

Fachbericht

Wetterstation mit Solar Energie

Windisch, 28. November 2018

Hochschule	Hochschule für Technik - FHNW
Studiengang	Elektro- und Informationstechnik
Autor/-en	Mischa Knupfer, Andres Minder
Betreuer	Prof. Dr. Taoufik Nouri
Auftraggeber	Prof. Dr. Taoufik Nouri
Version	1.0

Abstract

Inhaltsverzeichnis

1	Auftragsbeschreibung	2
2	Ziele	3
3	MCU	5
4	RTC	6
5	Sensoren	7
5.1	Messen der Lufttemperatur	7
5.2	Ermittlung der Niederschlagsmenge	7
5.3	Anemometer	9
5.4	Zählung der Sonnenstunden	9
6	Datenspeicherung	10
6.1	Breakoutboard	10
6.1.1	Verdrahtung	10
6.1.2	SPI (Serial Peripheral Interface)	11
6.2	μ SD-Karte	11
6.3	Implementation in die Firmware	12
7	Kommunikationsmodul	13
8	Konzeptvalidierung	14
9	Literatur	15
A	Lastenheft	16

Einleitung

1 Auftragsbeschreibung

Das Wetter spielt eine wichtige Rolle in der Agronomie. Regnet es nicht genug, müssen Pflanzen bewässert werden. Trifft auf ein Ort nur wenig Sonnenlicht, so sollten dort nicht die Pflanzen, welche viel Sonnenlicht brauchen, angebaut werden. Windet es zu stark, können Pflanzen beschädigt oder gar zerstört werden. Ist es Tagsüber heiss, so benötigen die Pflanzen mehr Wasser. Hiesige Bauern besitzen den Luxus von guten Wettervorhersagen dank dem Bundesamt für Meteorologie und Klimatologie (MeteoSchweiz). Dieser Luxus ist in anderen Ländern noch nicht gegeben. Prof. Dr. Nouri Taoufik ist aufgefallen, dass in tropischen Gegenden wie Südamerika oder teile Afrikas dieser Luxus ebenso fehlt.

Aus diesem Grund soll eine kostengünstige, erweiterbare und mobile Wetterstation entwickelt werden, welche diese Bauern unterstützt. Diese Wetterstation soll die Regenmenge, die Windstärke, die Lufttemperatur und die Sonnenstunden messen können. Ausserdem soll die Wetterstation mittels Photovoltaik unterstützt werden, und erhobene Daten via SMS abrufbar sein.

Im Nachfolgenden Dokument werden die Ziele dieses Projekts definiert, sowie das Gesamtkonzept näher erläutert.

2 Ziele

Die Ziele sind strikt aufgeteilt in die zwei Projekte 5 und 6. Darin enthalten sind die jeweiligen zu erreichenden Muss- und Wunschziele mit ihren quantifizierten Spezifikationen. Diese sind wichtig, da Ortsabhängig unterschiedliche Normwerte gelten und sich dieses Projekt grundsätzlich auf die Schweiz fokussiert.

Tabelle 2.1: Ziele P5

	Ziel	Messbereiche	Genauigkeiten	Einheiten
Mussziele P5				
Sensoren	Lufttemperaturmessung	[-20;60]	± 1	°C
	Windgeschwindigkeitsmessung	[10;25]	± 1	m/s
	Niederschlagsmenge	Wasser	± 100	ml/m ²
Datenspeicherung	Datenabfrage via PuTTY	≥ 9600		Bd/s
RTC	Implementation	Echtzeit	± 1	s/Jahr
Wunschziele P5				
Sensoren	Sonnenstunden Prototyp	Echtzeit		s

Tabelle 2.1 zeigt diverse Ziele im P5, unterteilt in Muss- und Wunschziele. Zu den Musszielen gehören die Lufttemperaturmessung, die Windgeschwindigkeitsmessung, die Niederschlagsmessung, die Implementation des RTC und die mögliche Datenabfrage via Putty vom Datenspeicher. Die Lufttemperatur soll zwischen -20 bis 60 °C ermittelbar sein, mit einer Genauigkeit von ± 1 °C. Die Windgeschwindigkeitsmessung soll vor allem stärkere Windgeschwindigkeiten erfassen, um vor Sturm warnen zu können, weshalb niedrigere Windgeschwindigkeiten vernachlässigt werden können. Die Windgeschwindigkeit soll zwischen 10 und 25 m/s auf ± 1 m/s genau gemessen werden. Die Niederschlagsmenge soll nur für Regenwasser bestimmt werden mit einer Genauigkeit von ± 100 ml/m². Als Wunschziel soll eine Möglichkeit getestet werden um Sonnenstunden zu detektieren, welche dann im P6 umgesetzt wird.

Tabelle 2.2: Ziele P6

	Ziel	Messbereiche	Genauigkeiten	Einheiten
Mussziele P6				
Speisung	Akkukapazität			
	Ladeschaltung Akku			
	Ladeschaltung Photovoltaik			
Kommunikationsmodul	GPS			
	Mobilfunk (SMS)			
Sensoren	Sonnenstunden			
Wunschziele P6				
Kommunikationsmodul	Mobilfunk (Website)			
Speisung	Akku austauschbar			

Tabelle 2.2 zeigt diverse Ziele im P6, unterteilt in Muss- und Wunschziele. Diese Tabelle ist unvollständig und wird im P6 nachgeführt. Generell kann gesagt werden, dass die Speisung, das Kommunikationsmodul mit GPS und Mobilfunk, sowie die Sonnenstunden-Sensorik implementiert werden sollen. Als Wunschziele sind ein austauschbarer Akku und eine Website zur Datensicherung und ggf. grafischen Darstellung aufgeführt.

3 MCU

4 **RTC**

5 Sensoren

5.1 Messen der Lufttemperatur

5.2 Ermittlung der Niederschlagsmenge

Dieses Unterkapitel befasst sich mit der Realisierung der Niederschlagsmessung. Diese soll nach einem Kipplöffelprinzip funktionieren und gemäss definierten Zielen eine Genauigkeit von $\pm 100 \text{ ml/m}^2$ aufweisen. Ausserdem soll als alternative zusätzlich ein Messbecher an der Wetterstation installiert werden, damit der Bauer die Niederschlagsmenge anhand einer Skala ablesen kann. In einem ersten Schritt soll das Kipplöffelprinzip näher erläutert werden. Darauf folgend sollen die Realisierung dieses Kipplöffels und anschliessend die Implementation in der Software thematisiert werden. Zu guter Letzt soll die Validierung des Teilsystems folgen.

Das Kipplöffelprinzip

Das Prinzip des Kipplöffels wird in Abbildung GRAPHKIPP graphisch dargestellt.

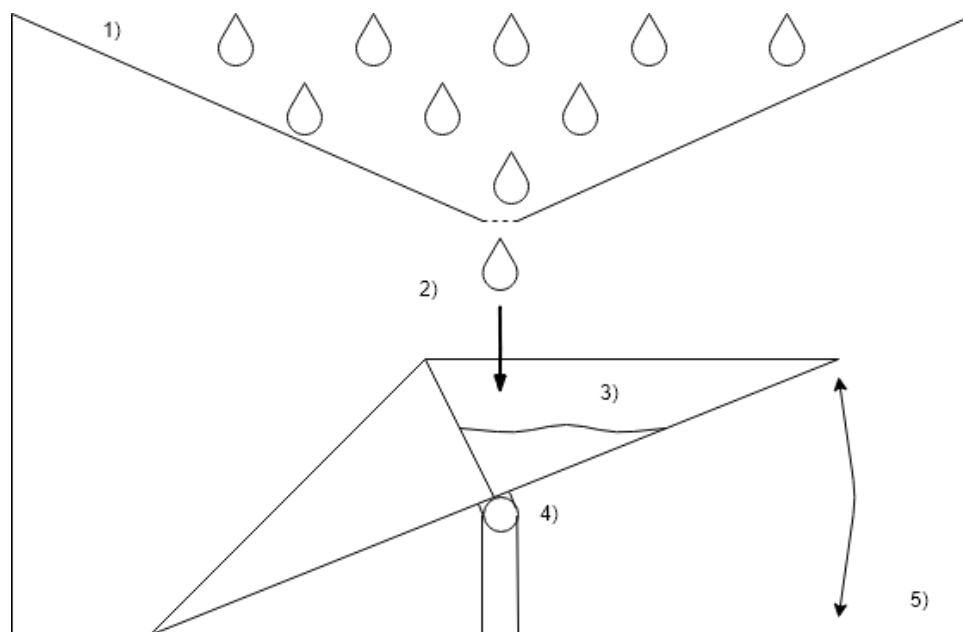


Abbildung 5.1: Darstellung des Kipplöffelprinzips

Abbildung 5.1 zeigt das Prinzip des Kipplöffels. Der Kipplöffel („3“) besteht im Grunde aus zwei Löffeln und ist in der Mitte drehbar mit dem Gehäuse befestigt („4“). Regenwasser wird über eine Öffnung im Gehäusedeckel (Trichter, „1“) zum Kipplöffel befördert („2“). Ist der Löffel mit Regenwasser gefüllt, so kippt dieser aufgrund des Gewichts und leert das Wasser über eine Öffnung im Gehäuseboden („5“) aus. Durch die Kippung wird der andere Löffel in die Ausgangsposition bewegt und kann sich nun mit Wasser füllen. Mit der Hilfe von Reedkontakten und Magneten wird die Anzahl der Kippbewegungen gezählt. Die Niederschlagsmenge ergibt sich aus der Anzahl Kippbewegungen, multipliziert mit dem Volumen des Kipplöffels.

Die Realisierung des Niederschlagsmengensensors

Der Niederschlagsmengensensor wird, wie im Pflichtenheft festgehalten, selbst erstellt. Die Erstellung kann in vier Etappen unterteilt werden. Die erste Etappe ist die Erstellung des Kipp-

löffels. Die zweite Etappe folgt die Erstellung der Drehbaren Lagerung. Als dritte Etappe folgt der Trichter und die vierte und letzte Etappe widmet sich dem Gehäuse.

Die Realisierung des Kipplöffels

Wichtig für die Erstellung des Kipplöffels sind die Dimensionierung und die Materialwahl.

Das Material soll wetterbeständig, einfach bearbeitbar und günstig sein und eine möglichst glatte Oberfläche haben. Die möglichst glatte Oberfläche ist notwendig, damit das Wasser im Kipplöffel sich nicht an der Oberfläche festhält und somit gut abfließt. Acrylglas erfüllt diese Bedingungen und ist in jedem Baumarkt erhältlich, weshalb es als Material gewählt wird.

Die Dimensionierung ist Abhängig von der gewählten Auflösung im Pflichtenheft. Damit eine Auflösung von $\pm 100 \text{ ml/m}^2$ erreicht werden kann, müssen beide Löffel des Kipplöffels bei genau 100 ml Fassungsvermögen kippen. Damit dies erreicht wird, kann man physikalisch die statische Gleichgewichtsbedingung aufstellen und daraus die Dimensionierung folgern. Dies ist jedoch ein sehr aufwändiger, komplizierter und zeitintensiver weg. Einfacher ist es, wenn der Kipplöffel extra zu gross dimensioniert und die Füllmenge im nachhinein justiert wird. Die Justierung erfolgt mittels einer in der Höhe verstellbaren Lagerung, sowie mit in der Höhe verstellbaren Schrauben im Gehäuseboden, welche die Neigung der Endposition des Kipplöffels beeinflusst. Ein weiterer Vorteil dieser Nachjustierung ist, dass auch eine andere Füllmenge einstellbar wäre.

Die Realisierung der drehbaren Lagerung des Kipplöffels

Die Drehbare Lagerung des Kipplöffels ist wichtig, damit der Kipplöffel auf beide Seiten kippen kann. Die Drehachse soll direkt unterhalb der Mitte des Kipplöffels befestigt sein um ein gleichmässiges Kippen zu ermöglichen. Die Höhe des Kipplöffels wird definiert durch die einstellbare Höhe der Drehachsenlagerung.

Die Drehachse wird aus einem Stück Holz und zwei Schrauben gefertigt, wobei das Holz direkt am Kipplöffel befestigt wird. Die zwei Schrauben werden auf einem höhenverstellbarem Gerüst gelagert, so dass ein drehen möglich ist. Dieses Gerüst wird auch aus Holz gefertigt und enthält eine Metallische Fläche an der Kontaktstelle der zuvor erwähnten Schrauben. Ausserdem ist dieses Gerüst höhenverstellbar über zwei mit Muttern feststellbaren Gewinden (für jede Seite eine).

Die Realisierung des Trichters

Der Trichter sorgt dafür, dass der Regen, welcher auf die Trichterfläche fällt, über der Mitte des Kipplöffels in den Löffel fliesst. Die Trichterfläche stellt gleichzeitig die Referenzfläche dar, da die gesamte Regenmenge dieser Fläche über den Kipplöffel erfasst wird. Ist diese Fläche von 1 m^2 abweichend, so muss in der Software ein Skalierungsfaktor implementiert werden, damit die Regenmenge wie gewünscht gemäss Pflichtenheft ermittelt werden kann. Aus diesem Grund ist es einfacher, wenn anstelle eines runden Trichters ein Trichter mit Rechteckiger Fläche realisiert wird, da sonst wegen der Kreiszahl π der Skalierungsfaktor nur angenähert werden kann. Der Trichter wird aus demselben Material gefertigt wie der Kipplöffel, da hier die gleichen Anforderungen gelten.

Die Realisierung des Gehäuses

Das Gehäuse soll den Sensor vor ungewollten äusseren Einflüssen schützen, sowie umgebende Elektronik vor eventuellen Regenwasserspritzer. Ausserdem soll ein Schaltkreis mit Reedrelais implementiert werden, damit die Kippbewegungen von der Elektronik erfasst werden können.

Die Endfertigung des Niederschlagsmengesensors

Implentation in der Software

Validierung der Niederschlagsmessung

5.3 Anemometer

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.



Abbildung 5.2: Anemometer [1]

5.4 Zählung der Sonnenstunden

6 Datenspeicherung

Als Speichermedium wird eine μ SD-Karte verwendet, welche direkt in ein Breakoutboard gesteckt wird. Als Kommunikationsprotokoll wird SPI verwendet.

6.1 Breakoutboard

Das Breakoutboard (siehe Abb. 6.1) kann wegen des intern implementierten *CD74HC4050 high-speed logic level translators*¹ mit 5V betrieben werden. Das Arduino Mega Board und das Breakoutboard werden über SPI (siehe Kapitel 6.1.2) nach dem Master-Slave Kommunikationsprinzip miteinander verbunden.

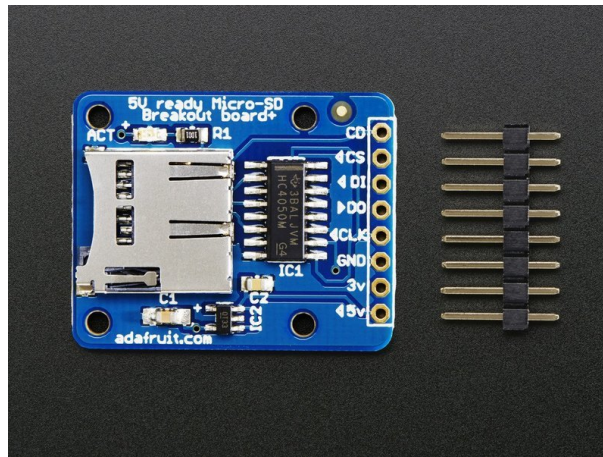


Abbildung 6.1: 254 μ SD-Breakoutboard von Adafruit [2]

6.1.1 Verdrahtung

SD-Karten erfordern viel Datenübertragung. Deshalb kann die beste Leistung erbracht werden, wenn sie an die Hardware-SPI-Pins eines Mikrocontrollers angeschlossen werden. Dabei wird es wie folgt miteinander verbunden: [2]

Hier vielleicht noch das Schema hinzufügen wie es Hardwaremässig implementiert wird.

- **5V** und **GND** Pins jeweils auf die **5V** und **GND** Pins des Arduino Mega Boards
- **CLK** auf die Pinnummer **52**
- **DO** auf die Pinnummer **50**
- **DI** auf die Pinnummer **51**
- **CS** auf die Pinnummer **53**

¹konvertiert eine high-level logik in eine low-level logik

6.1.2 SPI (Serial Peripheral Interface)

Das *Serial Peripheral Interface* ist ein synchrones serielles Datenprotokoll (Datenbus) bestehend aus drei Datenleitungen zur Datenübertragung. Diese sind, wie in Abbildung 6.2 zu sehen, **MISO** (Master In Slave Out), **MOSI** (Master Out Slave In) und **SCLK** (Serial Clock). Auf dem Breakoutboard (Abb. 6.1) sind die Pins mit **DI** (Data In), **DO** (Data Out) und **CLK** (Clock) beschrieben. Zu den Datenleitungen wird noch eine **SS** (Slave Select) oder auch **CS**-Leitung (Chip Select) benötigt. Damit wird vom Master aus den zur momentanen Kommunikation nötigen Slave selektiert. Große Vorteile von SPI sind die Vollduplexfähigkeit und das Taktfrequenzen bis in den MHz-Bereich reichen. [3][4]

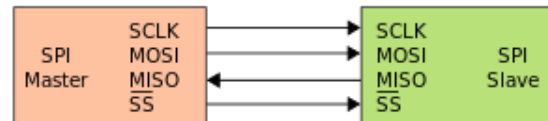


Abbildung 6.2: Einfacher SPI-Datenbus [4]

Möglicherweise noch die Taktfrequenz angeben, resp. die Einstellungen des SPI-Protokolls

6.2 μ SD-Karte



Abbildung 6.3: 16 GB μ SD-Karte [5]

Bei der μ SD-Karte muss auf die Kompatibilität mit dem Breakoutboard geachtet werden. Dafür sind folgende Kriterien zu beachten:

- Die μ SD-Karte muss FAT16 oder FAT 32 formatiert sein.
- Es sind nur die SD und SD High Capacity (SDHC) kompatibel.

Für die Umsetzung dieses Projektes wurde eine μ SD-Karte der SD-Familie SDHC I verwendet (siehe Abb. 6.3). SDHC sind Kapazitäten bis zu 32GB möglich und FAT32 formatiert. [6]

Es könnte noch auf die Anschlüsse der μ SD-Karte eingegangen werden. Gäbe aber nur Sinn wenn ein Print erstellt werden muss wo das Breakoutboard nachkonstruiert wird.

6.3 Implementation in die Firmware

Für die Implementation in die Firmware, um mit dem Breakoutboard über SPI zu kommunizieren und die μ SD-Karte zu beschreiben, resp. zu lesen, wurden direkt die bereits existierenden Librarys `<SPI.h>`² und `<SD.h>` von Arduino inkludiert. In der Arduino IDE können bereits vorgefertigte Example-Codes (siehe Abb. 6.4) zur weiteren Interpretation, wie mit diesen Librarys μ SD-Karten gelesen und geschrieben werden können, verwendet werden.

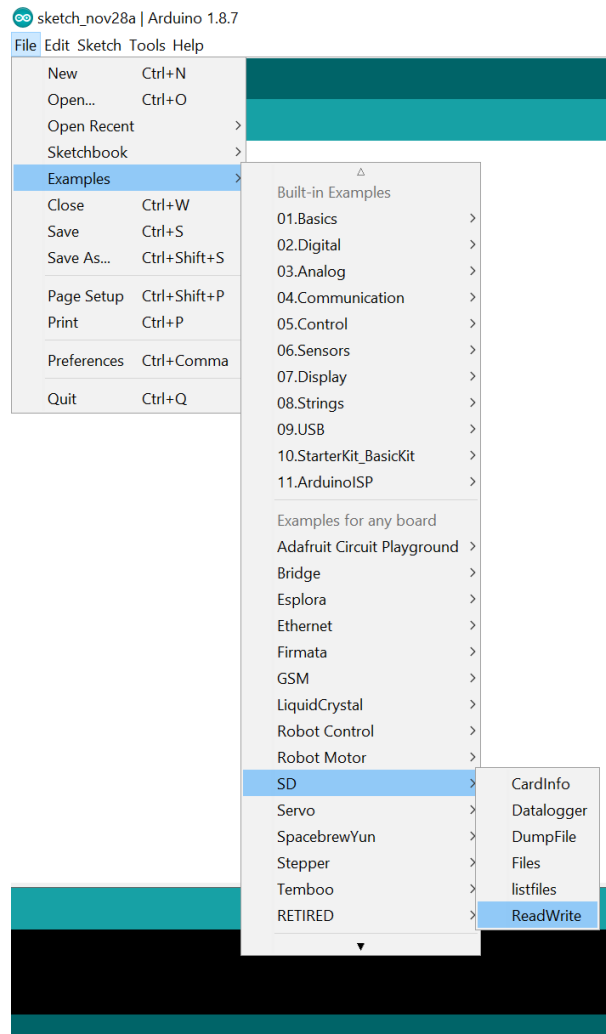


Abbildung 6.4: Example-Codes der Arduino IDE zum Lesen und Schreiben von SD-Karten.

Für eine übersichtlichere Programmstruktur und einfachere Handhabung wurden die Exampel-Codes für die korrekte Implementation angepasst und in Funktionen verpackt. Die Funktionen sind extern im Headerfile "SDCard.h"³ deklariert und im SDCard.cpp initialisiert.

- `void getCardInformations();`
- `void readFileSDCard(String filename);`
- `void writeFileSDCard(double value2save, String filename);`
- `void deleteFileSDCard(String filename);`

² `<*.h>` bezieht sich auf ein include-Verzeichnis unter dem Compiler-Installationsverzeichnis

³ `*.h` bezieht sich relativ auf das aktive Projektverzeichnis

Funktionen könnten noch beschrieben werden.

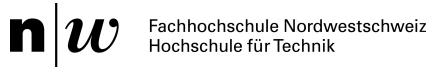
7 Kommunikationsmodul

8 Konzeptvalidierung

9 Literatur

- [1] Amazon. Ersatz Sensor Windgeschwindigkeit für Froggit WH1080 WH3080 WH1090. [Online]. Available: https://www.amazon.de/Ersatz-Sensor-Windgeschwindigkeit-Froggit-WH1080/dp/B00GGM5HEA/ref=pd_rhf_eetyp_p_img_1?_encoding=UTF8&psc=1&refRID=4R5P7RRF8H7RC3A4KAHQ
- [2] Lady Ada, “Micro SD Card Breakout Board Tutorial,” 2018.
- [3] mikrocontroller.net. (unbekannt) Serial peripheral interface. [Online]. Available: https://www.mikrocontroller.net/articles/Serial_Peripheral_Interface
- [4] Wikipedia. (2018) Serial peripheral interface. [Online]. Available: https://de.wikipedia.org/wiki/Serial_Peripheral_Interface
- [5] SanDisk. SanDisk microSDHC Speicherkarte. [Online]. Available: <https://www.sandisk.de/home/memory-cards/microsd-cards/sandisk-microsd>
- [6] ——. SD/SDHC/SDXC Spezifikationen und Kompatibilitäten. [Online]. Available: https://kb-de.sandisk.com/app/answers/detail/a_id/8317/~sd%2Fsdhc%2Fsdxc-spezifikationen-und-kompatibilit%C3%A4ten

A Lastenheft



Ausschreibung Studierendenprojekt P5/P6 Studiengang Elektro- und Informationstechnik

Titel:
Wetterstation mit Solar Energie
Betreuer:
Prof. Dr. Taoufik Nouri (Institut für Mobile und Verteilte Systeme)
Auftraggeber:
Prof. Dr. Taoufik Nouri (Institut für Mobile und Verteilte Systeme)
Aufgabenbeschreibung:
<p>Ausgangslage:</p> <p>Wetterstation sind viele verlangt besonders im Gebiete ohne Strom. Wir schlagen solche Möglichkeit zu realisieren.</p> <p>Zielsetzung:</p> <ol style="list-style-type: none"> 1. Diese Wetterstation misst Regen, Wind- Geschwindigkeit, -Richtung, Temperatur, Sonnenlicht, Feuchtigkeit, Zeit usw. 2. Sie ist dotiert mit verschiedener Kommunikation Module wie GPS, SIM Karte. 3. Sie ist fern abfragbar durch Handy 4. Sie speichert regelmässig die verschiedenen Parameter (Journal). 5. Sie ist komplett automatisiert z.B. Regenwasser wird automatisch ausgeleert. <p>Schlüsselwörter: Energie, Mikrokontroller, Programmierung, Elektronik</p>