

Ein Arduino-basiertes UV-Index Messgerät mit Datenerfassung für die jahreszeitliche Untersuchung der UV-Belastung der Bevölkerung

Bernd Laquai, 28.03.2025

Einleitung

Betrachtet man energiereiche Strahlung im Hinblick auf die Gefährlichkeit für den Menschen, dann ist die spontane, emotionale Bewertung meist die, dass die radioaktive Strahlung, die beim Zerfall instabiler Atome entsteht, an erster Stelle steht. Diese emotionale Bewertung ist sicher durch die militärische Anwendung der Kerntechnik und durch die Unfälle und Katastrophen bei der Anwendung der Kerntechnik in der Energieerzeugung geprägt. Geht man aber (in theoretischer Weise) von Friedenszeiten aus und von einem sicheren Betrieb von Nuklearanlagen, dann wären die Gefahren der radioaktiven Strahlung für den Menschen in der Tat eher gering. Dies ist vor allem einer zwischenzeitlich weitgehend international umgesetzten Strahlenschutz-Gesetzgebung zu verdanken, welche der kommerziellen Nutzung von Kerntechnik und der Nutzung von radioaktiver Strahlung selbst für medizinische Zwecke sehr strikte Auflagen macht. Geht man nun davon aus, dass die radioaktive Strahlung durch eine staatliche Regulierung in ihrer Gefährlichkeit ziemlich entschärft ist (eben von militärischen Konflikten und nuklearen Katastrophen abgesehen), dann bewegt sich -für viele ganz unerwartet- eine andere energiereiche Strahlung an die erste Stelle in der Gefährlichkeit für den Menschen: das Sonnenlicht mit seinem natürlichen UV-Strahlungsanteil.

Im Strahlenschutz für die ionisierende Strahlung wird die Gefährlichkeit für den Menschen mit dem „Detriment“ (Schadensmaß) quantifiziert, das ist eine Auftretenswahrscheinlichkeit für einen Schaden, die mit der Schwere des möglichen Schadens gewichtet ist. Es sind vorwiegend strahlungsinduzierte Krebserkrankungen, die hierbei als möglicher Schaden klassifiziert werden. Würde man nun ein vergleichbares Konzept auf energiereiches UV-Licht anwenden, dann müsste man die Wahrscheinlichkeit des Auftretens von Hautkrebs und die Schwere dieser Erkrankungen bewerten. Interessant ist, dass die Hautkrebserkrankungen in den letzten Jahren stark angestiegen sind. Man geht zwischenzeitlich von einer Inzidenzrate von rund 300000 Fällen pro Jahr in Deutschland aus, von denen etwa 4000 Fälle tödlich enden, meist weil der Hautkrebs metastasieren und sich so auf andere Organe ausbreiten kann. Hautkrebs ist damit eine der häufigsten Krebsarten und die Todesfälle sind ungefähr doppelt so häufig als beispielsweise die Todesfälle, die dem radioaktiven Gas Radon in Wohnhäusern zugeschrieben werden (ca. 2100 pro Jahr).

Die radioaktive Strahlung und die UV-Strahlung haben, was die schädigende Wirkung anbelangt einiges gemeinsam. Sie können die Erbinformation (DNS) in den Zellen des Gewebes schädigen und so zu Mutationen führen. Es gibt eine große Latenzzeit für die Spätschäden, wobei Krebsinzidenzen zu den Spätschäden zählen, die nach einer Akkumulation einer gewissen Strahlungsdosis über eine bestimmte Zeit erst Jahre später auftreten. Solange die Strahlung in einem normalen natürlichen und sinnvollen Rahmen bleibt, dann sorgt zunächst ein äußerst leistungsfähiges Immunsystem dafür, dass sowohl DNS-Schäden durch UV-Strahlung wie auch die DNS-Schäden durch radioaktive Strahlung sehr effizient repariert werden. Selbst wenn das nicht mehr geht, können die hochgradig defekten Zellen vom Immunsystem immer noch gezielt abgetötet werden, so dass der Körper diese, dann ausscheidet.

Bemerkenswert ist aber auch, dass der Mensch auf die UV-Strahlung in kleinen Dosen für die Vitamin-D Synthese sogar angewiesen ist. Sie ist also lebensnotwendig. Genauso weiß man, dass das schwach radioaktive Kalium für den Menschen lebensnotwendig ist, weil es für die Funktion der Nervenleitung benötigt wird. Das natürliche Kalium ist radioaktiv, weil immer ein kleiner Teil des Kaliums (0.0117%) aus dem radioaktiven Isotop K40 besteht. Da der menschliche Stoffwechsel die Kaliumkonzentration im Körper automatisch reguliert, wird die Radioaktivität im menschlichen Körper auf einen niedrigen und ungefährlichen aber dennoch leicht nachweisbaren Wert eingestellt.

Nun ist es so, dass wie bei der radioaktiven Strahlung auch bei der UV-Strahlung des Sonnenlichts die übermäßige Exposition beim gesunden Menschen dazu führt, dass die Reparaturleistung des Immunsystems überfordert wird, defekte Zellen nicht mehr vollständig repariert oder ausgesondert werden, und es so zur Krebsinzidenz kommen kann. Bei der UV-Strahlung im Sonnenlicht entsteht die übermäßige Exposition meist dadurch, dass zu wenig oder gar kein Sonnenschutz verwendet wird und oft auch kein körpereigener Schutz (vorgebräunte Haut) bzw. ein hochsensibler Hauttyp vorliegt. Auch ein verändertes Freizeitverhalten oder unsinnige Schönheitsideale führen zur Überforderung des Immunsystems, genauso wie Klimaveränderungen und Umwelteinflüsse, die beispielsweise die Ozonschicht schädigen und so die UV-Belastung über ein sinnvolles Maß anwachsen lassen.

Da es aber oft auch die Unwissenheit über die Zusammenhänge zwischen Strahlung und möglichen Gesundheitsschäden ist, welche dann tatsächlich zu Schäden führt, ist es zweckmäßig sich mit der UV-Strahlung zu befassen. Da die Strahlenschutzgesetzgebung für die UV-Strahlung noch nicht so weit vorangeschritten ist, ist die Auseinandersetzung und das Lernen mit dieser energiereichen Strahlung noch deutlich einfacher möglich als bei radioaktiver Strahlung. So sind vor allem die UV-Strahlungsquellen meist frei erhältlich (z.B. in Form von UV-LEDs), was aber nicht heißt, dass sie deswegen ungefährlich oder weniger gefährlich als radioaktive Quellen sind. Es ist daher bei den Experimenten mit UV-Licht (auch mit Sonnenlicht) unbedingt erforderlich, auf einen entsprechenden Strahlungsschutz zu achten, vor allem auf den Augenschutz in Form einer UV-dichten Schutzbrille.

Während bei ionisierender Strahlung die notwendige Strahlungs-Quantendichte (Teilchen oder elektromagnetische Strahlung) für eine schädigende Wirkung auf ein Gewebe relativ gering sein kann, ist die notwendige Quantendichte der UV-Strahlung, die auf die Haut schädigend wirkt relativ hoch. Daher ist die Messtechnik, die einen Nachweis in dem schädigenden Bereich der Strahlung liefern soll, meist unterschiedlich zum Nachweise der ionisierenden Strahlung. Zwar benutzt man für die Detektion der UV-Strahlung heutzutage vorwiegend Halbleiterdioden, wie sie auch bei ionisierender Strahlung verwendet werden, aber man zählt nicht die Rate der einzelnen Wechselwirkungen, die sich in Form von einzelnen Pulsen darstellen lassen, wie bei einem Geigerzähler oder Szintillationszähler, sondern man misst den kontinuierlichen Fluss des von der energiereichen Strahlung ausgelösten Fotostroms durch die Halbleiterdiode. Die Selektion des entsprechenden Energiebereichs, in dem eine solche UV-Detektor Diode empfindlich sein soll, erfolgt meist durch die Wahl des entsprechenden Halbleitermaterials und seiner Dotierung (zum Beispiel in Form von ionenimplantierten SiC-Halbleitern). Das „Finetuning“ (z.B. auf den erythem-wirksamen UVB-Bereich) wird dann meist noch durch zusätzliche optische Wellenlängenfilter erreicht.

Auf Grund der geringen optischen Bandbreite der zu detektierenden Strahlung sind die erzeugten Fotostrome von UV-Dioden meist sehr schwach. Daher ist eine Strom-Spannungswandlung mit hoher Impedanz und hoher Verstärkung erforderlich, um die Information über die

Bestrahlungsstärke auf ein Spannungssignal abzubilden, das dann einfach weiterverarbeitet werden kann (z.B. in Form einer Analog/Digitalwandlung). Dazu werden oft sogenannte Trans-Impedanz-Verstärker (trans-impedance amplifier, TIA) verwendet, die jedoch, wenn sie diskret aufgebaut sind, oft erheblich rauschen und sehr empfindlich gegen Störstrahlung sind. Da heute entsprechende Halbleiter-Verpackungstechnologien zur Verfügung stehen, die es ermöglichen, unterschiedlichste Halbleitertechnologien in ein Package zusammen zu integrieren, konnte man in den letzten Jahren einen starken Trend hin zum hochintegrierten „Smart-Sensor“ feststellen. Für die Messung der UV-Strahlung beispielsweise sind Sensoren entstanden, die Detektordiode, Trans-Impedanz-Verstärker und A/D-Wandler in einem Gehäuse integriert haben und mit einem digitalen, seriellen Bus (I2C oder SPI) konfiguriert und ausgelesen werden können. Durch Verwendung von leadless Packages oder Ball Grid Array Packages werden die Smart-Sensor Bauteile sehr klein und können daher nur noch maschinell gelötet werden.

Der Trend zum elektronischen Bauteil ohne Beinchen und zum rein maschinellen Löten ist zunächst aber einmal ein „Killing-Faktor“ für viele ambitionierte Hobby-Bastler, die mit Hilfe von selbst hergestellten elektronischen Lösungen im Rahmen von Citizen-Science Projekten an die vorderste Front der Forschung vorstoßen wollen. Glücklicherweise aber zeichnet sich mittlerweile aber eine Lösung in Form von Firmen ab, welche die sogenannte „Maker Szene“ mit kleinen „Break-Out“ oder Evaluierungsplatinen beliefern, auf welche selbst die neuesten hochintegrierten Smart-Sensoren aufgelötet werden und dann über Stiftleisten oder kleine standardisierten Konnektoren zugänglich gemacht werden. Das gleiche gilt für die Mikrocontroller, die bis zu den höchsten Leistungsklassen heutzutage auf kleinen Platinen mit Konnektoren und Stiftleisten versehen, fertig verfügbar sind. Dann können die Sensoren sehr einfach und schnell an die Mikrocontroller angeschlossen werden können. Meist werden sogar die Treiber Funktionen für die Smart-Sensoren in Form von Bibliotheken für die Mikrocontroller-Entwicklungsplattform (wie zum Beispiel für die Arduino Controller) mitgeliefert, so dass auch der Programmieraufwand für die hochkomplexen Bauteile stark vereinfacht wird. Auf diese Weise wird es als auch im Citizen Science Bereich wieder einfach möglich zum Beispiel die UV-Belastung der Bevölkerung kostengünstig und mit einfach beherrschbaren Mitteln zu untersuchen.

Zielsetzung

Wenn man sich nun also zum Ziel setzt, das UV-Spektrum des Sonnenlichts zu vermessen, um zum Beispiel einen UV-Index als Belastungskriterium für den UV-Strahlungsanteil im Sonnenlicht so zu berechnen, so dass er dem erythem-wirksamen (sonnenbrand-wirksamen) Strahlungsanteil in etwa entspricht (siehe Abb. 2), dann kommt man derzeit an dem neuen spektralen UVA/B/C Sensor AS7331 von ams-Osram eigentlich nicht herum. Dieser Sensor wird von Sparkfun auf einer Breakout-Platine mit der Artikelnummer SEN-23517 und der Bezeichnung „SparkFun Spectral UV Sensor - AS7331 (Qwiic)“, für die „Maker-Community“ verfügbar gemacht und die Firma Sparkfun stellt darüber hinaus auch noch eine gute und umfangreiche Arduino Library zu dem doch recht komplexen Sensor zur Verfügung. Damit kann man die Performance des Sensors mit wenig Aufwand recht umfangreich austesten und auch schnell in Applikationen direkt integrieren. Ganz billig ist das Board mit etwa 30Euro zwar nicht (Preis bei der Firma Botland), aber die Qualität des Sensors und die der Sparkfun-Implementierung ist schon ausgesprochen gut und sein Geld durchaus wert. Es gibt noch weitere Implementierungen des ams-Osram Sensors AS7331 auf Breakout-Boards, z.B. von MIKROE (UV 5 Click) oder von iorodeo.com, sowie von einigen chinesischen Anbietern.

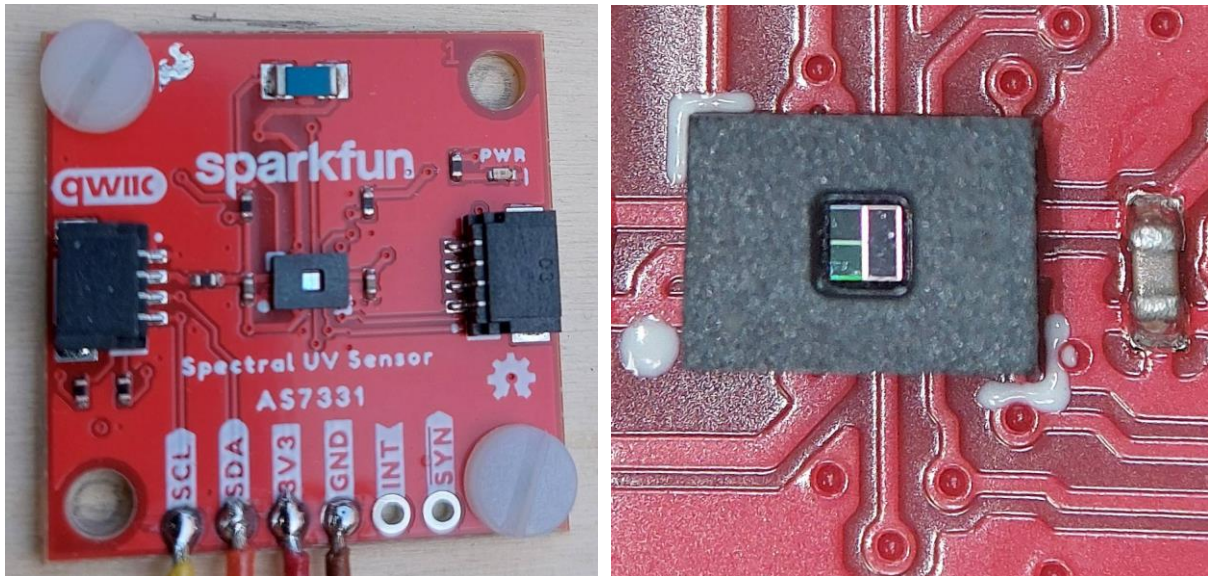


Abb. 1a, b: Sparkfun Breakout-Platine mit dem AS7331 Spectral UV-Sensor von ams-Osram, unter dem Mikroskop kann man die 3 UV-Dioden für UVA, UVB und UVC deutlich erkennen

Es werden noch etliche andere UV-Sensoren auf Breakout-Boards angeboten, die deutlich kostengünstiger sind und ebenfalls die Maker-Szene adressieren. Aber oft ist die Spezifikation der spektralen Empfindlichkeit nicht angegeben oder sie ist recht unklar. In sehr vielen Fällen wird die UV-empfindliche Diode des Typs GUVA-S12SD von Geni-uv.com (Korea) verwendet, deren spektrale Empfindlichkeit sich über fast den gesamten UV-Bereich von 240-370nm erstreckt. Wenn man aber wirklich Wert darauf legt, dass man einen Messwert erhält, der vorwiegend nur den UVB-Strahlungsanteil im Sonnenlicht abdeckt, der einerseits für die Vitamin-D Synthese in der Haut wichtig ist, aber andererseits für die Hautschädigung und die Kanzerogenese in der Haut bei hohen Strahlungsdosen verantwortlich ist, dann bewertet eine breitbandige UV-Diode den UVA-Anteil in der Regel zu stark. Einen UV-Index Wert aus der UVA-Strahlung abzuleiten ist in der Regel sehr ungenau, da das Verhältnis von UVB und UVA nicht konstant ist und stark von den Wetter-Verhältnissen in der Atmosphäre abhängt.

Auch wenn es z.B. darum geht, die Gefährlichkeit von UVC-Desinfektion Lampen (LEDs und Gasentladungslampen) zu bewerten, ist es ebenfalls zweckmäßig einen Sensor zu verwenden, dessen spektrale Empfindlichkeit nur im UVC-Bereich liegt und der dann nicht auf den UVA und UVB-Bereich anspricht. Der spektrale UV-Sensor von ams-Osram hat nun den Charme, dass er für jeden UV-Bereich, also für UVA, UVB und UVC eine eigene UV-Diode in den Sensor integriert hat, die sich spektral nur minimal überlappen. Das heißt für diesen Zweck kann dann speziell der UVC-Kanal verwendet werden. Die Verstärkung und die A/D Wandlung aller 3 spektralen Kanäle ist ebenfalls in dem Sensor integriert, genauso wie eine digitale I2C Schnittstelle für die Konfiguration und die Messdaten Erfassung. Durch die Integration der Verstärkung und der A/D-Wandlung hat man deutlich geringere Probleme mit Rauschen und mit Einstreuungen, im Vergleich zu Lösungen wo man diese Funktionalität mit separaten Komponenten bereitstellen und diese entsprechend verdrahten muss. Die Konfigurationsmöglichkeiten des AS7331 Sensors hinsichtlich Verstärkung und Integrationszeit erstrecken sich allerdings über alle 3 Kanäle gleichzeitig und sind nicht für jeden Kanal einzeln möglich.

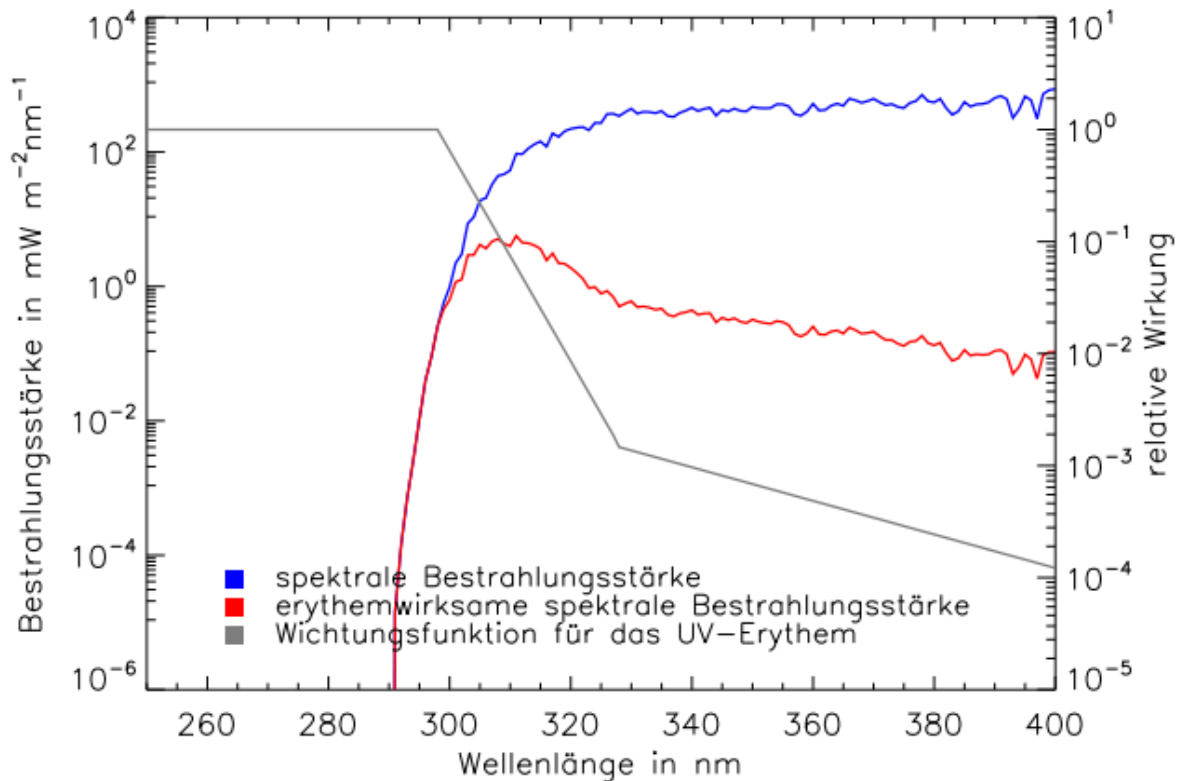


Abb. 2: Leistungsdichtespektrum des Sonnenlichts (blau), die erythemale Gewichtungsfunktion (Aktionsspektrum, grau) und das bewertete Leistungsdichtespektrum für die Erythem-Wirksamkeit des Sonnenlichts nach [1]. Deutlich kann man den Schwerpunkt der erythem-wirksamen Bestrahlungsstärke auf dem UVB-Bereich (280-315nm) erkennen

Schließlich verfügt der Sensor noch über einen thermischen Sensor, was insbesondere für Anwendungen in Zusammenhang mit Sonnenstrahlung, wo sich das Bauteil z.B. in einem Gehäuse stark erwärmt, wichtig werden könnte. Die Verwendung dieser Messdaten würde dann eine Kalibration über der Temperatur ermöglichen. Was allerdings etwas unangenehm anmutet, ist das relativ dicke Gehäuse, welches eine Bestrahlung mit einer geringen Winkel-Abweichung aus der Normalen-Richtung erforderlich macht, um nicht eine oder mehrere der drei Fotodioden teilweise abzuschatten. Das Datenblatt gibt dazu einen Winkel von $\pm 10^\circ$ für die Einfallrichtung der optischen Strahlung an. Zudem ist es auch schwierig das Deckglas für die UV-Dioden in der Gehäuseöffnung zu reinigen.

Auch wichtig zu erwähnen ist, dass das Sparkfun Breakout Board eine 3.3V Spannungsversorgung benötigt und deswegen auch entsprechende digitale Spannungspegel verwendet. Daher kommt es mit den 5V IO's älterer Mikrocontroller nicht so ohne weiteres klar.

Messanordnung

Die Herangehensweise für den Aufbau eines Prototyps für das UV-Index-Messgeräts mit Datenerfassung bestand nun darin, dass zunächst eine einfache Halterung angefertigt wurde, mit der sich die Sensorfläche in Normalen-Richtung auf die mittlere Sonnenhöhe einstellen lässt und mit der sich der Sensor dem Sonnen-Azimuth-Winkel manuell nachführen lässt. Mit dieser Messanordnung wurden Messwerte des UV-Sensors für die 3 Kanäle in Abständen von 5

Sekunden automatisiert gemessen und zusammen mit einem Zeitstempel auf eine Speicherkarte geschrieben. Gleichzeitig wurden Kalibrierwerte für den UV-Index mit Hilfe eines Referenzmessgeräts (Solarmeter 6.5R von solarlight.com) im 15-Minuten-Takt manuell erfasst. Die Messungen wurden an einem wolkenlosen Tag im März von 10Uhr vormittags bis ca. 2 Stunden nach dem Erreichen des Sonnenhöchststands durchgeführt, so dass zunächst ansteigende UV-Indexwerte (UVI) gemessen wurden, die zum Zeitpunkt des Sonnenhöchststandes ein Maximum erreichten und danach wieder abnahmen. Auf diese Weise entstanden Messwerte für die 3 spektralen Kanäle UVA, UVB und UVC für die unterschiedlichen UVI-Werte des Referenzmessgeräts.

Um nun Kalibrierwerte für den AS7331-Sensor zu erhalten, mit denen sich später der UV-Index ohne Verwendung des Referenzmessgeräts und nur aus den Sensor Messwerten für UVA, UVB und UVC errechnen lässt, wurde das Verfahren der multivariaten Regression eingesetzt. Dazu wurde angenommen, dass sich ein UVI-Wert mit Hilfe einer Linearkombination der UVA, UVB und UVC-Werte der 3 Sensor-Kanäle abschätzen lässt. Dies kann durch die folgende Gleichung ausgedrückt werden:

$$\text{UVI} = a \cdot \text{UVA} + b \cdot \text{UVB} + c \cdot \text{UVC}$$

Bereits mit 3 unterschiedlichen Referenz-Messwerten für UVI sowie UVA, UVB und UVC ergibt sich damit ein lineares Gleichungssystem mit den 3 Unbekannten a, b und c, das lösbar ist. Verwendet man mehr als 3 Messwerte, ist das Gleichungssystem überbestimmt, und die 3 Unbekannten a, b und c können mit Hilfe einer Regressionsrechnung so bestimmt werden, dass eine Approximation des tatsächlichen Zusammenhangs entsteht und der Fehler im Sinne des kleinstmöglichen Fehlerquadrats minimal wird (least mean square fit).

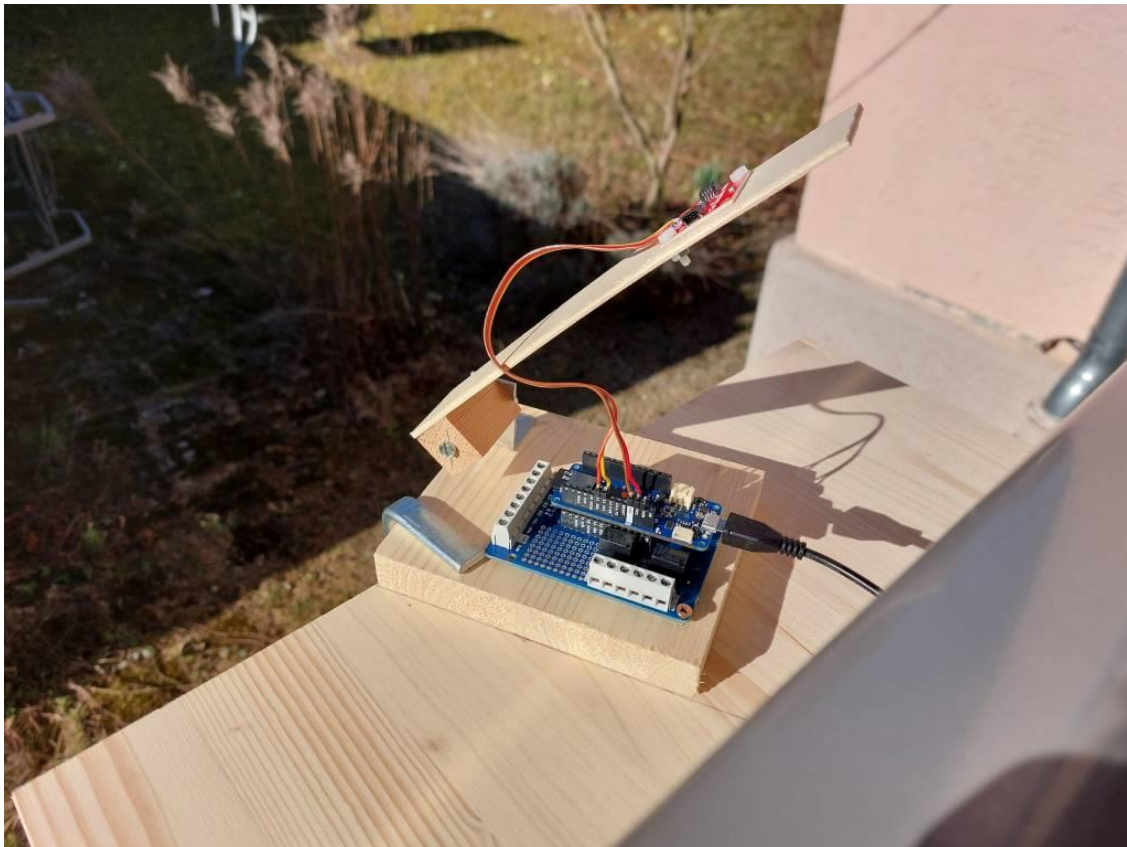


Abb. 3: Sensorhalterung mit ausgerichtetem Sensor und Arduino MKR Zero Mikrocontroller

In Abbildung 3 ist die rote Sparkfun Sensorplatine zu sehen, die auf einen schräggestellten Trägerstreifen des Prototyps montiert ist und auf die Sonne ausgerichtet ist. Der Trägerstreifen ist in der Neigung justierbar und auf der Trägerplatte für den Mikrocontroller montiert, die auf der ebenen Grundplatte mit einer Schraubklemme verdrehbar fixiert ist. Die Trägerplatte mit dem Mikrocontroller wurde alle 15 Minuten neu auf den Sonnen-Azimuth eingestellt und dann wieder auf der Grundplatte fixiert. Die Neigung des Trägerstreifens wurde über die gesamte Messzeit konstant gehalten.

Die Messwerte des AS7331 Sensors wurden über die I2C-Schnittstelle von einem Mikrocontroller des Typs MKR Zero von Arduino erfasst. Dieser wurde auf ein Arduino MKR Relay Proto Shield aufgesteckt, das ebenfalls auf der ebenen Trägerplatte montiert war. Als Programm (Sketch) für den Mikrocontroller wurde das in der Sparkfun Library vorhandene Beispielprogramm „Example01_Basic_OneShot“ dahingehend abgeändert, dass es die Echtzeituhr (Real Time Clock, RTC) verwendet, um für jeden Messwert einen Zeitstempel zu erzeugen und dass es die Messdaten anschließend auf die SD-Karte schreibt. Zusätzlich wurde noch der Temperatursensor ausgelesen und zusammen mit den 3 Strahlungsmesskanälen auf die SD-Karte geschrieben, so dass später eventuell noch eine Temperatur-Korrektur auf die Daten gerechnet werden kann. Die Messdaten wurden zur Kontrolle auch noch auf den Serial Monitor ausgegeben.

Die Sparkfun Platine wird über die Anschlüsse VCC und GND mit einer Spannung von 3.3V versorgt und die I2C Signalleitungen SDA und SCL der Sensorplatine wurden mit den gleichnamigen Pins des Mikrocontroller verbunden.

Damit das Programm auf dem Arduino MKR Zero korrekt läuft, müssen neben der SparkFun_AS7331_Arduino_Library noch die SparkFun_Toolkit und die RTCZero-master Bibliothek mit in das Bibliotheksverzeichnis der Arduino Entwicklungsplattform eingebunden werden.

Das Listing 1 zeigt das Programm, das für den Arduino MKR Zero zur Konfiguration, zum Auslesen des Sensors und für das Data Logging auf SD-Karte verwendet wurde.

```
//Derived from Sparkfun AS7331 Example
OneShot
//for MKZERO
//SD card
//derived from:
https://docs.arduino.cc/tutorials/mkr-
zero/mkr-zero-data-logger/
//use MKRZERO RTC
#include <SPI.h>
#include <SD.h>

const int chipSelect = SDCARD_SS_PIN;
char fileName[15] = "datalog.txt";
File myFile;

//RTC
//derived from
https://docs.arduino.cc/tutorials/mkr-
zero/mkr-zero-rtc-alarm/
//requires RTCzero library
//use my logging style
#include <RTCZero.h>
/* Create an rtc object */
RTCZero rtc;

//has no battery use reference date and time
/* Change these values to set the current
initial time */
const byte seconds = 0;

const byte minutes = 0;
const byte hours = 0;

/* Change these values to set the current
initial date */
const byte day = 1;
const byte month = 1;
const byte year = 25;

//AS7331
#include <Arduino.h>
#include <SparkFun_AS7331.h>
#include <Wire.h>

SfeAS7331ArdI2C myUVSensor;

void setup() {

    // Open serial communications and wait for
    port to open:
    Serial.begin(9600);

    pinMode(LED_BUILTIN, OUTPUT); // indicate
    SD card issue

    /*while (!Serial) { //endlos Schleife wenn
    kein Serial Monitor!
        ; // wait for serial port to connect.
        Needed for native USB port only
    }
}
```

```

    */
    Serial.print("Initializing SD card...");
    // see if the card is present and can be
    initialized:
    if (!SD.begin(chipSelect)) {
        Serial.println("Card failed, or not
        present");
        //blink LED fast to indicate SD card
        issue
        while (1){
            digitalWrite(LED_BUILTIN, HIGH);
            delay(250);
            digitalWrite(LED_BUILTIN, LOW);
            delay(250);
        }
    }
    Serial.println("card initialized.");
    if (!SD.exists(fileName)) {
        myFile = SD.open(fileName, FILE_WRITE);
        myFile.println("###");
        myFile.flush();
    }
    else {
        myFile = SD.open(fileName, FILE_WRITE);
        myFile.println("----");
        myFile.flush();
    }

    rtc.begin(); // initialize RTC 24H format
    rtc.setTime(hours, minutes, seconds);
    rtc.setDate(day, month, year);

    Wire.begin();
    // Initialize sensor and run default
    setup.
    if (myUVSensor.begin() == false)
    {
        Serial.println("Sensor failed to
        begin. Please check your wiring!");
        Serial.println("Halting...");
        while (1)
            ;
    }
    Serial.println("Sensor began.");
    // Set measurement mode and change device
    operating mode to measure.
    if
    (myUVSensor.prepareMeasurement(MEAS_MODE_CM
    D) == false)
    {
        Serial.println("Sensor did not get set
        properly.");
        Serial.println("Halting...");
        while (1);
    }
    Serial.println("Set mode to command.");
}

void loop() {
    //AS7331
    // Send a start measurement command.
    if (ksfTkErrOk !=
    myUVSensor.setStartState(true))
        Serial.println("Error starting
        reading!");

    // Wait for a bit longer than the
    conversion time.
    delay(2 +
    myUVSensor.getConversionTimeMillis());

    // Read UV values.

    if (ksfTkErrOk !=
    myUVSensor.readAllUV())
        Serial.println("Error reading
        UV.");

    // Read Temp value.
    if (ksfTkErrOk != myUVSensor.readTemp())
        Serial.println("Error reading
        temp.");

    Serial.print(rtc.getDay());
    Serial.print(".");
    Serial.print(rtc.getMonth());
    Serial.print(".");
    Serial.print(rtc.getYear());
    Serial.print(" ");
    Serial.print(rtc.getHours()); //retrieve
    hours
    Serial.print(":");
    Serial.print(rtc.getMinutes());
    //retrieve minutes
    Serial.print(":");
    Serial.print(rtc.getSeconds());
    //retrieve seconds
    Serial.print(" ");
    Serial.print(myUVSensor.getUVA());
    Serial.print(" ");
    Serial.print(myUVSensor.getUVB());
    Serial.print(" ");
    Serial.print(myUVSensor.getUVC());
    Serial.print(" ");
    Serial.print(myUVSensor.getTemp());
    Serial.println();

    myFile.print(rtc.getDay());
    myFile.print(".");
    myFile.print(rtc.getMonth());
    myFile.print(".");
    myFile.print(rtc.getYear());
    myFile.print(" ");
    myFile.print(rtc.getHours()); //retrieve
    hours
    myFile.print(":");
    myFile.print(rtc.getMinutes());
    //retrieve minutes
    myFile.print(":");
    myFile.print(rtc.getSeconds());
    //retrieve seconds
    myFile.print("\t");
    myFile.print(myUVSensor.getUVA());
    myFile.print("\t");
    myFile.print(myUVSensor.getUVB());
    myFile.print("\t");
    myFile.print(myUVSensor.getUVC());
    myFile.print("\t");
    myFile.print(myUVSensor.getTemp());
    myFile.println();
    myFile.flush();
    digitalWrite(LED_BUILTIN, HIGH);
    delay(50); //flash to indicate file write
    digitalWrite(LED_BUILTIN, LOW);
    delay(5000);
}

```

Listing 1: Arduino-Sketch für das UV-Index Messgerät mit dem spektralen UV-Sensor AS7331

Zur Verwendung der Real Time Clock (RTC) auf dem MKR Zero ist zu sagen, dass sich diese das Datum und die Uhrzeit nur merken kann, wenn auch der Prozessor läuft, d.h. wenn das Board mit Spannung versorgt ist. Ein Stützbatterie für die RTC gibt es auf dem Board leider nicht. Man kann eine lückenlose Spannungsversorgung für die Uhr aber dadurch garantieren, dass man einen Akku am entsprechenden Steckverbinder anschließt. Dann bleibt der Prozessor stets unter Spannung. Ist kein Akku angeschlossen, dann verliert die Uhr das Datum und die Zeit beim Ausschalten. Daher ist es zusätzlich sinnvoll, ein Startdatum und einen Startzeitpunkt per Software vorzugeben und sich zu notieren, wann man die Spannungsversorgung angeschaltet hat. So kann man später die geloggtten Zeitstempel der Messwerte der wirklichen Zeit sicher zuordnen. Ansonsten folgt die Verwendung der RTC dem Beispiel unter:

<https://docs.arduino.cc/tutorials/mkr-zero/mkr-zero-rtc-alarm/> .

Um den Temperaturwert aus dem Sensor auszulesen (hierfür gibt es von Sparkfun kein Beispiel), muss man zu Beginn der Hauptschleife zunächst den Rohwert aus dem Sensor auslesen:

```
// Read Temp value.
if (ksfTkErrOk != myUVSensor.readTemp())
    Serial.println("Error reading temp.");
```

Dann kann man später, wenn man beispielsweise mit der .print() Methode auf den Serial monitor oder die Speicherkarte schreiben möchte, den Messwert einfach mit .print(myUVSensor.getTemp()) abholen und umgerechnet in Grad Celsius ausgeben, sofern die Variable „myUVSensor“ zuvor als Klasse des Typs „SfeAS7331ArdI2C“ vereinbart wurde.

Ergebnisse

In Abbildung 4 sind nun die Messwerte des UVA-Kanals über der Tageszeit dargestellt. Die Sparkfun-Library erledigt dabei auch die Umrechnung die Rohdaten des A/D Wandlers (counts) in Bestrahlungsstärken in mW/m^2 .

Ab etwa 11Uhr kann man deutlich das viertelstündliche, manuelle Nachjustieren auf den Sonnen-Azimut deutlich erkennen. Der Verlauf der Messkurve zeigt wie erwartet bei 12:31h MEZ (12:00 wahre Ortszeit) ein Maximum. Danach nimmt der Messwert wieder ab und erreicht nach 13:00h MEZ einen instabilen Bereich. Dieser instabile Bereich wurde durch die Verschattung eines hohen Baums erzeugt, dessen Äste sich zusätzlich im Wind bewegten. Daher wurden für die Berechnung des UV-Indexes später nur Messwerte zwischen 10 und 13h MEZ herangezogen.

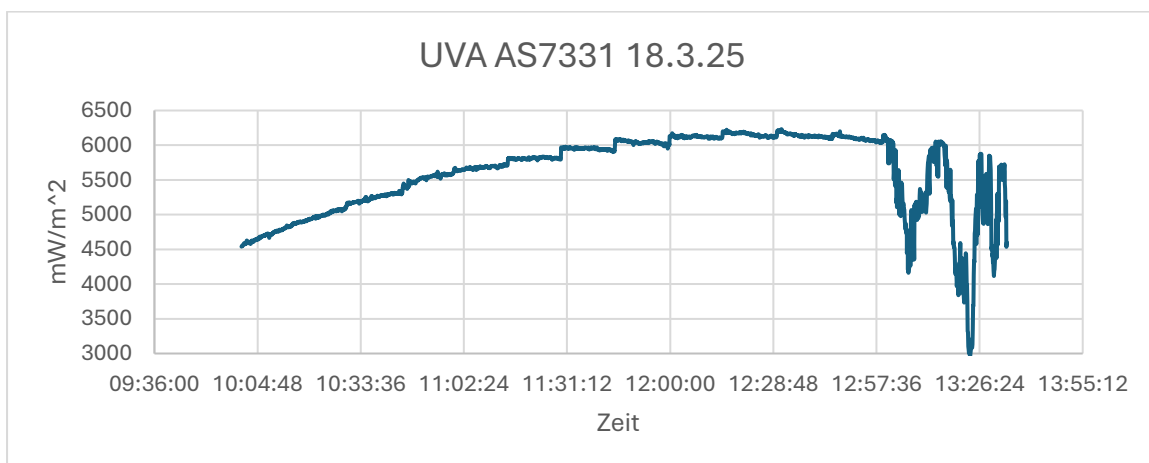


Abb. 4: Messwerte des UVA-Kanals des AS7331 Sensor

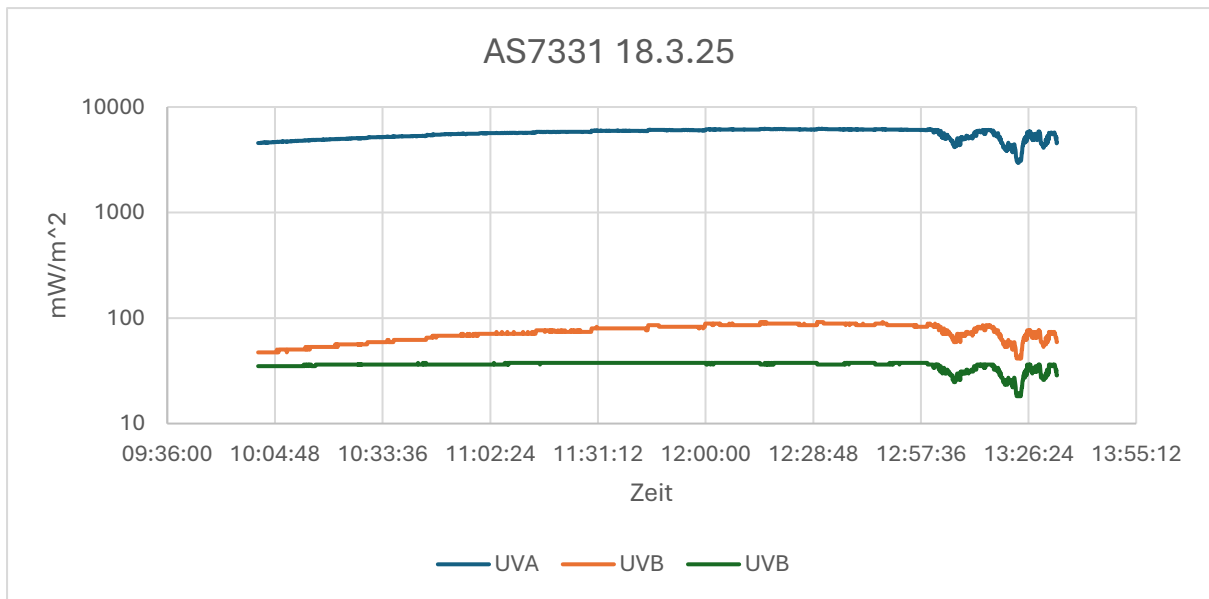


Abb. 5: Messwerte der 3 UV-Kanäle UVA, UVB und UVC des AS7331 Sensors

Abbildung 5 zeigt die Messwerte aller 3 Kanäle des AS7331. Man kann deutlich erkennen, dass der UVA-Anteil im Sonnenlicht um etwa den Faktor 100 größer ist. Noch deutlicher wird dies, wenn man die Verhältnisse der Kanäle darstellt (Abb. 6). Interessanterweise sind diese Verhältnisse nicht konstant, sondern hängen von der Tageszeit ab. Das vom AS7331 gemessene UVA/UVB Verhältnis schwankte zwischen 70 und 100, während das UVB/UVC Verhältnis zwischen 1.3 und 2.5 schwankte.

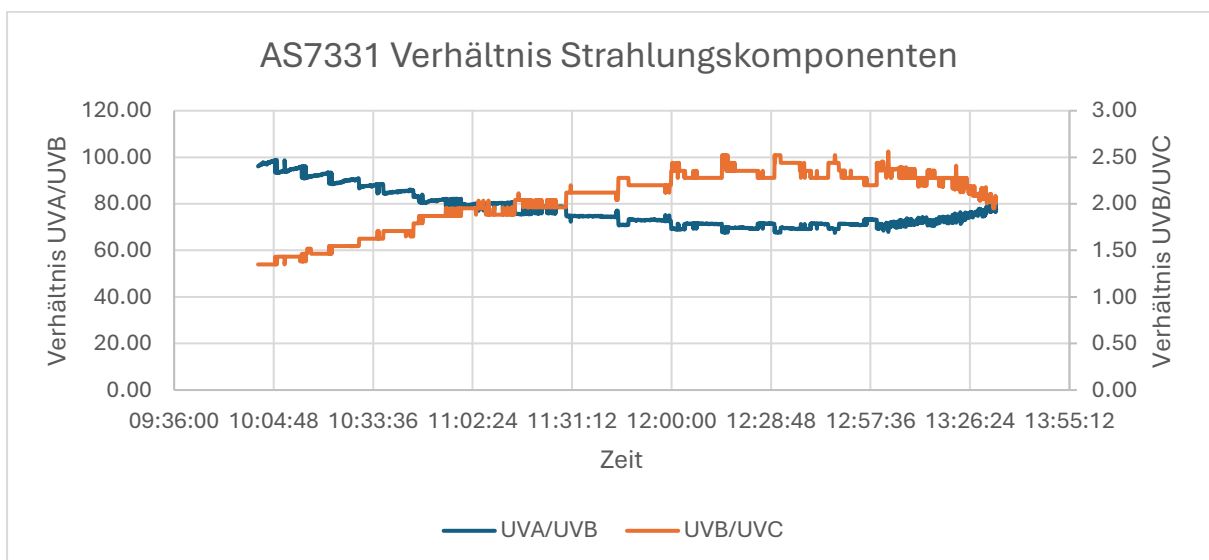


Abb. 6: Verhältnisse der 3 spektralen Kanäle des AS7331 Sensors

Betrachtet man nun die UVB-Messwerte des AS7331 im Vergleich zu den UVI-Werten, die mit dem Solarmeter 6.5R als Referenzmessgerät gemessen wurden (Abb. 7), so kann man eine sehr gute qualitative Übereinstimmung erkennen. Die UVB-Messwerte des AS7331 Sensors (orange) folgen praktisch identisch der Gestalt der UV-Index-Kurve (türkis) aus den Messwerten, welche das Referenzmessgerät durch Bewertung des Sonnenspektrums mit der Erythem-

Gewichtungsfunktion gemessen hat. Es muss dabei allerdings gesagt werden, dass die UVI-Messwerte durch Ausrichten des Solarmeters auf die Sonne (Azimut und Höhe) gewonnen wurden und nicht, wie es die offizielle Messvorschrift erfordert, durch Messung in senkrechter Richtung zum Boden. Die Ausrichtung auf die Sonne war erforderlich, da auch der Sensor auf die Sonne ausgerichtet werden musste, um zu verhindern, dass die Strahlungseintrittsöffnung im Sensorgehäuse die 3 Sensordioden unterschiedlich verschattet.

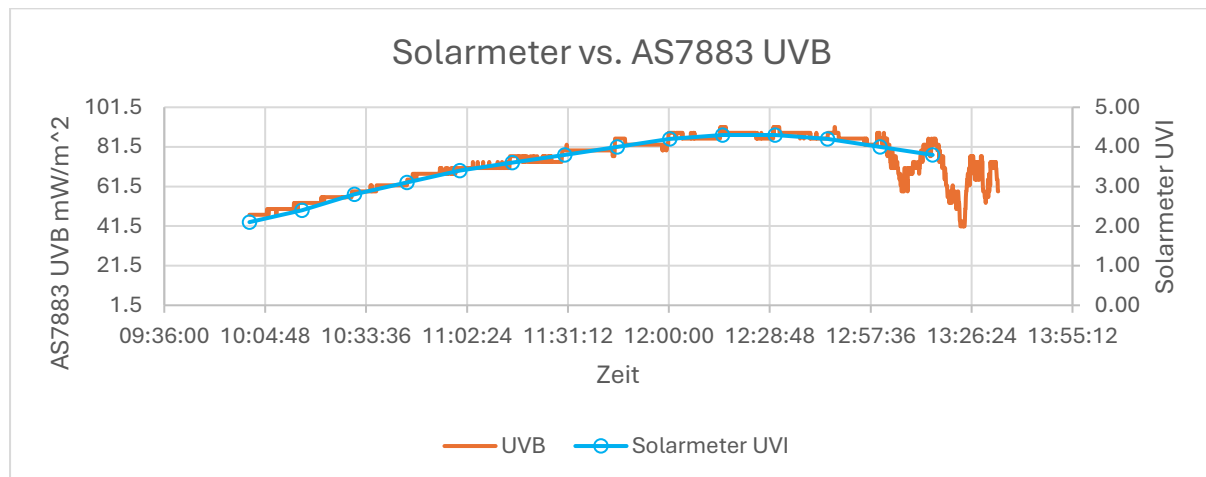


Abb. 7: Der UVB-Kanal des Sensors im Vergleich zur UV-Index Referenzmessung mit dem Solarmeter

Zum Vergleich wurden auch die UVI-Messwerte der Messstation Stuttgart im UV-Strahlungsmessnetzwerk des Bundesamts für Strahlenschutz (BfS) für diesen Tag abgefragt. Hier zeigte sich gegenüber den mit dem Solarmeter 6.5R gemessenen Werte ein gewisser Unterschied. So betrug der höchste gemessene UVI-Messwert der BfS-Messstation in Stuttgart nur etwa 3.8, während das Solarmeter einen Höchstwert von 4.3 anzeigte. Das liegt, wie oben bereits erwähnt wurde, mit großer Wahrscheinlichkeit daran, dass das Solarmeter auf die Sonne ausgerichtet wurde, während die Messtation des BfS den UVI definitionsgemäß senkrecht zur Erdoberfläche misst. Zudem liegt die Messtation des BfS geografisch gesehen etwa 200m tiefer, so dass die UVB-Strahlung vermutlich durch Streuung an der höheren Aerosolkonzentration im Talkessel von Stuttgart zusätzlich gedämpft wird.

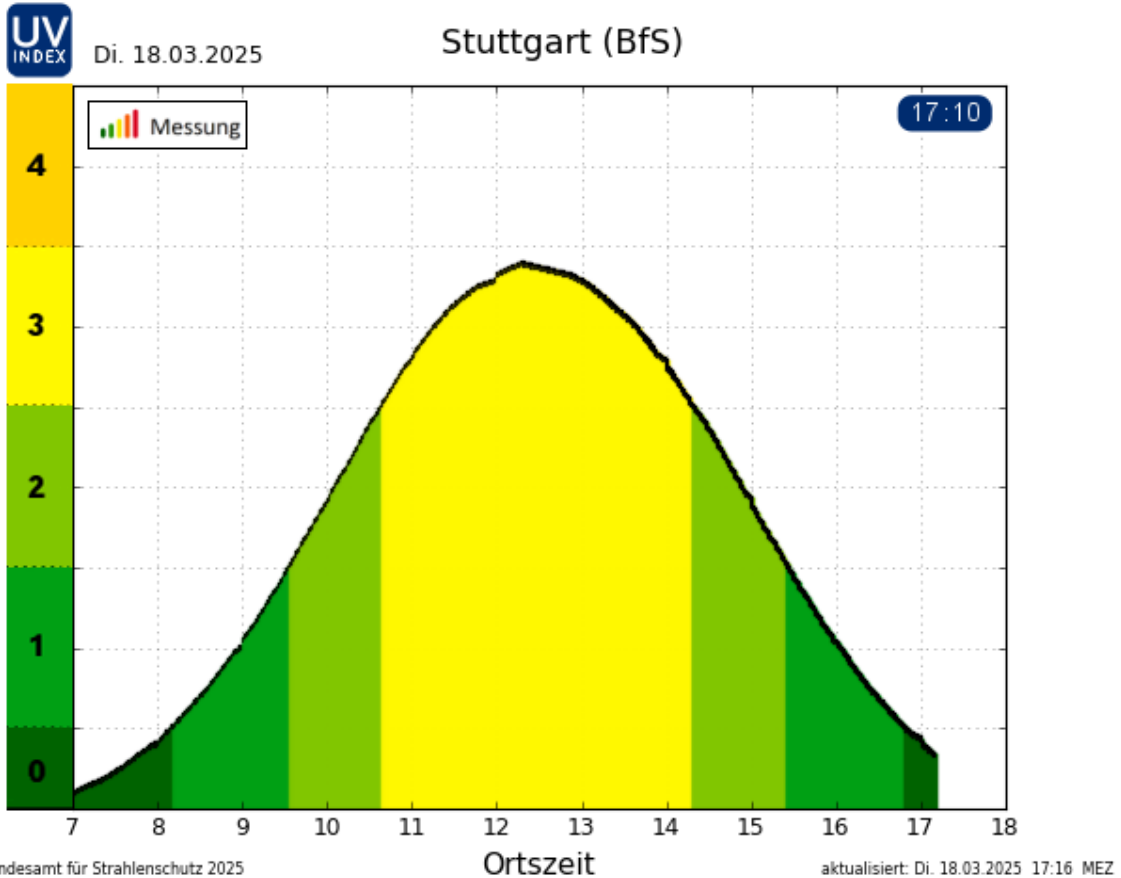


Abb. 8: Messergebnis der Messstation des BfS UV-Messnetzes für den UV-Index in Stuttgart am gleichen Tag

Nachdem die automatisierte Messdaten-Erfassung auf SD-Karte mit Hilfe des Mikrocontrollers abgeschlossen war und parallel dazu alle 15 min manuell ein UVI-Messwert mit dem Solarmeter als Referenzmessgerät (und mit einem UVA-Messgerät von Extech noch zusätzlich ein UVA-Wert erfasst wurde), wurden die Messdaten des Referenzmessgeräts und die des AS7331 Sensors über den Zeitstempel zusammengeführt. Dies wurde unter MS Excel mit Hilfe der Funktion SVERWEIS() erreicht, die in einer Spalte eines Feldes einen Wert (zum Beispiel einen Zeitwert) suchen kann, welcher einem Referenzwert (z.B. einem anderen Zeitwert) am nächsten kommt und dann in der gefundenen Reihe des Felds einen Wert in einer anderen Spalte aber in der gleichen Reihe als Messwert extrahiert. Auf diese Weise können die manuell erfassten und die automatisch gewonnen Messdaten über den Zeitstempel sehr einfach synchronisiert werden. Damit entstand nun folgendes Resultat zu 14 Zeitpunkten der manuellen UVI-Messung mit dem Referenzmessgerät:

Zeit	Solarmeter UVI	ExtechUVA	UVA	UVB	UVC
10:00	2.10	3.26	4542.19	47.25	35.02
10:15	2.40	3.30	4868.91	53.16	35.02
10:30	2.80	3.43	5158.44	59.06	36.31
10:45	3.10	3.66	5317.81	62.02	37.61
11:00	3.40	3.76	5639.22	70.87	36.31
11:15	3.60	4.00	5809.22	76.78	37.61
11:30	3.80	3.94	5947.34	79.73	37.61
11:45	4.00	4.13	6072.19	85.64	37.61
12:00	4.20	4.15	6133.28	88.59	37.61
12:15	4.30	4.16	6191.72	91.55	36.31
12:30	4.30	4.14	6220.94	91.55	36.31
12:45	4.20	4.19	6088.13	85.64	37.61
13:00	4.00	3.90	6101.41	85.64	36.31
13:15	3.80	3.80	6016.41	82.69	36.31

Tabelle 1: Messergebnisse für die synchronisierten Messdaten für die externen Referenz-Messgeräte und den Sensor AS7331

Diese Tabelle enthält nun neben den 15-minütigen UVI-Messwerten des Solarmeters (und dem hier nicht weiter verwendeten UVA-Messgerät von Extech) die UVA-, UVB- und UVC-Messwerte des AS7331-Sensors. Diese Daten wurden in ein mit Tabulatorzeichen getrenntes Textfile für die Weiterverarbeitung abgespeichert (measResult.txt).

Die Weiterverarbeitung zur Gewinnung der Koeffizienten a, b, c mit Hilfe der multivariaten Regression erfolgte mit Hilfe der Mathematik-Software Matlab bzw. der kostenlosen dazu kompatiblen GNU Software Octave (<https://octave.org/>), die dafür eine spezielle Routine (lscov()) bereitstellt. Dazu wurden die Daten aus der Tabelle 1 wieder eingelesen und zunächst die Kurvenverläufe der Messwerte über Zeit geplottet.

```
clear;
close('all');
infid = fopen('measResult.txt');
indat=textscan(infid,'%s','delimiter','\n');
nrLines=length(indat{1});
dtime = NaT(nrLines,1);
solarm = zeros(nrLines,1);
extUVA = zeros(nrLines,1);
UVA = zeros(nrLines,1);
UVB = zeros(nrLines,1);
UVC = zeros(nrLines,1);
for i=1:nrLines
    line = textscan(char(indat{1}(i)),'%s','delimiter','\t');
    dtime(i)=datetime(char(line{1}(1)));
    solarm(i)=str2double(char(line{1}(2)));
    extUVA(i)=str2double(char(line{1}(3)));
    UVA(i)=str2double(char(line{1}(4)));
    UVB(i)=str2double(char(line{1}(5)));
    UVC(i)=str2double(char(line{1}(6)));
end
UVI = solarm;
xArr = [UVA, UVB, UVC];
```

```

figure; semilogy(dtime, UVI*10, 'k-o'); hold on;
semilogy(dtime, UVA, 'b-o'); hold on;
semilogy(dtime, UVB, 'r-o'); hold on;
semilogy(dtime, UVC, 'g-o'); hold on;
title('Solarmeter vs. AS7331')
xlabel('Time');
ylabel('UVISm (sw), UVA (b), UVB (r), UVC (g)');

abc = lscov(xArr, UVI);
UVIextr = xArr*abc;
figure; plot(dtime, UVI, 'bx--'); hold on;
plot(dtime, UVIextr, 'ro'); hold off;
title('Solarmeter (b) vs. AS7331 (r)')
xlabel('Time');
ylabel('UVI');

fprintf('coeffUVA %.5f\n', abc(1));
fprintf('coeffUVB %.5f\n', abc(2));
fprintf('coeffUVC %.5f\n', abc(3));

rmsErr = sqrt(mean((UVIextr - UVI).^2));
fprintf('rmserr %.2f%%\n', rmsErr*100);

```

Listing 2: Matlab- (Octave-) Skript zur Berechnung der Kalibrier-Koeffizienten a, b und c

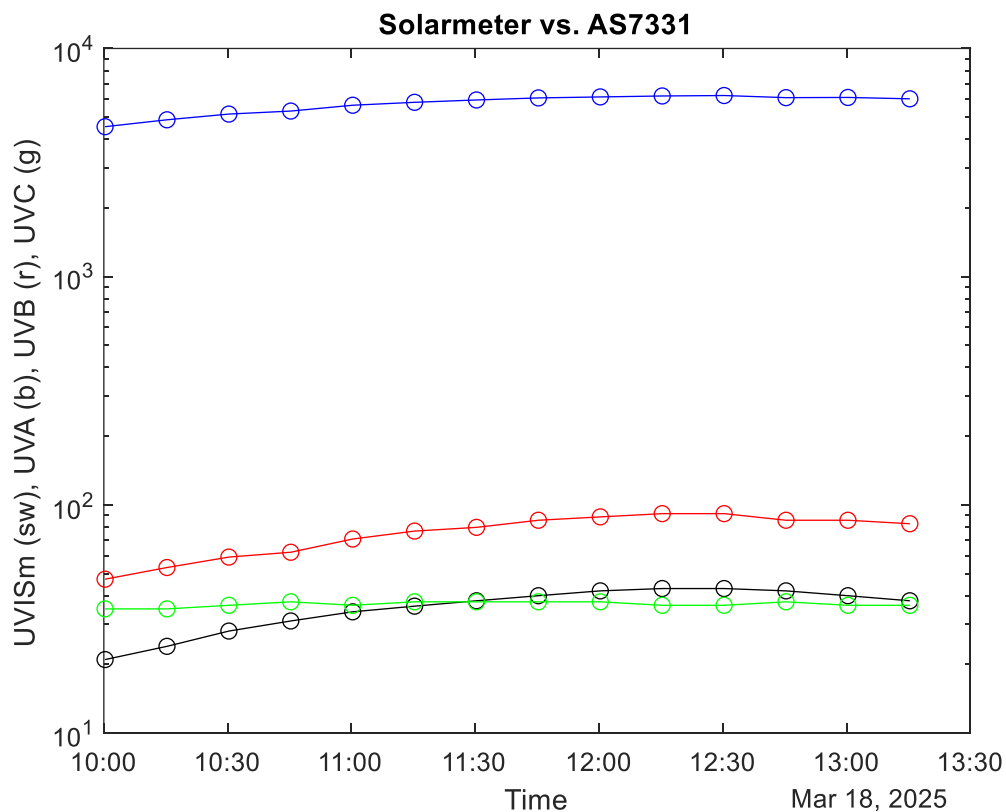


Abb. 9: Mit Matlab/Octave geplottete synchronisierte Messdaten

Die eingelesenen und zeit-synchronisierten Daten ergeben eine vergleichbare Grafik wie unter MS Excel (Abb. 5). Nun werden die Messdaten in den Matrizen xArr und UVI so abgelegt, dass mit

Hilfe der Anweisung $abc = \text{lscov}(xArr, UVI)$ die unbekannten Koeffizienten a , b und c zu der Gleichung $UVI_{extr} = a \cdot UVA + b \cdot UVB + c \cdot UVC$ berechnet werden können. Hierin sind jetzt UVI_{extr} die mit den extrahierten Koeffizienten a , b und c berechneten Schätzwerte für UVI. In Matrixschreibweise ergibt sich dann die Berechnungsvorschrift $UVI_{extr} = xArr \cdot abc$, bei der nur noch die Kalibrierkoeffizienten und die Messwerte des Sensors benötigt werden. Greift man nun auf die extrahierten Koeffizienten in der Matrix abc zu, erhält man folgendes Ergebnis:

rmserr: 8.05%

$a = \text{coeffUVA}: 0.00002$

$b = \text{coeffUVB}: 0.04760$

$c = \text{coeffUVC}: -0.00331$

rmserr: 8.05%

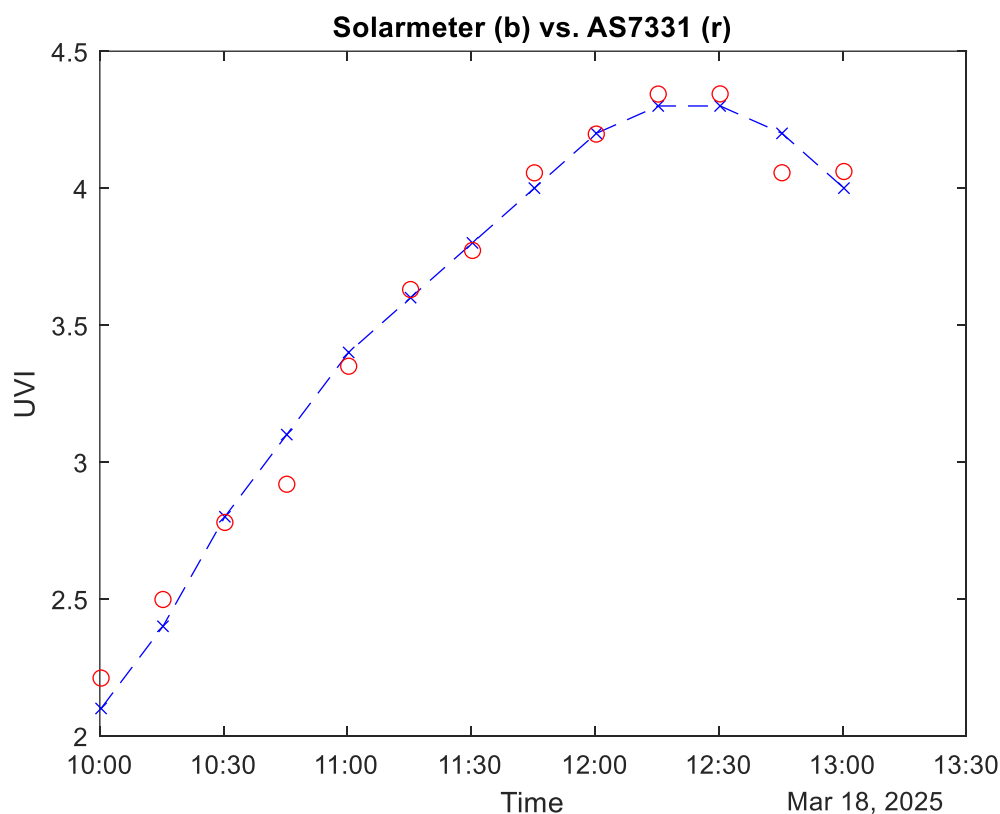


Abb. 10. Vergleich der mit dem Solarmeter gemessenen UVI-Messwerte mit denen, die mit Hilfe der Kalibrierkoeffizienten a , b und c , allein aus den Messdaten des Sensors berechnet wurden

Plottet man nun die Schätzwerte UVI_{extr} (rote Kreise) gegen die mit dem Referenzinstrument gemessenen Werte UVI (blaue Kreuze), kann man erkennen, dass eine Kalibration mit diesem Verfahren recht gut möglich ist und eine Approximation des UV-Index mit den Messwerten des Sensors für UVA, UVB und UVC und mit den nach obigem Verfahren extrahierten Kalibrierkoeffizienten a , b und c recht genau möglich ist. Der RMS-Fehler lag in diesem Experiment unter 10%. Das Solarmeter 6.5R wird mit einem auf das NIST rückführbaren Kalibrierzertifikat ebenfalls mit einer Genauigkeit von 10% angegeben.

Um einen weiteren Bereich an UVI-Werten sicher abdecken zu können, müsste eine vergleichbare Messung im Sommer bei höheren Maximalwerten für den UV-Index wiederholt werden. Auch eine Messung mit speziellen LED-UV-Emittern bei unterschiedlichen Wellenlängen wäre sinnvoll und würde die Messsicherheit bei höheren UV-Indexwerten oder anderer spektraler Zusammensetzung als dies beim Sonnenlicht der Fall ist, bestimmt noch weiter erhöhen.

Literatur

[1] Merle Glandorf; Untersuchungen zur Trenddetektion und zu hemisphärischen Unterschieden der spektralen UV-Bestrahlungsstärke; Dissertation an der Universität Hannover 2007

[2] Deutscher Wetterdienst DWD; Definition des UV Index;

https://www.dwd.de/DE/klimaumwelt/ku_beratung/gesundheit/uv/download/uv_def.pdf?__blob=publicationFile&v=7

[3] Mohammad-Reza Safaei; Dosimetrische Untersuchungen zur UV-Exposition bei unterschiedlichen Freizeitaktivitäten und zur UV-Protektion unterschiedlicher Sonnenschutzmittel; Dissertation an der Universität München 2005

[4] Fachverband für Strahlenschutz e.V.; Leitfaden „Sonnenstrahlung“; FS-2012-156-AKNIR

[5] BQS Institut für Qualität und Patientensicherheit; Evaluation der Screeninguntersuchungen auf Hautkrebs gemäß Krebsfrüherkennungs-Richtlinie des Gemeinsamen Bundesausschusses

[6] Wolfgang Johannes Schreier; UV-Strahlung und DNA-Schäden — Untersuchung UV-induzierter Prozesse in Nukleinsäuren mit Femtosekunden-Infrarotspektroskopie; Dissertation an der Universität München 2008