must be pointed out that some parameters are somehow contradicting to what the generator is for. As described in a previous chapter gust should be measured and therefore a little moment of inertia is required. The area and field strength is given by the generators. But more area or field means more moment of inertia. The parameter frequency is not selectable. It is influenced by the wind speed. The task is to find a given motor, out of Fig. 2.8, that can be used as a generator and matches the requirements.

To get an impression of the properties some motors are build into a test environment. The test environment consists of a driving motor, a photo-sensor to track the frequency and the motor under test. Indeed the device under test is a motor but since it is used as a generator it is called so. The driving motor is driven by a power supply and set to a speed according to a given frequency. The generators are tested with four different frequencies and with two different loads attached to their output. The motor itself is a voltage source with a source resistance. Therefore the output is loaded with different resistors to measure the source resistance. In Fig. 2.9 the environment is shown. On the left the driving motor and on the right the generator under test. The results are printed in Fig. 2.10. The digram shows the induced generator voltage versus the frequency of the shaft. In both diagrams the generators have different loads attached to the output. It turned out that two generators have a relative high output voltage compared to low frequencies. Therefore generator #1 and #2 are selected. Generator two has a smaller moment of inertia. Therefore it is used for the first trial.

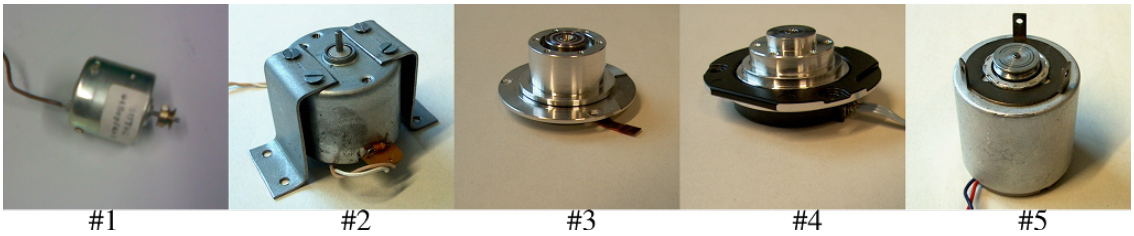


Figure 2.8.: Available motors for evaluation.

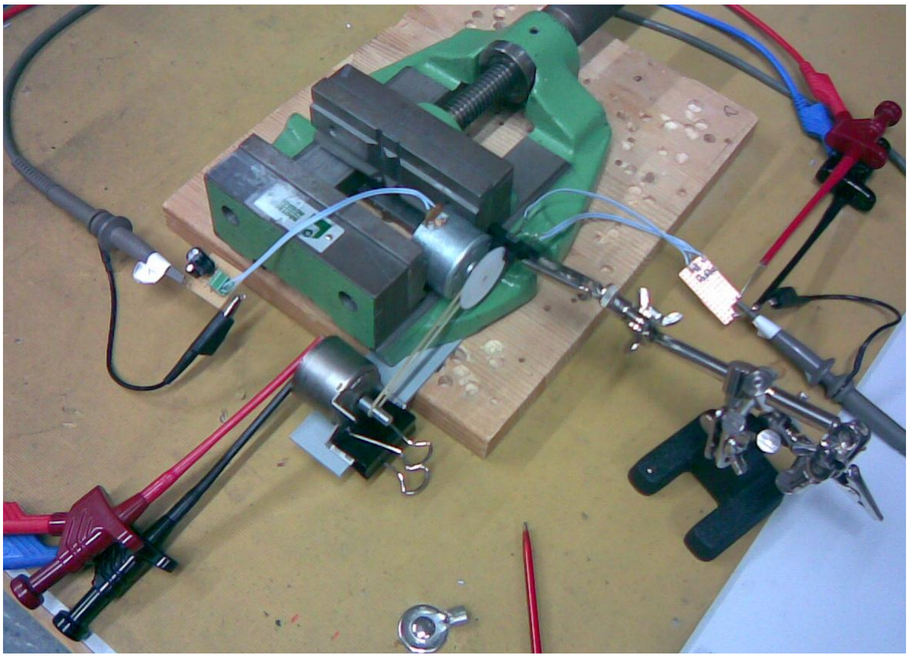


Figure 2.9.: Generator Test Environment.

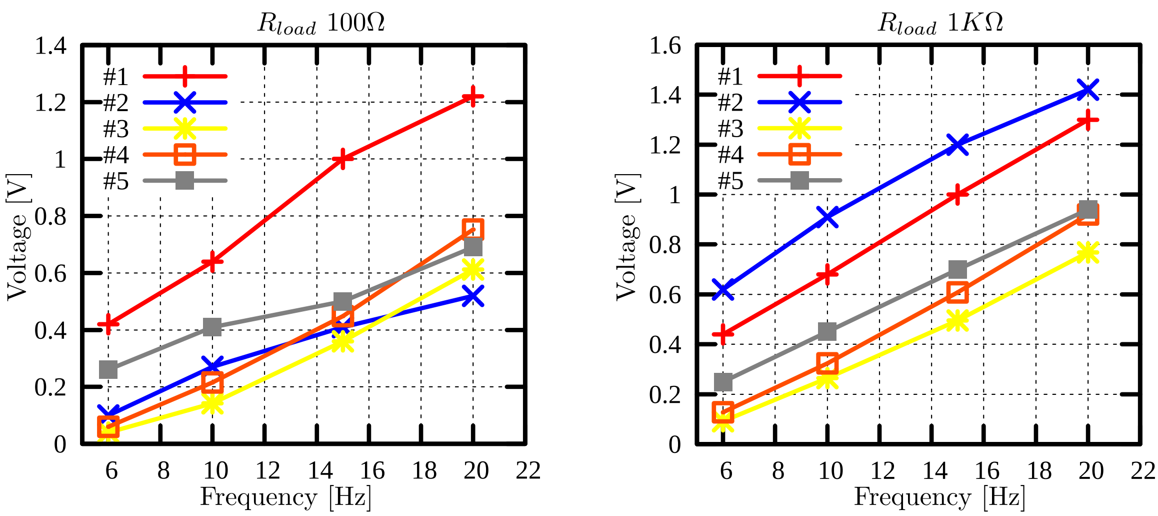


Figure 2.10.: Motor output voltage with different load.

3 Chapter 3. Prototype Deployment

3.1. Cup Anemometer

For the wind harvesting mechanical part a cup anemometer has been chosen. Following the idea of measuring, it is evident to employ this kind of windmill. It is obviously not the best for energy harvesting but it is a trade between measuring and harvesting. Additionally the cup anemometer is insensitive to the wind direction. This is helpful because the generator can be mounted in a fixed position. Otherwise the wiring is a problem because of twisting around the mounting-rod.

One device has been bought which is originally a wind speed measurement with a magnetic reed-contact inside to count pulses. The original device is totally disassembled and any part is removed from inside. To adapt the cups to the generator a bushing is required. Its manufacturing data is shown in the appendix in Sec. B.1. The mounted device with its connectors is shown in Fig. 3.1.

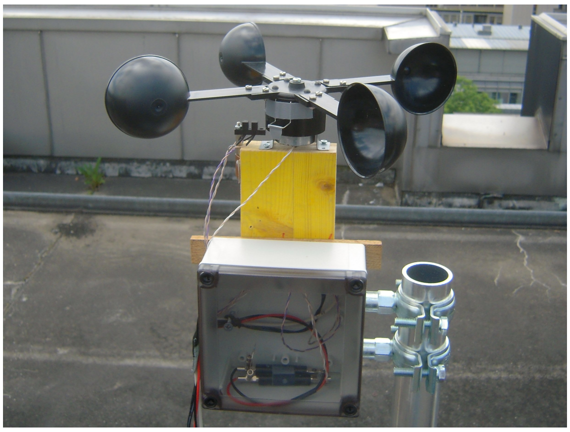


Figure 3.1.: Generator #2 Deployment.

3.2. Trial - Generator #2

Generator #2 has been mounted on top of the roof of the building to expose it to wind. The output voltage of the generator, the rotation frequency and the wind speed have been captured and analyzed. It turned out that with a load of 100Ω the amount of energy and accordingly the voltage at the port is too low. This generator is, due to its little moment of inertia not suitable for the application. The same turned out with a connection to the power manager. The LTC3108 is not able to start working even with turning frequencies of approximately $3Hz$.

3.3. Trial - Generator #1

Number 1 is the generator with a stronger field. This can be noticed when turning the shaft by hand. A first shot with the generator mounted to the power manger and turned by hand looks promising. The regulator starts working in a very minimalistic way, but it works. With this generator nearly the same deployment is build and used first inside the laboratory. The output (V_{OUT}) is loaded with a $220K\Omega$ resistor, see Eqn. 3.1 for dimensioning. The current equals two times the standby current of $3\mu A$ plus another $6\mu A$ for additional circuitry.

$$\frac{2.35V}{12\mu A} \approx 196K\Omega \rightarrow 220K\Omega \rightarrow 10.7\mu A \tag{3.1}$$

This margin is necessary because the mode with the least power consumption requires external components. According to Tab. 2.2 and the AVR data-sheet the mode can only quit with an external interrupt.

The environment of the measurement setup is shown in Fig. 3.2. The associated results in Fig. 3.3. The lines are labeled according to the schematic of the regulator in Fig. 2.4.

The shown traces are an extract of about 3 hours of measurement. The system is not able to work at wind velocities of less than $2m/s$. Due to uncontrollable wind situations this a time slot where a velocity above the threshold is remaining for a reasonable time. The rest of the measurement is not that interesting because of no or very little activity.

The bold solid line shows the Pin V_{aux} of the LTC3108, the light curve in the background is the wind velocity measured by the reference device. This is either a hand-held device for in-house or the weather station of the PermaSense project(Sec. A.4) outside. Here one can see the start-up of the regulator at velocities of greater than 2 m/s. At this point the oscillating transformer starts pushing the voltage at the V_{out} pin to 2.5 V. Since enough energy is available, the output V_{out} starts to source current. Therefore the capacitor voltage is increasing. This happens at velocities of greater than 2.5 m/s. Shorter low power wind gusts, have not enough energy to drive the anemometer so the electronic try to start but cannot. This can be observed by the peaks in the V_{out} curve.

The thin dashed lines show the slope of the load curves at interesting points. With the slope one can approximate when the maximum voltage of the capacitor is reached. Unfortunately under given conditions it will take up to 1000 seconds (16 minutes) to load the capacitor. But on the other hand, $3m/s$ velocity is not that much. It is expected based on formulas that higher velocities cause more available power ergo higher current to load the capacitor. A linear behavior of the load curve is assumed because the time constant of the RC combination is in the order of 2000 seconds. This is twice as much as the observed range and therefore the exponential load curve can be treated as linear.

How much power is delivered by the source? Due to not knowing the source current and voltage the output of the power manager is used to estimate the power. Knowing the slope of the voltage one can calculate the current into the capacitor.

$$C\Delta U = I\Delta t \tag{3.2}$$

$$I = \frac{C\Delta U}{t} = 4700\mu F * 2.35V/1000s = 22\mu A \tag{3.3}$$

$$P = V \cdot I = 2.35V \cdot 22\mu A = 51.9\mu W \tag{3.4}$$

This power is delivered to the consumer and the capacitor. Compared with a result computed in the first chapter of some Milli Watts it is very little. From Eqn. 1.5 the formula for available wind power is known. It can be shown that the wind speed measured by the reference device in Fig. 3.3 between 50 and 200 seconds is $3.35m/s$ in average. Inserting the data for the anemometer from Tab. 3.1. The available harvesting wind power would be

$$P = P_0 \cdot c_p = P_0 = \frac{1}{2}\varrho\pi(3.2cm)^2(3.35m/s)^3 0.2 = 14.5mW \tag{3.5}$$

One could think this is a mistake, in fact it is not. It happens that the generator is running at the maximum harvesting ratio, see Betzlimit in Sec. 1.2.3. In Fig. 3.3 the average of the rotation frequency is $1.7Hz$ for the same time range as before. This is according to the anemometer data a velocity of

$$v = 2\pi r f = 1.28m/s \tag{3.6}$$

and the ratio between the real wind speed and the anemometer speed is

$$\frac{v_w}{v_r} = \frac{1.28m/s}{3.35m/s} = 0.38 \tag{3.7}$$

According to Betz's limit, the maximum is at 0.33. So the assumption of a harvesting ratio of 20% still holds. The question remains. Where the power is lost. It is in fact a question of the generator efficiency and the power manager's power consumption and efficiency. Due to the fact that the generator is not a optimized device for a generator application because it is build to work as a motor. It is assumed that the major part is lost in mechanical and magnetic losses. And secondly according to the datasheet of the power manager the efficiency for low input voltage is less than 10%. Therefore the results doesn't look uncommon.

Cup ∅	: 6.4cm
Cup A	: $\pi r^2 = 3.217 \cdot 10^{-3} m^2$
Anemometer ∅	: 24 cm

Table 3.1.: Cup Anemometer Data

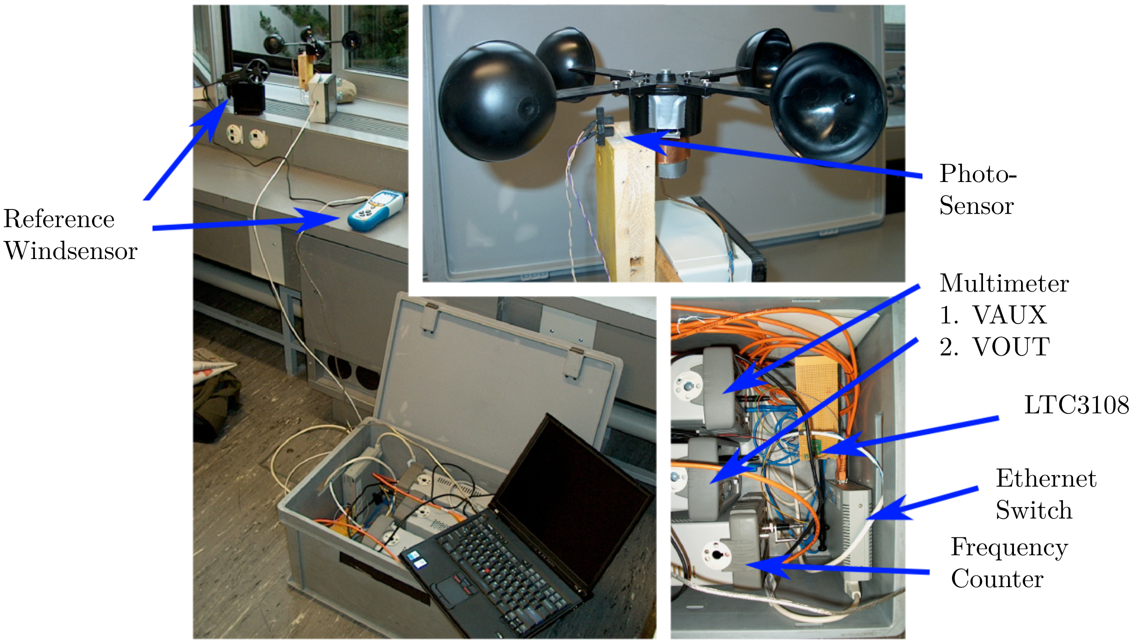


Figure 3.2.: Generator #2 Laboratory Deployment.

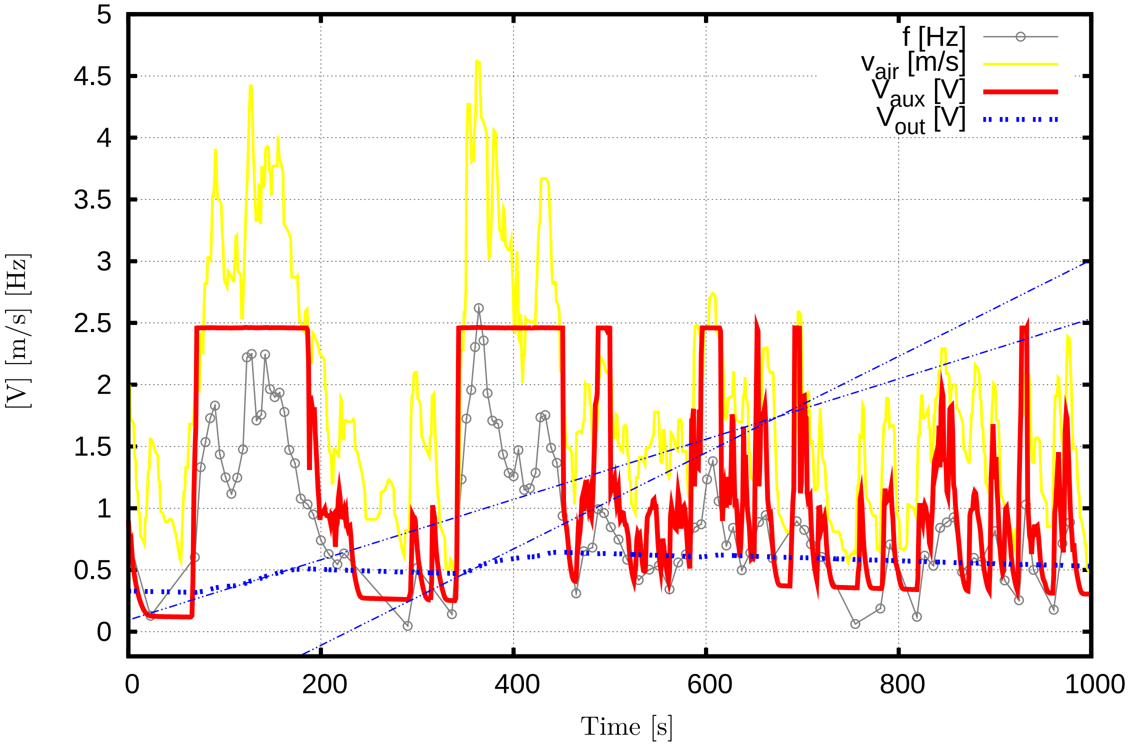


Figure 3.3.: Generator #2 Laboratory Results.

Chapter 4.

4 System Pre-Design

In this chapter one possible concept for a prototype has been worked out with its advantages and problems. The prototype is not build yet so this is just a design proposal. The concept is based on the schematic diagram in Fig. 4.1. The system is designed to the least amount of additional components. Never the less a few are necessary to control the startup and power down sequence. The processor system is attached to the switchable output (V_{OUT2}) of the power manager. The control ping for V_{OUT2} is called V_{OUT2EN} and driven by the comparator or a processor pin. Both together must form a OR combination. Assume a system with no power stored in the buffers. As soon as the generator makes power available, the short term buffer will store energy. With the comparator the available power in the short term buffer is observed. If the voltage exceeds a reference level the comparator can switch on the processor. The reference is supplied by the LDO output. This is a low power regulated source at 2.2V. The threshold should be at approx. 95% of the nominal value. At this time the pin `PGOOD` is already set because its triggering threshold should be less than the one of the comparator.

So if enough power is available the system is starting up. Now the processor must set the `PWR_ON` line as fast as possible. Because during operation the voltage of the buffer capacitance is decreasing. In fact the comparator must have a hysteresis but the power must be locked within the drop to the lower threshold of the comparator. Otherwise a falling level can reset the `PWR_UP` signal and make the system again power down. It is estimated that the time to set the pin and therefore lock the system can be power by the buffer.

As said before a running processor will decrease the buffer level and reset `PWR_UP`. The processor can monitor that but doesn't have to respond. And as long as the internal comparator of the power manager keeps `PGOOD` high the processor can go on. As soon as the voltage drops below 2.1385V a critical level for the processor is reached this is indicated by `PGOOD` to low level. This means there is a range of another 0.3V until the processor will get in trouble. According to the datasheet it works down to 1.8V else a brown-out-reset occurs. Make sure that the AVR works with the brown-out-reset enabled otherwise the declining voltage could cause problems.

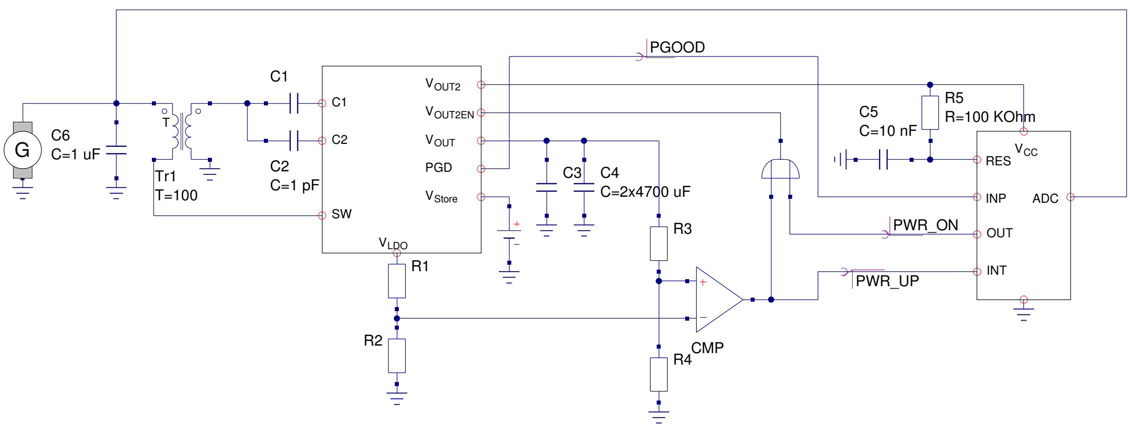


Figure 4.1.: Schematic Proposal for Prototype.

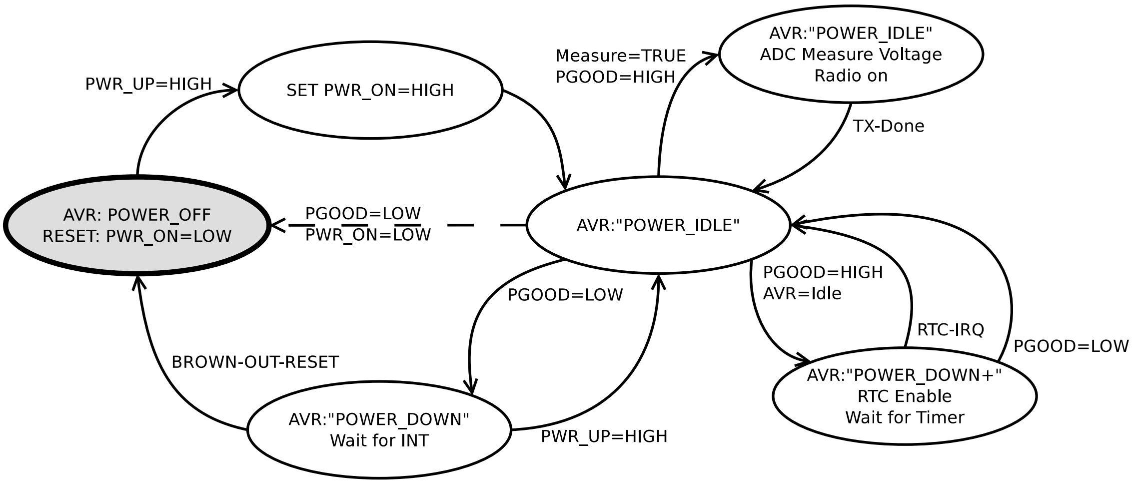


Figure 4.2.: System State Diagramme.

Here the results from Sec. 2.3.2, trial #2 are important. At a certain time the processor must either go to a standby mode or disconnect its own supply by setting `PWR_ON` to low level. This decision depends on how fast the reset of `PWR_UP` and `PGOOD` appears. Remember that the deepest sleep mode must have a external interrupt for wake. That is the reason for connecting `PWR_UP` to one of the INT pins.

It is obvious that both, the comparator and the logic gate have to be very low power devices. Otherwise their current consumption will exceed the standby of the processor's. Potentially the ADC requires some external circuit like level matching or amplification.

Recall that the main task of the node is wind velocity measuring. Measuring the voltage at the generator is somehow related to the wind speed. But clearly the voltage depends on the load attached to the generator and thus depends on the energy required by the system. It is to be observed what the correlation between both indicators is. The first concept was a control circuit to disconnect the power manager from the generator. In this mode a measurement can be made and afterwards the generator can be reconnected. But due to the input transformer of the power manger it will be a challenge to develop a circuit.

The software architecture for the AVR is shown as a state diagram in Fig. 4.2. The states for the AVR are labeled according to Tab. 2.2. In `POWER_OFF` state the AVR or in general the whole ZigBit module has no power. The processor I/Os are floating. `POWER_DOWN` is the deepest sleep mode without RTC support. This state can be left only by a external interrupt. `POWER_DOWN+` is one level higher, it includes RTC support. The processor can wakeup by a RTC interrupt, usually fired by a overflow of the counter register. `POWER_IDLE` is every state in which the processor is running code and is not stopped. Either a calculation is running or a idle loop is running.

The dashed transition is the one where the processor is able to switch its own power off. This path is dangerous and it is not fully evaluated if this is properly working. Possibly the fast execution time requires another code for the AVR than the standard tinyOS or at least modifications in the boot process.

Chapter 5.

5 Discussion

A system for a sensor node has theoretically been designed and partially evaluated. A research about the basics of wind energy harvesting have been worked out. And a set of requirements are defined for most of the components of the system. But the main focus will be laid on the energy harvesting part and the evaluation of the single components.

Currently the project is in a prototype phase where nothing is able to be used for a network of sensors. In fact the project reached a level from where one can start building a node. But it is far away from setting up small series of nodes. Indeed it is one of the future goals to build a small series of nodes. But this requires to leave the prototype stage behind and select components available in equal quality and quantity. Especially the mechanical parts must be made more solid.

Beside series production problems there are many topics left for evaluation. Most of them are practical topics which simply take a lot of time to prepare and setup. One topic to be theoretically analyzed is the optimization of the generator. There are many parameters in selecting a good generator or even try to develop a optimized one matching the application. This requires a lot of knowledge in electro-magnetics.

Measurements are the most necessary things to do. One is the response to higher wind velocities. Up to the time of writing the report it was not windy enough to capture real data. It is of interest to see wind of more then 5 m/s or wind strength of minimum level 4. Further topics are about the accumulator. As long as the system is configured to supply 2,35 Volt it should be sufficient to attach two 1.2 Volt accumulators. It would be interesting to see the current flow sourcing or sinking into the accumulator under this configuration. The value of interest is the threshold voltage of the input. So that the accumulator is charging or discharging. Depending on the direction of energy flow the question is how much the generator supports the system. Two setups could be assembled. One without a generator attached to the power-manager and one without. Comparing both setups by measuring the accumulator capacity. It is expected that the one without support is not able to work as long as the other. A hint for the setup could be to use a real consumer. Attaching it to the power manager is a more realistic case. It is proposed to use a ZigBit module running a program equal to the one in Chap. 4. By capturing the different power signals the interaction between wakeup and sleep states can be observed.

What has been neglected since the main application requires a particular type of device is the wind mill itself. Since it was defined that the main topic is the wind measurement, the whole system analysis was done with a cup anemometer. But it turned out that not the measurement is the big problem rather than the harvesting. That is due to the contradicting requirements for measuring and harvesting. The focus for further development must be the energy harvesting. Therefore it is proposed to go on with another type of wheel. Not a drag device but rather a lift device. And if in the end the correlation between the wind speed and the output voltage is good enough than it can be used for measuring. So it seems to be a that lot of work still open and ready for another thesis.

Appendix A.

A Measurement Equippment

All measurements with a large time span are captured by Linux shell scripts. The Linux shell, in particular the bash shell is one of the powerful tools of Linux. With the bash it is possible to read, modify and write data streams in various combinations. This has been utilized in the testing environments. It must be said that this is not a introduction to bash and its tricks. A good introduction and many examples are provided by [21].

One thing every setup has in common, is to add a time-stamp to the data stream. This timestamps are set in UNIX format. It counts second from the 1st of January 1970 beginning at 00:00. It is the easiest method for post processing timestamps because they are directly comparable to each other.

A.1. Multimeter - Agilent 34411A

To access the multimeter a LAN port is available. With different tools and tricks it is possible to use a TCP socket in a bash script. The bash script is given here:

```
#!/bin/bash

meas_loop(){
        echo "CONFigure:VOLTage:DC" >&3
        echo "Start" >&4
        while [ true ]
        do
                echo -n "$(date␣+%s\␣%N)␣" >&4
                echo "READ?" >&3
                sleep 1
        done
}

IP="192.168.0.2"        #IP address of dmm1

term=$(who -m | sed -e 's/^[[:alnum:]]*\ *\([[:alnum:]/]*\)\ *.*$
   /\1/g')
echo "Terminal:␣$term"

fname=/home/user/dmm/dmm1.txt   #filename to save data

# Descriptor 4 is the logfile and the pipe from LAN into the file
exec 4>$fname
# Descriptor 3 is the command output to the equipment
exec 3> >(netcat $IP 5025 | tee /dev/$term >&4)
echo "netcat␣seems␣OK..."

# General System Settings
echo "*RST" >&3
```

```
echo "SYSTem:BEEPer:STATe␣OFF" >&3
echo "DISPlay␣ON" >&3

meas_loop        # start data transmission loop

echo "End␣measuring!"
killall netcat          # kill the socket
exec 3>&-               # close descriptor
```

A full description of the commands for the multimeter can be found in the user manual of the 34411A [22, 23, p.105 ff].

A.2. Frequency Counter - Agilent 53131A

The frequency counter is one of the older devices thus there is no USB or LAN port. But there is a serial printer output. It turned out that this is nothing else than a simple serial UART with 9600 bps and ASCII character output. The device sends the frequency to this port every time a measurement is finished. In this case a gate-time of 3 seconds is chosen. So the results arrive maximal every 3 seconds.

This output can be read by a computer using a serial to USB converter. The converter itself registers in Linux as a tty device and therefore can be read with a simple script.

```
#!/bin/bash
stty -F /dev/sttyUSB0 9600
cat /dev/sttyUSB0 | while read l; do echo -e "$(date␣+%s)\t$l";
done | tee output.txt
```

Before using the Linux device directly, the port must be setup. That is what the first line does.

A.3. PeakTech 5060

The PeackTech 5060 is a anemometer and temperature sensor. It has a USB port and is recognized as a serial device by the Linux kernel. The anemometer is a bit more complex to understand because it does not deliver ASCII data. Instead of viewable characters it is raw binary data. The serial port is running on "4800,8,N,1" configuration. In principle it is still possible to open the /dev/ttyUSBx device, configure its baudrate and try to receive data. In fact it turned out to be not a good idea. This method delivers corrupted frames. A view into the binary data shows that the frame lengths are unequal. Somewhere in the chain bytes are lost.

To not fell into deep analyzes of this behavior a fast workaround was constructed with a python program. A library to use serial ports in python already exists. It is small, easy to install and easy to implement. The library can be found under the key word pyserial [24]. The implementations looks like this:

```
>>> import serial
>>> ser = serial.Serial('/dev/ttyS1', 4800)
>>> x = ser.read()             # read one byte
>>> s = ser.read(10)           # read up to ten bytes
>>> line = ser.readline()      # read a '\n' terminated line
>>> ser.close()
```

If pyserial does not accept the string as device it could be replaced by a number representing the serial port number. Unfortunately the library only accepts numbers from ttySx. To use a USB-Serial device a symbolic link is required from /dev/ttyUSBx to a new /dev/ttySx device. And pyserial will understand the parameter.

A.4. Reference Wind-speed Data - Vaisala Weather station

For any outdoor measurements is is not practical to use the PeakTech 5060 Anemometer because it is not able to turn into the wind. Therefore another reference source is required which is independent of the wind direction. Fortunately there is a weather station from the Permasense Project at ETH which can be used via a web interface. The URL for the raw data is `http://tik31x.ee.ethz.ch:22001/`. The data is supplied in a ASCII formated string. The encoding is describe in the user guide of the weather station [25, p.70]. An example string looks like the following:

```
wu 0R1,Dn=111D,Dm=149D,Dx=209D,Sn=1.5K,Sm=8.1K,Sx=14.9K
```

The value of interest is called `"Sm"`. It is the averaged wind speed of one minute in kilometers per hour.

A.5. Deployment - Measurement Setup

This is the setup used in Chap. 3 in different variations. For the roof deployment the reference sensor is not the handheld device. Instead the PermaSense data were used.

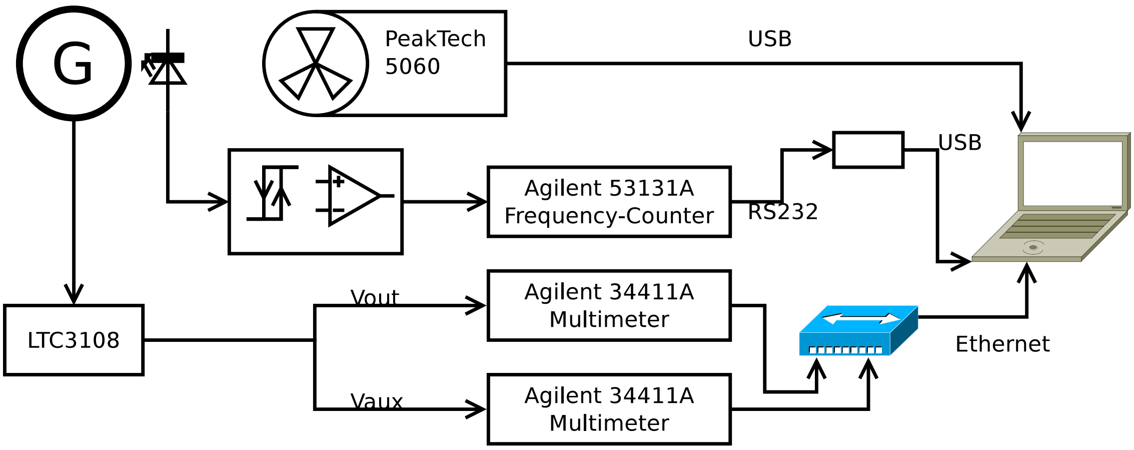


Figure A.1.: Mesurement setup

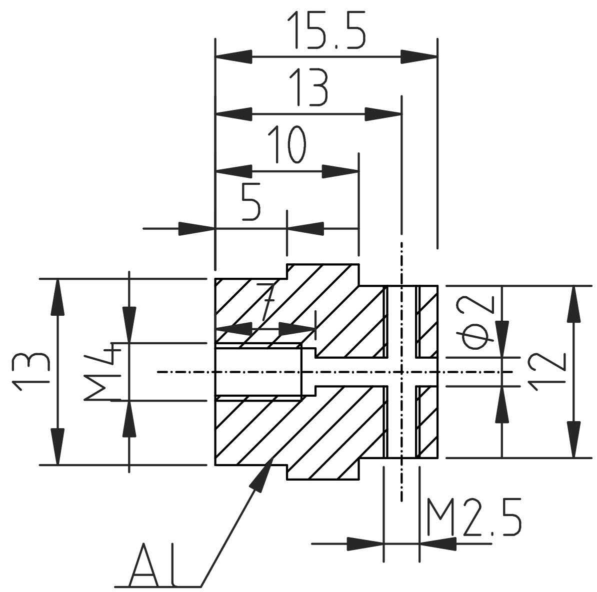
Appendix B.

B.1. Mechanical Drawings

This is the mechanical drawing the workshop requires to produce additional parts.

B.2. Wind Data - Roof Deployment

The whole deployment was setup for four days on top of the building. Unfortunately there was not enough wind to show the full energy harvesting capability. The results are attached in the appendix because it does not give any further knowledge.



Masstab 2 : 1

(Gewicht)

Aluminium

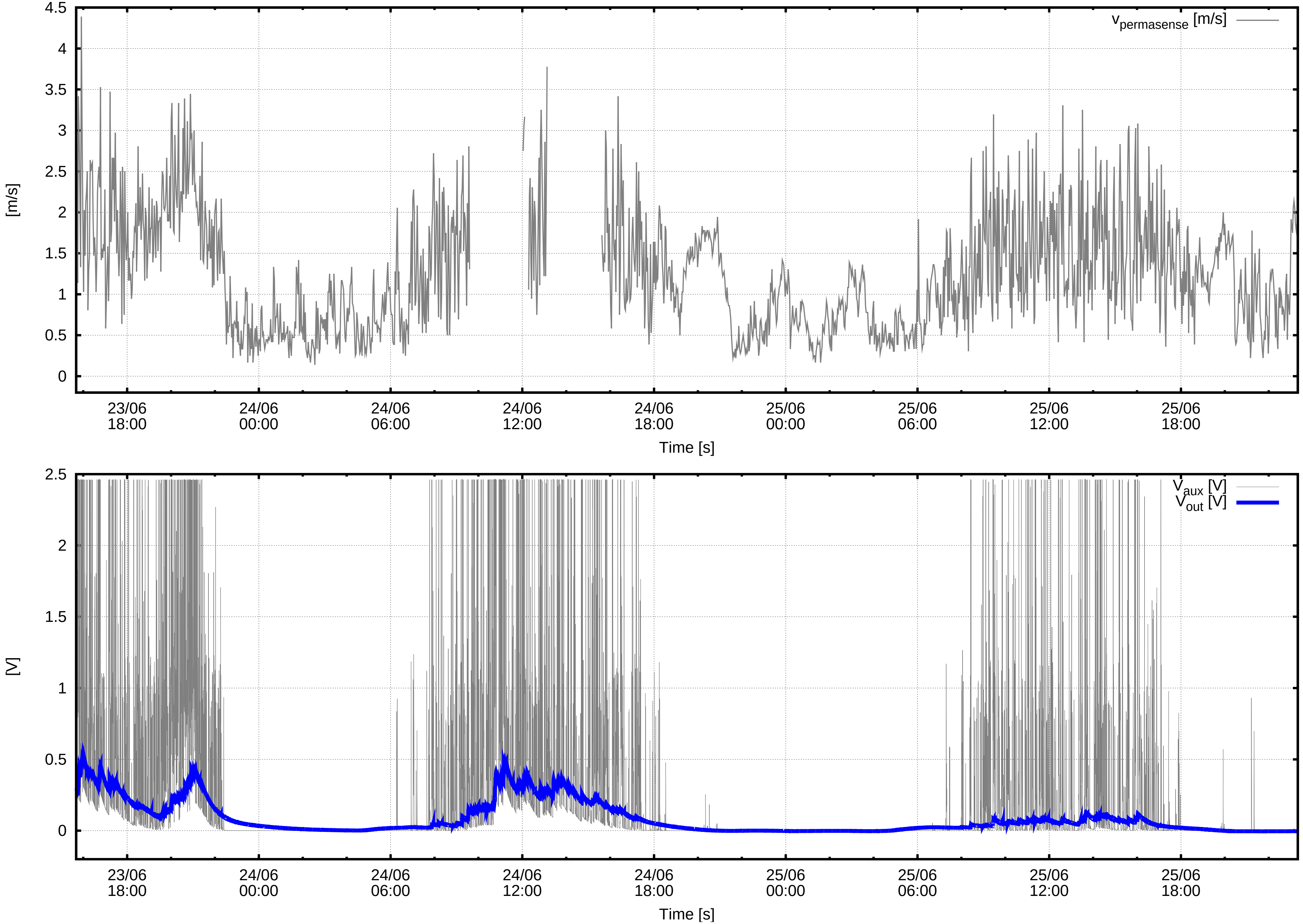
	Datum	Name
Bearb		
Gepr		
Norm		

Anemometer-Generator Adapter 2
(#8)

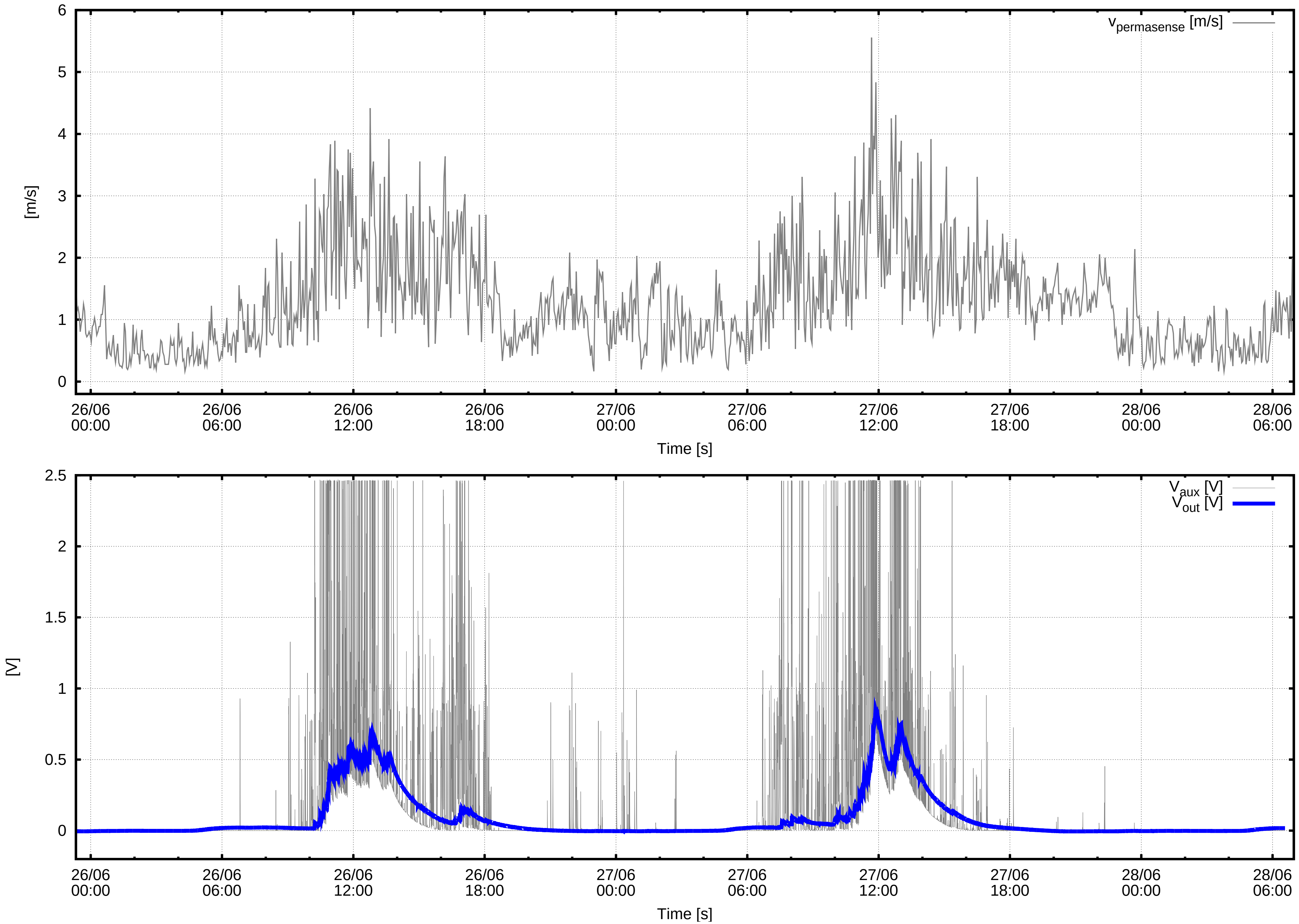
Oliver Schultes

1 / 1

Zust nderung | Datum | Name | Ursprung



4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
[m/s]
$v_{permasense}$ [m/s]
23/06 18:00
24/06 00:00
24/06 06:00
24/06 12:00
24/06 18:00
25/06 00:00
25/06 06:00
25/06 12:00
25/06 18:00
Time [s]
2.5
2
1.5
1
0.5
0
[V]
V_{aux} [V]
V_{out} [V]
23/06 18:00
24/06 00:00
24/06 06:00
24/06 12:00
24/06 18:00
25/06 00:00
25/06 06:00
25/06 12:00
25/06 18:00
Time [s]



v_permasense [m/s]
[m/s]
6
5
4
3
2
1
0
26/06 00:00
26/06 06:00
26/06 12:00
26/06 18:00
27/06 00:00
27/06 06:00
27/06 12:00
27/06 18:00
28/06 00:00
28/06 06:00
Time [s]
V_aux [V]
V_out [V]
[V]
2.5
2
1.5
1
0.5
0
26/06 00:00
26/06 06:00
26/06 12:00
26/06 18:00
27/06 00:00
27/06 06:00
27/06 12:00
27/06 18:00
28/06 00:00
28/06 06:00
Time [s]

Bibliography

[1] B. für Statistik, "Bodennutzung und Bodenbedeckung," 2010. [Online]. Available: http://www.bfs.admin.ch/bfs/portal/de/index/themen/02/03/blank/key/01/zustand_und_entwicklung__tabelle.html 7

[2] Meteo Swiss, *SwissMetNet*, 2010. [Online]. Available: http://www.meteoschweiz.admin.ch/web/de/klima/messsysteme/boden/swissmetnet.html 7

[3] E. Hau, *Wind Turbines*. Springer Berlin Heidelberg, 2006. [Online]. Available: http://www.springerlink.com/content/g65656/ 8, 9, 10, 11

[4] A. J. Bowen and H. Clucas, "The measurement and interpretation of peak-gust wind-speeds over an isolated hill," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 41-44, pp. 381–392, 1992. 8

[5] D. W. D. (DWD), "Tägliche Windspitzen in Beaufort von Deutschland ," 2009. [Online]. Available: http://www.dwd.de/ 9

[6] P. A. Sheppard, "An improved design of cup anemometer," *Journal of Scientific Instruments*, vol. 17, no. 9, p. 218, 1940. [Online]. Available: http://stacks.iop.org/0950-7671/17/i=9/a=301 10

[7] F. J. Scrase and P. A. Sheppard, "The errors of cup anemometers in fluctuating winds," *Journal of Scientific Instruments*, vol. 21, no. 9, p. 160, 1944. [Online]. Available: http://stacks.iop.org/0950-7671/21/i=9/a=304 10

[8] E. L. Deacon, "The over-estimation error of cup anemometers in fluctuating winds," *Journal of Scientific Instruments*, vol. 28, no. 8, p. 231, 1951. [Online]. Available: http://stacks.iop.org/0950-7671/28/i=8/a=303 10

[9] H. Eckelmann, *Einführung in die Strömungsmeßtechnik*. Stuttgart: B. G. Teubner, 1997. 10

[10] "Wikipedia: Anemometer," Apr. 2010. [Online]. Available: http://de.wikipedia.org/w/index.php?title=Anemometer&oldid=71615262 10

[11] T. Foken, *Angewandte Meteorologie*, 2nd ed. Springer Berlin Heidelberg, 2006. 10

[12] M. Sathyajith, *Basics of wind energy conversion*. Springer Berlin Heidelberg, 2006. [Online]. Available: http://www.springerlink.com/content/m6r5700848253473/ 11

[13] K. John, F. Steven, K. David, A. Elsayed, A. Roger, S. William, A. Anthony, J. Chand, Y. G. D, P. Roberto, T. James, S. Thomas, B. Dale, B. Carl, M. Gregory, R. Kitt, R. Mysore, F. Jean-Pierre, J. William, and L. Alex, "Energy conversion," in *Handbook Series for Mechanical Engineering*. CRC Press, Dec. 2009, vol. null, ch. 8.11, pp. –. [Online]. Available: http://dx.doi.org/10.1201/9781420041583.ch8 11

[14] "Wikipedia: Density of air," Apr. 2010. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Density_of_air&oldid=346850915 11

[15] R. Ang, Y. Tan, and S. Panda, "Energy harvesting for autonomous wind sensor in remote area," Nov. 2007, pp. 2104–2109. 12

[16] M. A. Weimer, T. S. Paing, and R. Zane, “Remote area wind energy harvesting for low-power autonomous sensors,” june 2006, pp. 1–5. 12, 13

[17] *ATmega640/1280/1281/2560/2561*, Atmel Corp., 2008, 2549L–AVR–08/07. [Online]. Available: http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf 15

[18] S. Robert, L. Philip *et al.*, *Microcontroller Power Management*, draft ed., tinyOS, Sep. 2005. 16

[19] T. Paing and R. Zane, “Resistor emulation approach to low-power energy harvesting,” Jun. 2006, pp. 1–7. 19

[20] *Ultralow Voltage Step-Up Converter and Power Manager*, Linear Technology Corporation, Apr. 2010, rev A. 20

[21] M. Cooper, *Advanced Bash-Scripting Guide*, Mar. 2010, revision 6.2. [Online]. Available: http://tldp.org/LDP/abs/html/index.html 33

[22] *Agilent 34401A 6 1/2 Digit Multimeter - User's Guide*, Seventh edition ed., Agilent Technologies, Inc., 3501 Sevens Creed Blvd. Santa Clara, CA 95052 USA, Aug. 2007. 34

[23] *Agilent 34410A/11A 6 1/2 Digit Multimeter - User's Guide*, Fourth edition ed., Agilent Technologies, Inc., 3501 Sevens Creed Blvd. Santa Clara, CA 95052 USA, Feb. 2007. 34

[24] *Python Serial Port Library Documentation*. [Online]. Available: http://pyserial.sourceforge.net 34

[25] *Vaisala WXT520 User Guide*. [Online]. Available: http://www.vaisala.com/files/WXT520_User_Guide_in_English.pdf 35