

Hochschule München
Fakultät für Elektrotechnik und Informationstechnik

Masterprojekt

Kommunikation Donnerwetter

von Michael Niedermayr

Datum: 21.12.12

Betreuer: Prof. Jürgen Plate

Inhaltsverzeichnis

1	Donnerwetter Kommunikation	1
1.1	Übersicht	1
1.2	Boards	2
1.2.1	Weatherstation Board	2
1.2.2	Sensor Board	2
1.2.3	Temperature Board	3
1.3	Donnerwetter-Client	4
1.3.1	Authentisierung	8
1.3.2	Datenbankzugriff	8
1.3.3	Berechnung der Regenmenge	9
1.3.4	Berechnung der Globalstrahlung	10
1.3.5	Speichern der Daten in eine Datenbank	11
1.3.6	Webseite	12
1.3.7	Display	13
1.4	Fehlerbehandlung	14
1.5	Die Konfigurationsdatei	14
1.6	Board Reset	16
1.7	Debug-Funktionen	16
1.8	Implementierung der Boards	20
	Anhang	23
A.1	Literatur	23
	Abbildungsverzeichnis	25

1 Donnerwetter Kommunikation

1.1 Übersicht

Bei dem Projekt Donnerwetter handelt es sich um eine verteilte Wetterstation, die über Ethernet miteinander kommuniziert und die gewonnenen Wetterdaten in eine Datenbank einträgt. Die aktuellen Werte der Außen- und Innentemperatur, Globalstrahlung, Regenmenge, Luftdruck, Luftfeuchte, Windgeschwindigkeit und Windrichtung werden auf einem Display und der Webseite `donnerwetter.ee.hm.edu` angezeigt.

Die Hardware der Sensorik besteht aus drei Netzwerk-Boards, die die Wetterdaten über TCP/IP bereitstellen. Diese Daten werden von einem Linux betriebenen Atom-PC abgerufen, der sowohl die MySQL-Datenbank, als auch den Webserver für die Webseite bereitstellt. Im Display steckt ein weiteres netzwerkfähiges Board, das die Daten seriell an die Schieberegister der Anzeige weitergibt.

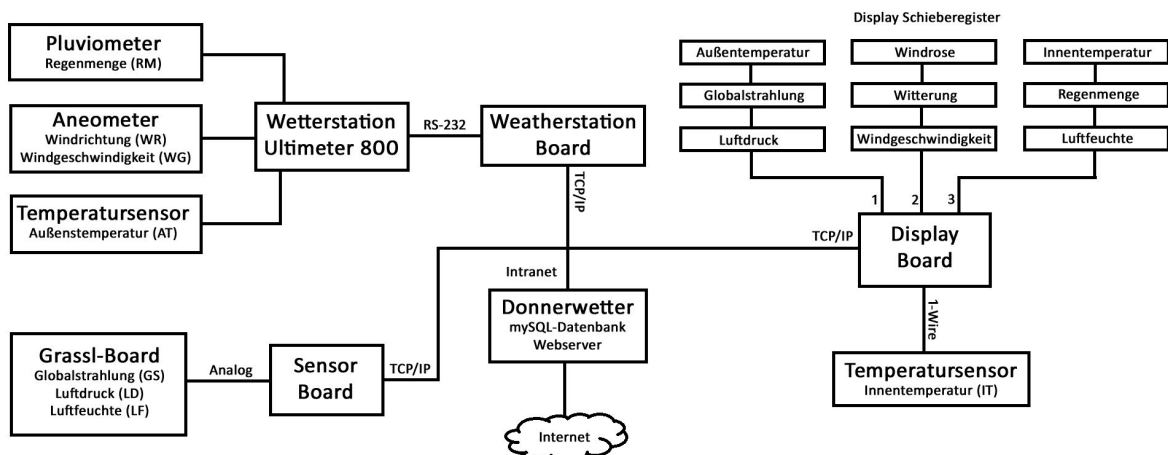


Abbildung 1.1: Schematische Darstellung der Kommunikationswege

1.2 Boards

1.2.1 Weatherstation Board

Auf dem Dach der Hochschule München befindet sich die Wetterstation „ULTIMETER 800“ von der Firma Peet Bros¹. Diese misst die Außentemperatur (AT) und mithilfe eines Pluviometers den Niederschlag (RM). Über ein Anemometer wird außerdem die Windgeschwindigkeit (WG) und die Windrichtung (WR) bestimmt. Die Daten werden über die serielle Schnittstelle RS-232 vom Daten-Board „Weatherstation“ ausgelesen, wobei bei der Windgeschwindigkeit ein Mittelwert über die sekundlichen Messungen der letzten Minute gebildet wird.



Abbildung 1.2: Pluviometer, Anemometer und Globalstrahlungsmesser



Abbildung 1.3: Ulimeter 800 und Weatherstation Board

¹<http://www.peetbros.com>

1.2.2 Sensor Board

Das „Graßl-Board“ misst die Werte Globalstrahlung (GS), Luftdruck (LD) und relative Luftfeuchte (LF) und bereitet diese auf, so dass das Daten-Board „Sensor“ die analogen Werte mithilfe eines 10-Bit-ADC in digitale Werte umwandeln kann. Bei der Globalstrahlung wird ebenfalls ein Mittelwert über die letzte Minute gebildet.

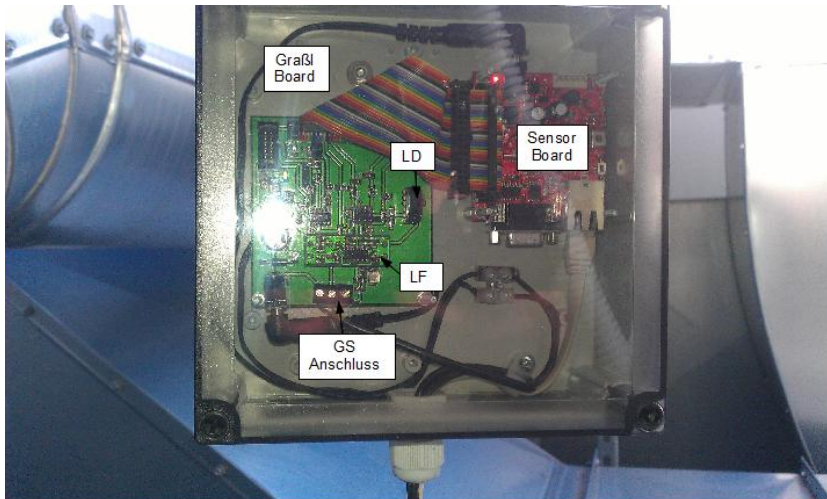


Abbildung 1.4: Graßl-Board und Sensor-Board

1.2.3 Temperature Board

Die Innentemperatur (IT) wird von dem Sensor „DS18S20“ der Firma Maxim² gemessen und über das 1-Wire Protokoll vom Daten-Board „Temperature“ ausgelesen. Das „Temperature“-Board ist zugleich das „Display“-Board und befindet sich zusammen mit dem Temperatursensor im Display.

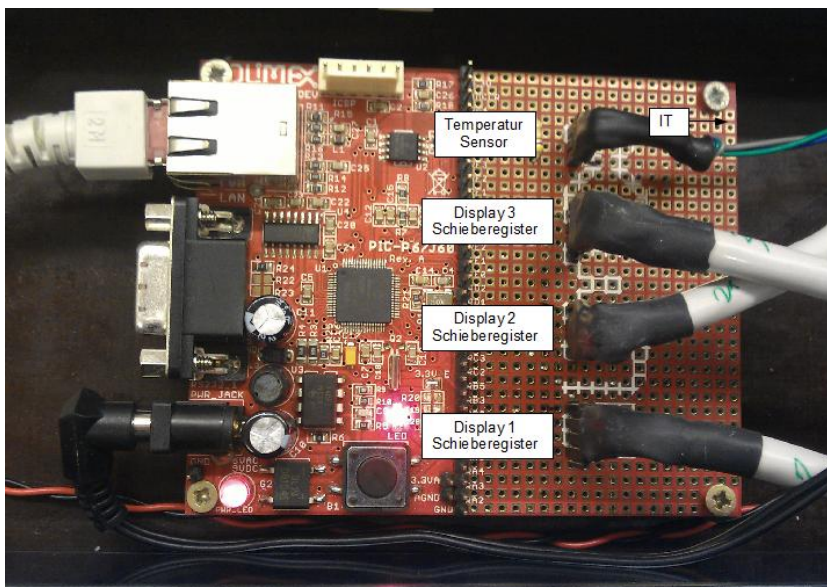


Abbildung 1.5: Display/Temperatur-Board

²<http://www.maximintegrated.com>

1.3 Donnerwetter-Client

Auf dem Atom-PC wird jede Minute ein Cronjob gestartet, der das Skript „donnerwetterskript“ ausführt, welches überprüft ob noch Prozesse „donnerwetter“ laufen und diese gegeben falls beendet. Dieser Sicherheitsmechanismus sorgt dafür, dass sich hängengebliebene Instanzen des Programms niemals anhäufen und sich gegenseitig blockieren können. Anschließend wird das Programm „donnerwetter“ gestartet. Alle fünf Minuten wird über den Cronjob das Programm zudem mit dem zusätzlichen Kommandozeilenparameter „-save“ gestartet, wodurch die Wetterdaten in die Datenbank gespeichert werden. Alle Ausgaben werden unterdrückt und Fehlermeldungen werden mit Zeitstempel an die Datei „error.log“ angehängt. Der Cronjob kann über den Linux Befehl „crontab -e“ bearbeitet werden.

Eintrag im Cornjob:

```
# m h dom mon dow    command

PATH=/home/mike/Donnerwetter/DonnerwetterClient

# Alle 5 Minuten werden die Wetterdaten in die Datenkank gespeichert
*/5 * * * * $PATH/donnerwetterskript -save >/dev/null 2>/dev/null

# Jede Minute wird das Display und die Webseite aktualisiert
1-4,6-9,11-14,16-19,21-24,26-29,31-34,36-39,41-44,46-49,51-54,56-59 * * * *
$PATH/donnerwetterskript >/dev/null 2>/dev/null

# Einmal am Tag wird das Display zurueckgesetzt
1 4 * * * $PATH/donnerwetterskript -reset >/dev/null 2>/dev/null
```

Inhalt von donnerwetterskript:

```
#!/bin/sh

PATH=/home/mike/Donnerwetter/DonnerwetterClient

if [ $(ps | grep -c 'donnerwetter') -gt 1 ]
then

    PID=$(ps ax | grep 'donnerwetter' | awk '{print $1;}')
    kill -HUP $PID || kill -9 $PID

fi

$PATH/donnerwetter $PATH/config.ini -noprint $1 >/dev/null 2>>$PATH/error.log
```

Wenn der Donnerwetter-Client mit dem Befehl „./donnerwetter“ gestartet wird, dann wird zuerst die Datei „config.ini“ gesucht, deren Pfad als Kommandozeilenparameter übergeben werden kann. Ansonsten wird im aktuellen Verzeichnis nach der Datei gesucht. In der „config.ini“ stehen alle relevanten Einstellungsmöglichkeiten wie zum Beispiel die IP-Adressen der Boards und deren Passwörter. Danach versucht das Programm nacheinander zu den drei Daten-Boards eine Verbindung über TCP/IP aufzubauen.

Beispiel der Ausgabe des Donnerwetter Clients:

```
./donnerwetter -save
```

```
##### DONNERWETTER #####
```

```
Loaded from 'config.ini':
```

```
Use Board 1      : TRUE
Use Board 2      : TRUE
Use Board 3      : TRUE
Use Display      : TRUE
Use Website SQL  : TRUE
Use Data SQL     : TRUE
Save Data SQL    : TRUE

Use Debug Values : FALSE

Max Board Failure : 4
Failure Wait Time : 100 ms
Connect Wait Time : 3 s

Board 1 Name      : Weatherstation
Board 1 IP        : 10.27.210.110
Board 1 Port      : 1011
Board 1 Password  : df5FjghdjGE584hg5jkE

Board 2 Name      : Sensor
Board 2 IP        : 10.27.210.111
Board 2 Port      : 1022
Board 2 Password  : ksJ3ksEks5jWkf4jGske

Board 3 Name      : Temperature
Board 3 IP        : 10.27.210.112
Board 3 Port      : 1033
Board 3 Password  : s4qFGqw8SqeFvgd4vrE

Display Name      : Display
Display IP        : 10.27.210.112
Display Port      : 1044
Display Password  : isdrS7Gi4jrF4gjdSi6

SQL Hostname      : 127.0.0.1
SQL Port          : 3306
SQL User          : donnerwetter
SQL Database      : donnerwetter
SQL DataTable     : data
SQL Website Table : website
SQL Lasr Value Table : lastvalue
SQL Password      : vA1SVuR5saG7orfgDg

DP Sun Full      : 300.0 W/m2
DP Sun Clouds    : 200.0 W/m2
DP Sun Rain      : 250.0 W/m2

Temperatur       Min:   -99.9 Max:    99.9
Windgeschwindigkeit Min:    0.0 Max:  999.9
Windrichtung     Min:    0.0 Max:  359.0
```


1.3.1 Authentisierung

Aus Sicherheits- und Integritätgründen findet bei der Kommunikation zwischen den Boards und dem Donnerwetter-Client immer zuerst ein 3-Wege-Handshake statt, bei dem das Daten-Board eine Challenge stellt, sobald es eine Verbindungsanfrage über TCP auf dem Datenport empfängt, an dem es lauscht. Zuerst werden „AUTH“ und zwei zufällige Zahlen (Z1, Z2) zurücksendet. Der Donnerwetter-Client sendet daraufhin den MD5-Hashwert über die beiden Zufallszahlen und dem Board-Passwort zurück. Weiterhin wird eine Zufallszahl (Z3) als eigene Challenge gesendet. Das Daten-Board berechnet ebenfalls den selben MD5-Hashwert und vergleicht ihn mit dem empfangenen Hashwert. Wenn die beiden Hashwerte übereinstimmen, scheinen beide über das selbe Passwort zu verfügen, und der Donnerwetter-Client hat sich somit gegenüber dem Daten-Board authentisiert. Bei Misserfolg bricht das Daten-Board die Kommunikation ab, andernfalls errechnet es über die erste eigene Zufallszahl und die empfangene Zufallszahl und das Passwort ebenfalls einen MD5-Hashwert und sendet diesen an den Client. So wird auch das Daten-Board gegenüber dem Donnerwetter-Client authentisiert.

3-Wege-Handshake:

```
Client <-- Board: AUTH + Z1 + Z2
Client --> Board: MD5(Z1, Z2, Passwort) + Z3
Client <-- Board: MD5(Z1, Z3, Passwort)
```

Anschließend werden die Wetterdaten unverschlüsselt im Klartext übertragen, da diese nicht geheim sind. Allerdings sollte erkannt werden können, ob die Daten bei der Übertragung oder durch einen Angriff verändert wurden. Deshalb wird anschließend ein MD5-Hash über die Wetterdaten und dem Passwort gesendet. Bei Erfolg schickt der Donnerwetter-Client „ACK“ zurück und beide Seiten beenden ihre Verbindung.

Beispiel der Kommunikation:

```
Client <-- Board: AUTH 1195024151 47196
Client --> Board: 8369962d18e2b929b1442356f05b2e5c 259186158
Client <-- Board: fc51131b703282f694f983144b13d499
Client <-- Board: WG: 1.7 WR: 221 AT: 2.6 RM: 0.2
Client <-- Board: c27db465babaa5d1cb8066656faa8869
Client --> Board: ACK
```

1.3.2 Datenbankzugriff

Nachdem die drei Datenboards abgefragt wurden, wird eine Verbindung zur MySQL Datenbank aufgebaut, wobei die Login-Daten aus der Konfigurationsdatei verwendet werden. Bei Erfolg wird die Datenbank „donnerwetter“ geöffnet. Falls dies nicht existiert, dann wird dies mit dem folgendem MySQL-Befehl angelegt und genutzt.

```
CREATE DATABASE IF NOT EXISTS donnerwetter;
USE donnerwetter;
```

Für die folgenden Berechnungen und Zugriffe wird die Datenbank offengehalten und erst nach Abarbeitung des Programms geschlossen.

1.3.3 Berechnung der Regenmenge

Die Wetterstation „ULTIMETER 800“ addiert jeden neuen Niederschlag auf und setzt diesen um Mitternacht auf null um die tägliche Niederschlagsmenge anzuzeigen. Da nicht garantiert werden kann, dass die Uhrzeit der Wetterstation immer synchron zum Atom-PC läuft, wird immer nur die Änderung zum vorherigen Wert in der Datenbank gespeichert. Dazu wird der letzte Wert „RM“ aus der MySQL Tabelle „donnerwetter/lastvalue“ geholt und die Differenz zum neuen Wert in „RM“ abgelegt.

Um den Wert aus der Datenbank zu lesen wird folgender Befehl verwendet:

```
SELECT RM_LAST FROM lastvalue LIMIT 1;
```

Der alte Wert „RM_LAST“ wird mit dem neuen Wert überschrieben, wodurch sich immer nur ein einziger Wert in der Tabelle befindet.

```
REPLACE lastvalue (RM_LAST) VALUES ( 0.2);
```

So ist es möglich die Regenmenge aus der Datenbank über einen Tag, eine Woche oder einen Monat aufzuaddieren. Wenn die Tabelle „lastvalue“ nicht existiert, wird diese vorher mit folgendem Befehl angelegt:

```
CREATE TABLE IF NOT EXISTS lastvalue
(RM_LAST FLOAT, Id INT NOT NULL, TX timestamp, primary key (Id));
```

Datenformat der MySQL-Tabelle „donnerwetter/lastvalue“:

```
+-----+-----+-----+
| RM_LAST | Id | TX |
+-----+-----+-----+
| 5.2 | 0 | 2012-12-13 19:29:01 |
+-----+-----+-----+
```

1.3.4 Berechnung der Globalstrahlung

Da der Sensor für die Globalstrahlung temperaturabhängig ist, wird der korrekte Wert mithilfe der gemessenen Außentemperatur nach folgendem Algorithmus errechnet. Die Formel wurde anhand des spezieifischen Datenblatts des Globalstrahlungssensors entwickelt³ und mithilfe einer Messreihe einer Referenz-Globalstrahlung angepasst. Dabei handelte es sich um eine Globalstrahlungsmessung die ca 500 Meter vom Standort der Wetterstation entfernt installiert ist und somit annähernd ähnlichen Sonneneinstrahlungen und Umweltbedinungen ausgesetzt ist.

```
if (AT < 0.0)
  Offset = -184.0
else if (0.0 <= AT < 10.0)
  Offset = AT * 6.0 - 182.0
else if (10.0 <= AT < 25.0)
  Offset = AT * 8.1333 - 203.3333
else
  Offset = AT * 8.96 - 224.0

GS = GS_RAW - Offset

if (GS < 1705.0)
  GS = GS * 0.3135 - 59.565
else
  GS = GS * 0.3879 - 186.3695

if (GS < 50.0)
  GS = GS * 0.4
else if (GS < 300.0)
  GS = GS * 0.3
else
  GS = GS * 0.25
```

³siehe Quelle Stefan Graßl- Sensorplatine

1.3.5 Speichern der Daten in eine Datenbank

Wenn der Donnerwetter-Client mit dem Kommandozeilenparameter „-save“ aufgerufen wird, dann werden die Daten in die Datenbank unter „donnerwetter/data“ gespeichert. Ohne den Kommandozeilenparameter werden nur das Display und die Webseite aktualisiert.

```
INSERT INTO data (IT,AT,WG,WR,RM,LD,LF,GS) VALUES
( 23.8, 2.6, 1.7, 221.0, 0.0, 939.0, 64.0, 105.3);
```

Falls die Tabelle nicht vorhanden ist, wird diese vorher mit folgendem Befehl angelegt:

```
CREATE TABLE IF NOT EXISTS data (IT FLOAT, AT FLOAT, WG FLOAT, WR FLOAT,
RM FLOAT, LD FLOAT, LF FLOAT, GS FLOAT, Id int NOT NULL AUTO_INCREMENT,
TX timestamp, primary key (Id));
```

Da alle fünf Minuten die aktuellen Werte in die Datenbank gespeichert werden, füllt sich diese kontinuierlich mit Messwerten an. Über den Timestamp und die Id können diese Daten zur Auswertung oder für Erstellung von Graphen verwendet werden.

Datenformat der MySQL-Tabelle „donnerwetter/data“:

IT	AT	WG	WR	RM	LD	LF	GS	Id	TX
22.1	-3.3	0	86	0	948	62	100	12116	15:00:01
22.3	-4.7	0.6	138	0.2	948	60	102	12117	15:05:01
22	-4.7	0.2	103	0	947	60	110	12118	15:10:02
22.1	-4.7	0.5	121	0.1	948	62	92	12119	15:15:02
22.1	-5.2	0.4	96	0	947	61	102	12120	15:20:02

1.3.6 Webseite

Unabhängig davon, ob die Wetterdaten gespeichert wurden, werden immer die aktuellen Daten in der Tabelle „donnerwetter/webseite“ abgelegt. Dabei werden die alten Daten ersetzt, so dass sich immer nur ein aktueller Wert pro Wetterdatensatz in der Tabelle befindet. Beim Aufruf der Webseite holt sich diese über eine einzige MySQL-Abfrage diesen Datensatz und zeigt ihn an. Die notwendigen Berechnungen wurden dann bereits erledigt.

Um die maximal und minimal Werte der Außen- und Innentemperatur des Tages für die Webseite zu ermitteln, werden folgende Anfragen an die Datenbank gestellt. Dabei wird ab Mitternacht gerechnet.

```
SELECT MIN(AT) FROM data WHERE TX >= CURDATE();
SELECT MAX(AT) FROM data WHERE TX >= CURDATE();
SELECT MIN(IT) FROM data WHERE TX >= CURDATE();
SELECT MAX(IT) FROM data WHERE TX >= CURDATE();
```

Für die Regenmenge pro Tag, pro Woche und pro Monat werden folgende Abfragen verwendet:

```
SELECT SUM(RM) FROM data WHERE TX >= CURDATE();
SELECT SUM(RM) FROM data WHERE TX >=
DATE_SUB(CURDATE(), INTERVAL WEEKDAY(CURDATE()) DAY);
SELECT SUM(RM) FROM data WHERE TX >=
DATE_ADD(LAST_DAY(DATE_SUB(CURDATE(), INTERVAL 1 MONTH)), INTERVAL 1 DAY);
```

Bei der Regenmenge pro Monat wird ab dem erster Tag des Monats gerechnet. Zu beachten ist, das es vorkommen kann, dass es mehr Niederschlag die Woche gegeben hat, als es Niederschlag pro Monat gab. Dies tritt auf, wenn der Monatswechsel unter der Woche passiert.

Nachdem alle erforderlichen Daten erhoben wurden, wird der Datensatz in der Tabelle „donnerwetter/webseite“ aktualisiert:

```
REPLACE website (ATEMP,CHILL,ATMIN,ATMAX,GLOBS,DRUCK,WDGES,ITEMP,ITMIN,ITMAX,
REGHE,REGWO,REGMO,FEUCH,WINDR,DISPL) VALUES ( 2.6, 0.9, -0.4, 3.1,
105.3, 939.0, 1.7, 23.8, 22.3, 24.4, 0.0, 0.0, 7.2, 64.0, 221.0, 32.0);
```

Falls die Tabelle nicht vorhanden ist, wird diese vorher mit folgendem Befehl angelegt:

```
CREATE TABLE IF NOT EXISTS website (ATEMP FLOAT, CHILL FLOAT, ATMIN FLOAT,
ATMAX FLOAT, GLOBS FLOAT, DRUCK FLOAT, WDGES FLOAT, ITEMPT FLOAT, ITMIN FLOAT,
ITMAX FLOAT, REGHE FLOAT, REGWO FLOAT, REGMO FLOAT, FEUCH FLOAT, WINDR FLOAT,
DISPL FLOAT, Id INT NOT NULL, TX timestamp, primary key (Id));
```

Anschließend kann die Datenbank geschlossen werden.

Datenformat der MySQL-Tabelle „donnerwetter/webseite“:

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ATEMP | CHILL | ATMIN | ATMAX | GLOBS | DRUCK | WDGES | ITEMP | ITMIN |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  0.1  |  0.1  | -5.6  |  4.3  |    0   |  942  |    0   | 23.3  |  22   |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ITMAX | REGHE | REGWO | REGMO | FEUCH | WINDR | DISPL | Id | TX
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 26.6  |    0  |    0  |  7.2  |   62  |  303  |    0   | 0 | 2012-12-13 19:24:02 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

1.3.7 Display

Um die Wetterdaten auf dem Display anzuzeigen, wird zum Display-Board eine Verbindung über TCP/IP aufgenommen. Die Authentisierung über den 3-Wege-Handshake ist die Gleiche wie bei den Daten-Boards. Einziger Unterschied ist, dass danach der Donnerwetter-Client die Wetterdaten an das Board schickt und nicht umgekehrt. Aus Integritätsgründen wird daran ebenfalls ein MD5-Hash über die Wetterdatenzeichenkette und das Passwort angehängt. Das Display-Board überprüft die eingegangenen Daten sowohl auf Richtigkeit als auch auf Plausibilität und testet außerdem, ob die Werte den anzeigbaren Zahlen-Bereich des Displays überschreiten. Wenn alles korrekt ist, gibt das Board die Daten auf dem Display aus und sendet „ACK“ zurück bevor die Verbindung beendet wird.

Die Kommunikation mit dem Display:

```

Client <-- Board: AUTH 24676717 36217
Client --> Board: 4d3e0735007d29b0be0494e07c4f893c 1495945037
Client <-- Board: daa1424322678260b066e9de53f67edb
Client --> Board: WR: 221 LF: 64 RM: 0.0 IT: 23.8 AT: 2.6 GS: 105
                LD: 939 WG: 1.7 DP: 32 0 0f7acbd7994256ad91dee92efca8525
Client <-- Board: ACK

```

Der String muss dabei eine genaue Ordnung einhalten, wobei die Anordnung, die Dezimalpunkte und die Leerzeichen zu beachten sind. Das folgende Beispiel gibt die jeweils gültigen minimal und maximal Werte an.

```

WR:  0  LF:  0  RM:  0.0 IT:-99.9 AT:-99.9 GS:  0  LD:  0  WG:  0.0 DP:  0  0
WR: 359 LF: 100 RM:999.9 IT: 99.9 AT: 99.9 GS:1500 LD:1150 WG:999.9 DP:255255

```

Der letzte Parameter „DP“ gibt die Anweisung an die Witterungsdarstellung weiter. Dieses ist ein 16-Bit Schieberegister, wobei die beiden Bytes in dezimaler Schreibweise im Bereich 0 bis 255 übergeben werden.

1.4 Fehlerbehandlung

Wenn eine Verbindungsanfrage mit einem Board unbeantwortet bleibt, dann wird die Zeit „Connect Wait Time“ in Sekunden gewartet, bis der Versuch abgebrochen wird. Diese Einstellungen können in der „config.ini“ angepasst werden. Danach wird die „Failure Wait Time“ in Millisekunden abgewartet und ein neuer Verbindungsversuch wird gestartet. Wenn die maximale Anzahl von Versuchen „Max Board Failure“ aufgebraucht ist, dann kann das Board endgültig nicht erreicht werden und der Programmablauf wird fortgesetzt.

Alle Wetterdaten, die nicht abgerufen werden konnten, gehen als „NULL“ Wert in die Datenbank ein, um anzuzeigen, dass kein gültiger Wert vorhanden ist. Der gleiche Mechanismus wird verwendet, wenn andere Fehler auftreten. Zum Beispiel wenn das Board bei einem oder mehreren Wetterdaten „NIL“ zurückgibt, da kein gültiger Wert vorliegt, oder wenn die Werte außerhalb der zulässigen Grenzen liegen. Alle erzeugten Fehlermeldungen werden über das vom Cronjob aufgerufene „donnerwetterskript“ an die Datei „error.log“ zusammen mit einem Zeitstempel angehängt.

1.5 Die Konfigurationsdatei

Aus der Datei „config.ini“ werden alle Konfigurationen und Einstellungen bei jedem Aufruf des Donnerwetter-Clients ausgelesen. Da die Datei auch alle Passwörter enthält, sollte nur der Besitzer Schreib- und Leserecht für diese Datei haben. Leerzeilen und Kommentarzeilen, die mit „#“ anfangen, werden nicht ausgewertet. Die Zuweisung der Werte muss in derselben Zeile erfolgen und durch mindestens ein Leer- oder Tabulatorzeichen vom Bezeichner abgetrennt sein. Einzelne Kommandos können durch ein „#“-Zeichen am Zeilenanfang auskommentiert werden.

Beispiel der Konfigurationsdatei „config.ini“:

```
# Welche Module werden verwendet
Use_Board1
Use_Board2
Use_Board3
Use_Display
Use_WebsiteSQL
Use_DataSQL

# Wie -save als Kommandozeilenparameter um das Speichern dauerhaft einzuschalten
#Save_DataSQL

# Allgemeine Einstellungen
Max_Board_Failure 4
Failure_Wait_Time 100
Connect_Wait_Time 3

# Die Konfigurationen der Boards

# WG, WR, AT, RM
Board1_Name Weatherstation
Board1_IP 10.27.210.110
Board1_Port 1011
Board1_Password df5FjghdjGE584hg5jkE

# GS, LF, LD
Board2_Name Sensor
Board2_IP 10.27.210.111
Board2_Port 1022
```



```
Board2_Password ks3ksEks5jWkf4jGske

# IT
Board3_Name Temperature
Board3_IP 10.27.210.112
Board3_Port 1033
Board3_Password s4qFGqw8SqeFvgd4vrE

# WR,LF,RM,IT,AT,GS,LD,WG,DP
Display_Name Display
Display_IP 10.27.210.112
Display_Port 1044
Display_Password isdrS7Gi4jrF4gjdSi6

# Zugangsdaten fuer die MySQL Datenbank
SQL_Hostname 127.0.0.1
SQL_Port 3306
SQL_User donnerwetter
SQL_Database donnerwetter
SQL_DataTable data
SQL_WebsiteTable website
SQL_LastValueTable lastvalue
SQL_Password vAlSVuR5saG7orfgDg

# Max und Min Werte
Min_Temperatur -99.9
Max_Temperatur 99.9

Min_Windgeschwindigkeit 0.0
Max_Windgeschwindigkeit 999.9

Min_Windrichtung 0
Max_Windrichtung 359

Min_Regenmenge 0.0
Max_Regenmenge 999.9

Min_Luftdruck 0
Max_Luftdruck 1150

Min_Luftfeuchte 0
Max_Luftfeuchte 100

Min_Globalstrahlung 0
Max_Globalstrahlung 1500

# Grenzwerte Globalstrahlung des Displays
DP_Sun_Full 300
DP_Sun_Clouds 200
DP_Sun_Rain 250

# Debug Values aktiviert mit -debugvalues als Kommandozeilenparameter
Set_IT 10.0
Set_AT 10.0
Set_WG 0.4
```

```
Set_WR 90
Set_RM 10.0
Set_LD 10.0
Set_LF 10.0
Set_GS 10.0
Set_DP 64

#DP_OFF = 0
#DP_SUN_FULL = 128
#DP_SUN_CLOUDS = 64
#DP_CLOUDS = 32
#DP_CLOUD_RAIN = 48
#DP_CLOUD_SNOW = 40
#DP_SNOW_RAIN = 56
#DP_SUN_RAIN = 176
```

1.6 Board Reset

Bei Aufruf des Donnerwetter-Clients mit dem Übergabeparameter „-reset“ wird nacheinander eine Verbindung zu allen Boards aufgebaut und nach der Authentisierung durch den 3-Wege-Handshake anstelle des „ACK“ ein „RES“ gesendet. Beim Display-Board wird das „RES“ anstelle der anzuzeigenden Wetterdaten gesendet. Dadurch werden die Boards angewiesen, sich neu zu starten. Wenn das Display-Board neu gestartet wird, dann initialisiert es die Schieberegister mit einem Muster, so dass alle Elemente des Display aufleuchten, bis das Display-Board gültige Werte empfängt. Die Boards verfügen außerdem über einen Watchdog-Timer, der immer dann zurückgesetzt wird, wenn erfolgreich Daten von dem Board abgerufen wurden, beziehungsweise wenn Daten erfolgreich an das Display geschickt wurden. Ansonsten läuft der Timer nach ca 2:30 Min ab und veranlasst einen Reset des Boards. Da im Normalfall die Boards jede Minute abgefragt werden, tritt dieser Fall nur dann ein, wenn ein Board nicht zu erreichen ist. Daraus folgt auch, dass das Display nach 2:30 Min den Initialzustand anzeigt, wenn das Display vom Netzwerk getrennt wurde.

1.7 Debug-Funktionen

Wird der Donnerwetter-Client mit dem Übergabeparameter „-noprnt“ aufgerufen, dann wird die gesamte Ausgabe unterdrückt und nur die Fehlermeldungen werden angezeigt. Andernfalls wird die gesamte Kommunikation mit den Daten-Boards und der Datenbank angezeigt. Am Ende wird zudem eine Zusammenfassung der Wetterdaten ausgegeben.

In der erste Sektion der Konfigurationsdatei „config.ini“ kann definiert werden, welche Module bei der Ausführung des Donnerwetter-Clients abgearbeitet werden sollen. Wenn beispielsweise nur die Daten aktualisiert und auf dem Display ausgegeben werden sollen, ohne das eine Verbindung zur Datenbank aufgebaut werden soll, so sind die letzten beiden Einträge auszukommentieren.

Wenn der Donnerwetter-Client mit dem Übergabeparameter „-debugvalues“ aufgerufen wird, dann werden die Debug-Werte wie z.B „Set_IT“ aus der Konfigurationsdatei anstelle der echten Wetterdaten verwendet. Der Parameter von „Set_DP“ definiert das verwendete Byte für die

Witterungsdarstellung des Displays. Jedes gesetzte Bit des Bytes bringt ein Element wie zum Beispiel die Regentropfen zum Leuchten.

Gültige Werte für die Witterungsdarstellung:

```
#DP_OFF           = 0
#DP_SUN_FULL     = 128
#DP_SUN_CLOUDS   = 64
#DP_CLOUDS       = 32
#DP_CLOUD_RAIN   = 48
#DP_CLOUD_SNOW   = 40
#DP_SNOW_RAIN    = 56
#DP_SUN_RAIN     = 176
```

Die Daten-Boards und das Display-Board verfügen ebenfalls über einige Debug-Funktionen. Diese sind aber nur zu erreichen, wenn das jeweilige Board mitsamt den Debug-Funktionen geflasht wurde. Im Normalbetrieb sind alle Debug-Funktionen nicht vorhanden. Um diese zu aktivieren muss in der Datei „DonnerwetterFirmware/Config/ProjectConfig.h“ das define „ENABLE_DEBUG_MODE“ einkommentiert sein. Danach muss das gesamte Projekt neu kompiliert werden und in das jeweilige Board geflasht werden. Dabei gilt es zu beachten, dass auch nur das für das jeweilige Board zutreffende Define wie zum Beispiel „BOARD_WEATHER“ für das Board Weatherstation einkommentiert ist. Andernfalls kann es vorkommen, dass mehrere Boards dieselbe IP- und MAC-Adresse im Netzwerk haben. Nähere Hinweise zum flashen der Firmware finden Sie im Kapitel „Implementierung“.⁴ Wenn das Board korrekt geflasht wurden, erlaubt es einen Terminalzugriff über die serielle Schnittstelle. Schließen Sie dazu am Board ein serielles Kabel am RS-232 Anschluß an und verbinden Sie dieses mit einem PC oder Laptop. Als Terminalsoftware eignet sich zum Beispiel „Putty“⁴. Wählen Sie den verwendeten Com-Port aus und geben Sie die Baudrate mit 19200 an. Nun sollten Befehle über die Tastatur möglich sein. Die Eingabe von „help“ zeigt die unterstützten Befehle an und der Befehl „status“ listet die gegenwärtigen Einstellungen auf.

Beispiel der Debugfunktionen des Display-Boards:

```
help
```

```
----- Commands -----
status           - shows the status
enable X         - to enable Debug Module Number X
disable X        - to disable Debug Module Number X
show data        - to show the current sensor and weather data
set input        - to enter a string for the display
display data     - send one random data string to the display
refresh display  - to refresh the display with current string
debug default    - resets the debug settings
reset            - to reset the board
help            - shows this screen
```

```
status
```

```
IP Address:   10. 27.210.112
Subnet Mask: 255.255.  0.  0
Gateway:     10. 27. 64.  1
MAC:         00-04-E2-00-E9-EF
```

```
Display enabled      Port 1044
Temperature enabled  Port 1033
```

⁴<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

Aktive Debug Modules:

Nr	Name	ON=1, OFF=0
0	MD5_CHECK_ALWAYS_FALSE	0
1	MD5_CHECK_ALWAYS_TRUE	0
2	SENSOR_SENDS_WRONG_DATA_HASH	0
3	WEATHER_SENDS_WRONG_DATA_HASH	0
4	DISPLAY_SENDS_WRONG_DATA_HASH	0
5	SEND_RECV_INFO_TO_UART	1
6	DISPLAY_DATA_TO_UART	1
7	CREATE_RANDOM_SENSOR_DATA	0
8	CREATE_RANDOM_WEATHER_DATA	0
9	CREATE_RANDOM_TEMPERATURE_DATA	0
10	USE_UART_FOR_DEBUG	1
11	TCP_RETURN_WEATHER_DATA	0

Mit den Befehlen „enable X“ und „disable X“ können die einzelnen Debug-Funktionen ein und ausgeschaltet werden, wobei „X“ für die jeweilige Nummer der Funktion steht. Mit der aktivierten Funktion „SEND_RECV_INFO_TO_UART“ wird die gesamte Datenkommunikation aus der Sicht des Boards in Echtzeit ausgegeben. Dabei steht „WST“ für Weatherstation-Board, „SEN“ für das Sensor-Boards, „TMP“ für das Innentemperatur-Board und „DSP“ für das Display:

```
WST: Send: AUTH 1195024151 47196
WST: Recv: 8369962d18e2b929b1442356f05b2e5c 259186158
WST: Send: fc51131b703282f694f983144b13d499
WST: Send: WG: 1.7 WR: 221 AT: 2.6 RM: 0.2
WST: Send: c27db465babaa5d1cb8066656faa8869
WST: Recv: ACK
WST: CLOSE SOCKET
```

```
SEN: Send: AUTH 1195014894 29855
SEN: Recv: 9b53626d86ac896413ea848c60792b00 1819989205
SEN: Send: 2076fb76f0238e51296f24a312daa4c4
SEN: Send: GS: 282 LF: 64 LD: 939
SEN: Send: d202a9efd181b02f4daa9b1a6c87b178
SEN: Recv: ACK
SEN: CLOSE SOCKET
```

```
TMP: Send: AUTH 193133 7967
TMP: Recv: a75b6a9163ae03727eca55b5666408e8 881347503
TMP: Send: be304c402ea0dd1accb0e0ec15782854
TMP: Send: IT: 24.4
TMP: Send: 38b22e587838cfa68d29ffc6378b013e
TMP: Recv: ACK
TMP: CLOSE SOCKET
```

```
DSP: Send: AUTH 215178 12608
DSP: Recv: 4e9c91068136e76ed4116d06dee28d49 108852142
DSP: Send: bd7d149d22379666b35c08b8e5f5f41e
DSP: Recv: WR: 207 LF: 65 RM: 0.0 IT: 24.4 AT: 2.7 GS: 120 LD: 939
          WG: 5.3 DP: 32 0 dfd30dc8aa501150243c1d729cec3288
DSP: Send: ACK
DSP: CLOSE SOCKET
```

Mit der aktivierten Debug-Funktion „DISPLAY_DATA_TO_UART“ gibt das Display-Board immer dann die folgenden Informationen aus, wenn neue Werte erfolgreich auf das Display geschrieben wurden. Dabei wird einerseits das Display in Textform dargestellt und andererseits der genaue Bitstrom für die Schieberegister der Displayanzeige ausgegeben:

Input:

```
WR: 207 LF: 65 RM: 0.0 IT: 24.4 AT: 2.7 GS: 120
LD: 939 WG: 5.3 DP: 32 0 dfd30dc8aa501150243c1d729cec3288
```

Output:

```
WR: 104 (208 deg) WG: 30 Rose = 0110100000011110
Display: 32 0 = 001000000000000000
| AT | GS | LD | ROSE | DP | WG | IT | RM | LF |
| 27 | 120 | 939 | XXXX | XXXX | 53 | 244 | 00 | 65 |
```

```
AT: 2.7 C          IT: 24.4 C
GS: 120 W/m2      RM: 0.0 mm
LD: 939 hPa       LF: 65 o/o
WG: 5.3 m/s
```

```
|      AT      ||      GS      ||      LD      |
0111001010101010000000010000110101001001110011010
```

```
|      RO      ||      DP      ||      WG      |
000111100110100000100000000000000011010110101010
```

```
|      IT      ||      RM      ||      LF      |
0100010000101010000000000101010100101011010101010
```

Wenn die Debug-Funktion „TCP_RETURN_WEATHER_DATA“ aktiv ist, wird die Authentifizierung bei einer Verbindungsanfrage übersprungen und die Wetterdaten werden sofort übergeben. Die anderen Debug-Funktionen sind selbsterklärend und dienen zum Beispiel der Erzeugung von zufälligen Wetterdaten.

1.8 Implementierung der Boards

Als Hardware für die Daten-Boards werden PIC-Web-Boards⁵ der Firma Olimex verwendet, die jeweils mit einem PIC18F67J80-Mikrocontroller⁶ betrieben werden.



Abbildung 1.6: PIC-Web Development-Board von Olimex

Das Sensor-Board ist über ein kurzes Flachband-Kabel mit dem „Graßl-Board“ verbunden und hängt außen auf dem Dach der Hochschule München. Um die Elektronik vor Feuchtigkeit zu schützen wurde es mit Plastik-70-Schutzlack überzogen. Damit die analogen Wert möglichst wenig verfälscht werden, wurden das Potentiometer und der Temperaturfühler des PIC-Boards abgewickelt. Das Weatherstation-Board wird unverändert betrieben.

Das Display verwendet das „PIC-P67J60“-Board⁷ von Olimex, dass zum PIC-Web sehr verwandt ist.

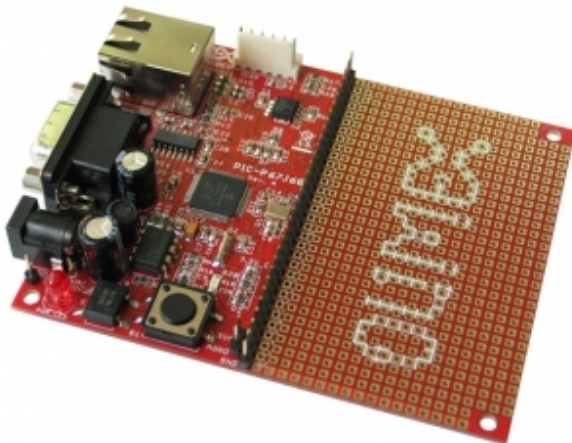


Abbildung 1.7: PIC-P67J60 Prototype-Board von Olimex

Auf der Lochraster-Platine des Boards wurden die Anschlüsse für die Schieberegister und für den Innentemperatur-Sensor aufgelötet. Über dieses Anschlüsse wird der serielle Datenstrom für die Schieberegister parallel auf den drei Anschlüssen an die Schieberegister ausgegeben. Die Verbindung besteht aus vier Datenleitungen: Clock, Ground, Data und Store.

⁵www.olimex.com/Products/PIC/Development/PIC-WEB/

⁶<http://ww1.microchip.com/downloads/en/DeviceDoc/39762f.pdf>

⁷www.olimex.com/Products/PIC/Proto/PIC-P67J60/

Pin 1 - Store
 Pin 2 - Ground
 Pin 3 - Data
 Pin 4 - Clock

Der Clock sorgt bei steigender Taktflanke für das weiterschieben der Daten die an Data anliegen. Wenn die Daten bis in die richtigen Register geschoben wurden, werden diese mit dem Low-aktiven Store für einen Takt übernommen und an den Ausgängen der Schieberegister ausgegeben. Diese sind mit einem BCD-Decoder verbunden, der aus jeweils 4 Bit eine Ziffer erstellt, die auf den Siebensegmentanzeigen angezeigt werden. Eine leere Stelle wird durch „1010“ definiert. Da der BCD-Decoder lediglich Nummern versteht, aber keine Zeichen, ist die erste Stelle bei der Außen- und Innentemperatur für das Minuszeichen reserviert, indem nur der mittlere Strich angeschlossen ist. Um ein Minus auszugeben wird eine „8“ auf dies Stelle ausgegeben.

Beispiel für die Umsetzung der Eingangsdaten in drei Bitstreams für die Schieberegister:

WR: 207 LF: 65 RM: 0.0 IT: 24.4 AT:- 2.7 GS: 120 LD: 939 WG: 5.3 DP: 32

```
|      AT      ||      GS      ||      LD      |
| 7|| 2||  || -|| 0|| 2|| 1||  || 9|| 3|| 9||  | (1)
011100101010100000000010000110101001001110011010
```

```
|      RO      ||      DP      ||      WG      |
| WS3 || 104 || 32 ||      || 3|| 5||  ||  | (2)
00011110011010000010000000000000011010110101010
```

```
|      IT      ||      RM      ||      LF      |
| 4|| 4|| 2||  || 0|| 0||  ||  || 5|| 6||  ||  | (3)
0100010000101010000000000101010100101011010101010
```

Das erste Byte der Windrose gibt die Geschwindigkeit des Windes nach der Beaufortskala⁸ in zehn Stufen an, die aus Fehlersicherheitsgründen Hemming-codiert ist.

Beaufortskala der Windstärke (Hemming-codiert):

WS0	00000000	(WS < 0.3 m/s)
WS1	00000111	(0.3 m/s <= WS < 1.6 m/s)
WS2	00011001	(1.6 m/s <= WS < 3.4 m/s)
WS3	00011110	(3.4 m/s <= WS < 5.5 m/s)
WS4	00101010	(5.5 m/s <= WS < 8.0 m/s)
WS5	00101101	(8.0 m/s <= WS < 10.8 m/s)
WS6	00110011	(10.8 m/s <= WS < 13.9 m/s)
WS7	00110100	(13.9 m/s <= WS < 17.2 m/s)
WS8	01001011	(17.2 m/s <= WS < 20.8 m/s)
WS9	01001100	(sonst)

⁸<http://de.wikipedia.org/wiki/Beaufortskala>

Da die Windrose „RO“ nur eine Auflösung von zwei Grad hat, wird nur die Hälfte der gewünschten Windrichtung übermittelt und im zweiten Byte angegeben. Ein Wert über 180 veranlasst die Windrose sich zurückzusetzen und genau nach Norden auszurichten. Das erste Byte der Witterungsanzeige „DP“ gibt bitweise die Element an, die aufleuchten sollen. Dabei bedeutet ein High-Pegel im Schieberegister, das das jeweils angeschlossene Element aktiv sein soll. Da nur die ersten fünf Bit verwendet werden, ist das zweite Byte immer Null.

Pin 1 - Sonne oben
Pin 2 - Sonne mitte
Pin 3 - Wolken unten
Pin 4 - Regen
Pin 5 - Schnee

Die Firmware der Boards wurde in der Entwicklungsumgebung MPLAB X⁹ von Microchip in der Programmiersprache C geschrieben. Dabei wird für die Kommunikation über Ethernet der TCP/IP-Stack¹⁰ von Microchip verwendet. Das Projekt kann mit MPLAB X über „File/Open Project > DonnerwetterFirmware.X“ geöffnet und bearbeitet werden. Über die Datei „Donnerwetter/DonnerwetterFirmware/Config/ProjectConfig.h“ kann die Firmware angepasst und konfiguriert werden. Mit den defines „BOARD_WEATHER“, „BOARD_SENSOR“, „BOARD_DISPLAY“ und „BOARD_DEBUG“ kann das jeweilige Ziel-Board ausgewählt werden. Dabei werden dann nur die jeweiligen Einstellungen verwendet, die weiter unten zu finden sind. Über das define „ENABLE_DEBUG_MODE“ können die Debug-Features der Boards aktiviert werden. Die Firmware wurde so entwickelt, dass auf einem physikalischen Board alle Module gleichzeitig laufen können. Da die einzelnen Module an unterschiedlichen Ports lauschen, können sie sich eine IP-Adresse teilen. Beachten Sie, dass auf dem Display-Board die Module „Innentemperatur“ und „Display“ gleichzeitig laufen.

Um ein Board zu flashen muss das Projekt zuerst compiliert werden über „Run/Clean and Build Main Project“. Dazu muss der Compiler MPLAB C18¹¹ installiert sein der in einer 60 Tage Testversion betrieben werden kann, die die volle Optimierung unterstützt. Wenn das Projekt erfolgreich compiliert hat kann es ins Board geflasht werden. Dazu kann zum Beispiel der Programmer MPLAB ICD 3¹² oder das PICKit 3¹³ verwendet werden. Achten Sie darauf, dass nicht versehentlich mehrere Boards im Netzwerk die gleichen IP- und MAC-Adresse haben.

Der Quellcode der Firmware der Boards und des Donnerwetter-Clients ist mit ausführlichen Kommentaren versehen und hilft die Implementierung anhand des Codes zu verstehen. Dieser ist im Verzeichniss „DonnerwetterFirmware“ in den Ordnern „Code“ und „Config“ zu finden. Der verwendete Quellcode des TCP/IP-Stacks von Microchip befindet sich in „TCPIP Stack“ und „Include“.

Um den Quellcode des Donnerwetter-Client zu kompilieren muss folgender Befehl ausgeführt werden:

```
/home/mike/Donnerwetter/DonnerwetterClient/  
gcc -Wall -o donnerwetter $(mysql_config --cflags)  
donnerwetter.c $(mysql_config --libs)
```

⁹www.microchip.com/mplabx/

¹⁰www.microchip.com/TCPIP/

¹¹www.microchip.com/c18/

¹²www.microchip.com/ICD3/

¹³www.microchip.com/pickit3/

Anhang

A.1 Literatur

1. Olimex Ltd. - *PIC-WEB-B [Stand: Rev B, Mai 2009]*,
<http://www.olimex.com/dev/pdf/PIC/PIC-WEB-B.pdf>
2. Olimex Ltd. - *PIC-P67J60 development board [Stand: Rev C, Mai 2009]*,
<https://www.olimex.com/Products/PIC/Proto/PIC-P67J60/resources/PIC-P67J60.pdf>
3. Microchip Technology Inc. - *PIC18F97J60 Family Data Sheet [Stand 2009]*,
<http://www.microchip.com/wwwproducts/Devices.aspx dDocName=en026439>
4. Peet Bros Company, Inc. - *ULTIMETER 800 OWNERS MANUAL*,
<http://www.peetbros.com/pdf/ULTIMETER800Manual.pdf>
5. Maxim Integrated TM - *DS18S20 High-Precision 1-Wire Digital Thermometer*,
<http://datasheets.maximintegrated.com/en/ds/DS18S20.pdf>

Abbildungsverzeichnis

1.1	Schematische Darstellung der Kommunikationswege	1
1.2	Pluviometer, Anemometer und Globalstrahlungsmesser	2
1.3	Ulimeter 800 und Weatherstation Board	2
1.4	Graßl-Board und Sensor-Board	3
1.5	Display/Temperatur-Board	3
1.6	PIC-Web Development-Board von Olimex	20
1.7	PIC-P67J60 Prototype-Board von Olimex	20