

🏠 Startseite / Die Zeitgleichung

Die Zeitgleichung

[Inhaltsverzeichnis \[Öffnen\]](#)

Version Januar 2005

Eine einfache Formel zu Sonnenaufgang und Untergang

Nicht selten erhalten wir folgende Frage: „Bitte senden Sie mir eine einfache aber genaue Formel zur Berechnung von Sonnenaufgang und Sonnenuntergang“. Die Begriffe „einfach“ und „genau“ widersprechen sich. Im Folgenden versuchen wir, einen brauchbaren Kompromiss zwischen „einfach“ und „genau“ zu finden. Die erreichte Genauigkeit wird bei wenigen Minuten liegen – genau genug für die Auf- und Untergangszeiten der Sonne.

Zunächst müssen ein paar Hilfsmittel bereitgestellt werden. Im Folgenden wird auf ein paar grundsätzliche Eigenschaften der scheinbaren Sonnenbewegung am Himmel eingegangen. Die Genauigkeit liegt (ausser polnahe Gebiete) im Bereich weniger Minuten.

- » Hier gibt es einfachere, etwas weniger genaue Formeln
- » Hier gibt es ein Java-Script, das den Tagbogen berechnet!
- » ein genaueres Java-Script für Berechnungen rund um Sonne/Mond!

Der Lauf der Sonne

Als Hilfsmittel benötigen wir zunächst eine Formel für die Koordinaten der Sonne am Himmel. Sie läuft scheinbar auf der Ekliptik über den Himmel. In einem (tropischen) Jahr legt sie 360 Grad auf der Ekliptik zurück. Den seit Frühlingsanfang zurückgelegten Weg nennen wir L. Die Rektaszension und Deklination der Sonne sei RA und DK.

Formeln Sonnenlauf

$T = (\text{Anzahl Tage seit 1.1.2000 Mittag Weltzeit})/36525$

$M = 2\pi * (0.993133 + 99.997361 * T)$ [radians]

$$L = 2\pi * (0.7859453 + M/2\pi + (6893.0*\sin(M)+72.0*\sin(2.0*M)+6191.2*T) / 1296000) \text{ [radians]}$$

$$\text{Neigung der Erdachse } e = 2\pi * (23.43929111 + (-46.8150*T - 0.00059*T*T + 0.001813*T*T*T)/3600.0)/360 \text{ [radians]}$$

$$\text{Deklination DK} = \text{asin}(\sin(e)*\sin(L)) \text{ [radians]}$$

$$\text{Rektaszension RA} = \text{atan}(\tan(L)*\cos(e)) \text{ [radians]}$$

Folgende beiden Korrekturen sind in der angegebenen Reihenfolge notwendig, da der Arcustangens nicht eindeutig ist:

1. Falls $RA < 0$, bitte π hinzuaddieren.
2. Falls L grösser ist als π , dann müssen Sie zu RA π hinzuaddieren.

Stellen Sie bitte Ihren Taschenrechner auf Bogenmass (Radians, RAD) um, oder multiplizieren Sie die Werte in den Klammern jeweils vor dem Drücken der sin-Taste mit 57.29578.

Nun sollten Sie die Rektaszension und die Deklination der Sonne auf etwa eine Bogenminute genau ausrechnen können. Achten Sie insbesondere auf die Definition der Zeit T . Es gibt genauere -jedoch aufwändigere - Methoden um den Lauf der Sonne zu berechnen. Für unsere Zwecke genügt dies völlig.

Unterschiede zwischen Sonnenuhr und Armbanduhr.

Traditionell basiert unsere Zeitrechnung auf dem (scheinbaren) Lauf der Sonne. Erst in moderner Zeit wurden Zonenzeiten und Atomuhren eingeführt. Wenn die Sonne genau im Süden steht, ist es 12 Uhr Mittags (sogenannte Ortszeit). Die Zeit zwischen zwei Mittaggen beträgt genau 24 Stunden. Zumindest sollte es so sein, doch gibt es Abweichungen.

Die Erde dreht sich in 23 Stunden und 56 Minuten einmal um ihre eigene Achse. Die restlichen 4 Minuten bis zur Erreichung der vollen Länge des Tages von 24 Stunden erklären sich dadurch, dass wir im Laufe des Tages etwas auf der Umlaufbahn der Erde um die Sonne vorangekommen sind. Deshalb steht die Sonne in einer leicht anderen Richtung. Die Erde muss sich noch 4 Minuten (ca. = 24 Stunden / 365) lang weiter drehen, bis für den gleichen Ort die Sonne wieder genau im Süden steht.

Die Neigung der Erdbahn und die leichte Exzentrizität der Erdbahn bewirken, dass die wahre Taglänge (Zeitspanne zwischen Mittag und dem folgenden Mittag) um den Mittelwert 24 Stunden schwankt. Man denkt sich eine mittlere [Sonne](#), die scheinbar entlang des Himmelsäquators mit konstanter Geschwindigkeit läuft. Für einen vollständigen (scheinbaren) Umlauf um den Himmelsäquator benötigt diese mittlere Sonne wie die reale Sonne ein Jahr. Die mittlere Ortszeit bezieht sich auf diese mittlere Sonne.

Eine mechanische (oder elektronische) Uhr läuft gleichmässig: Jeder Tag hat 24 Stunden. Deshalb kann eine solche Uhr nur die mittlere Ortszeit anzeigen. Die wahre Sonne kann bis zu 15 Minuten zu früh oder zu spät im Süden stehen als die gedachte mittlere Sonne. Dieser Unterschied wird **Zeitgleichung** genannt. Die Zeitgleichung kann auch als Unterschied zwischen einer die Ortszeit anzeigenden mechanischen Uhr und einer Sonnenuhr verstanden werden.

Die mittlere Sonne läuft auf dem Himmelsäquator. Ihre Rektaszension (RA_m) nimmt gleichmässig zu und ihre Deklination ist stets gleich null. Aus der hier nicht näher erläuterten Definitionsgleichung der Ortssternzeit für den Null-Meridian folgt (Ergebnis in Stunden):

$$RA_m = 18.71506921 + 2400.0513369*T + (2.5862e-5 - 1.72e-9*T)*T*T \text{ [Stunden]}$$

Die Zeitgleichung

Die Bewegung der Erde um die Sonne ist genau betrachtet eine komplizierte Angelegenheit. Selbst wenn die Erde der einzige Planet wäre, hätten wir es mit einer nur numerisch (d.h. nur durch spezielle Näherungsverfahren) lösbaren Gleichung zu tun. Die Erdbahn erfährt zudem noch kleine Störungen durch die anderen Planeten. Dies alles zu berücksichtigen, würde die eingangs gestellte Forderung nach Einfachheit verletzen.

Wir müssen deshalb einen anderen Weg gehen. Dieser besteht darin, dass wir durch die genauen Werte der Zeitgleichung eine Näherungskurve legen. Wenn man sich die in Fig. 1 als Funktion der Tagnummer dargestellte Zeitgleichung ansieht, so kann man sich zwei sich überlagernde harmonische Schwingungen vorstellen. In der Tat lässt sich damit mit der folgenden, aus zwei harmonischen Schwingungen zusammengesetzten Formel die Differenz zwischen Wahrer Ortszeit (WOZ, Sonnenuhr) und mittlerer Ortszeit (MOZ) auf besser als eine Minute genau annähern:

Zeitgleichung:

$$\text{WOZ} - \text{MOZ} = 1.0027379 * (\text{RAm} - \text{RA}) \text{ [Stunden]}$$

Die Berechnung von RA (Rektaszension der Sonne) und RA_m (Rektaszension der mittleren Sonne) finden Sie oben. RA müssen Sie noch in Stunden umrechnen: Multiplizieren Sie die Werte für RA in Bogenmass mit 3.81972, um die entsprechende Stundenanzahl zu erhalten.

In Tabellen finden Sie oft Werte der Zeitgleichung in Minuten. Um Vergleichen zu können, müssen Sie das Ergebnis hier mit 60 multiplizieren, denn hier wird die Zeitgleichung in Stunden verwendet.

Die Deklination der Sonne

Die Höhe eines Gestirns über dem Horizont beim Passieren der Südrichtung (Meridian) bestimmt, wie lange es über dem Horizont bleibt. Mit der Zeitgleichung kennen wir den Zeitpunkt, zu dem die Sonne im Süden steht (Mittag). Wenn wir noch wissen, wie lange die Sonne über dem Horizont bleibt, können wir die Aufgangs- und Untergangszeit berechnen. Dazu brauchen wir die Deklination der Sonne. Vereinfacht gesprochen ist die Deklination eines Gestirns der Breitengrad über dem das Gestirn im Zenit steht (dies stimmt nicht exakt, weil die Erde keine Kugel ist). Negative Werte bedeuten, dass die Sonne südlich des Äquators steht. Die Deklination der Sonne könnte man als einfache harmonische Schwingung darstellen:

$$\text{Deklination} = 0.40954 * \sin(0.0172 * (\text{T} - 79.35)) \text{ [radians]}$$

Ergebnis in Bogenmass. 180 normale Grad = pi in Bogenmass. Multipliziere mit 57.29578 für Ergebnisse in Grad.

Es zeigt sich jedoch, dass dies für genaue Berechnungen nicht ausreichend ist. Man verwende deshalb mit Vorteil die Deklination der Sonne, wie sie oben unter „Formeln Sonnenlauf“ als DK beschrieben wird.

Auf- und Untergang

Die Zeit vom wahren Mittag bis die Sonne eine bestimmte Horizonthöhe h erreicht, kann durch folgende Formel dargestellt werden. Sie ist exakt für Punkte, deren astronomische Koordinaten (Deklination, Rektaszension) fix sind und wenn der Einfluss durch die Erdatmosphäre ignoriert wird:

$$\text{Zeitdifferenz} = 12 * \arccos((\sin(h) - \sin(B) * \sin(DK)) / (\cos(B) * \cos(DK))) / \pi \text{ [Stunden]}$$

wobei B die geographische Breite bedeuten soll. Geographische Breiten auf der südlichen Hemisphäre haben einen negativen Wert. Hier sind die Werte für h, B und DK in Bogenmass zu verwenden. Falls diese in Grad vorliegen, dividieren Sie die Zahlen durch 57.29578. Wenn das Argument des arccos im Betrag grösser als eins ist, so erreicht die Sonne im Laufe des Tages nie die geforderte Horizonthöhe h. Dies kann z.B. im hohen Norden während der Polarnacht der Fall sein oder während der Zeit der Mitternachtssonne.

» Hier gibt es ein Java-Script, das diesen Tagbogen, d.h. die Zeitdauer zwischen Aufgang und Kulmination, bzw. Transit bis zum Untergang berechnet!

Auf- und Untergangszeit in mittlerer Ortszeit:

$$\text{Aufgang Ortszeit} = 12 - \text{Zeitdifferenz} - \text{Zeitgleichung} \text{ [Stunden]}$$

Untergang Ortszeit = 12 + Zeitdifferenz – Zeitgleichung [Stunden]

Unsere Uhren zeigen jedoch nicht die mittlere Ortszeit unseres Beobachtungsorts an, sondern eine bestimmte Zonenzeit. Diese Zonenzeit ist die mittlere Ortszeit eines bestimmten Längengrades. Die Weltzeit ist z.B. die mittlere Ortszeit von null Grad Länge. Geographische Längen westlich von Greenwich sind durch ein negatives Vorzeichen ersichtlich.

Aufgang = Aufgang Ortszeit – geographische Länge /15 + Zeitzone [Stunden]

Untergang = Untergang Ortszeit – geographische Länge /15 + Zeitzone [Stunden]

Zeitzone ist die Differenz Zonenzeit – Weltzeit. In Mitteleuropa beträgt diese Differenz während der Winterzeit eine Stunde (Mitteleuropäische Zeit MEZ), während der Sommerzeit MESZ zwei Stunden. Gemäss den gesetzlichen Regeln verwendet die EU jeweils zwischen den letzten Sonntagen im März und Oktober die Sommerzeit (Mitteleuropäische Sommerzeit MESZ).

Das Licht der Sonne wird in der Atmosphäre gebeugt. Es läuft besonders in Horizontnähe auf einer leicht zum Boden hin gekrümmten Bahn. Deshalb kann man die Sonne auch noch sehen, wenn sie rein geometrisch schon untergegangen ist. Deshalb wird der Untergang und auch der Aufgang der Sonne für eine geometrische Horizonthöhe h von -50 Bogenminuten berechnet. Von bürgerlicher Dämmerung spricht man, wenn $h = -6$ Grad ist, nautische Dämmerung entspricht $h = -12$ Grad und schliesslich astronomische Dämmerung entspricht $h = -18$ Grad.

Mit den oben stehenden Formel lassen sich, abgesehen für die Polarregionen Sonnenaufgang und Untergang sowie die Dämmerungszeiten auf besser als 5 Minuten berechnen.

Zahlenbeispiel (bezieht sich auf den C++ Quelltext unten) :

Es soll der Sonnenaufgang für die Mitte Deutschlands am 30. September 2005 bestimmt werden.

Wir verwenden 10 Grad Ost, 50 Grad Nord als geographische Position

Am 30. September 2005, sind seit dem 1.1.2000 2099 Tage vergangen.

Gregor = 486

JD = 2453644

T = 0.057467488021902803

B = $\text{Pi} * 50 / 180 = 0.87266462599716477$ rad

Koordinaten der Sonne:

M = 4.64786158141676 rad (bzw. 42.3469734 bevor M in das Interval 0 bis $2 * \text{Pi}$ gebracht wird)

L = 3.27133743280786 rad (bzw. 47.25363458 falls weder M, noch L in das Interval 0 bis $2 * \text{Pi}$ gebracht werden)

e = 0.40907976105658 rad

$RA = 3.2607373 \text{ rad} = 12.4551 \text{ Stunden}$, $DK = -0.0514860 \text{ rad}$

$RA_m (=RA_{\text{Mittel}}) = 12.639990750 \text{ Stunden}$ (bzw. $156.63999 \text{ Stunden}$, Rest von $156.63999/24 = 12.63999$)

Sonnenaufgangshöhe $h = -50 \text{ Bogenminuten} = -0.0145 \text{ rad}$

Zeitdifferenz = $5.8519235109278 \text{ Stunden}$.

Zeitgleichung:

$RA_{\text{Mittel}} - RA = 0.184891613194782 \text{ Stunden}$

Zeitgleichung $WOZ - MOZ = 0.185397827942548 \text{ Stunden}$ (= 11.12 Minuten)

Sonnenaufgang: Ortszeit = 5.962678 , Weltzeit = 5.29601199 Uhr

Die MOZ des 15. östlichen Längengrades ist die Mitteleuropäische Zeit MEZ.

Sonnenaufgang: Lokalzeit MOZ = 6.28081 Uhr MEZ , Lokale Sommerzeit = $7.29601 \text{ MESZ} = 7.17 \text{ Uhr}$

Also ist Sonnenaufgang für die Mitte Deutschlands um 7 Uhr 17 MESZ.

Sonnenuntergang: Ortszeit = 17.666525 , Weltzeit = 16.999859 Uhr

Sonnenuntergang: Lokalzeit MOZ = $17.999859 \text{ Uhr MEZ}$, Lokale Sommerzeit = $18.999859 \text{ MESZ} = 19.00 \text{ Uhr}$

Ein Vergleich mit CalSky.com ergibt 7 Uhr 18 für den Sonnenaufgang.

Das ist OK, mit so einfachen Formeln kann man keine bessere Genauigkeit erwarten.

Ich hoffe, dieses Zahlenbeispiel hilft bei der Anwendung dieser Formeln. **Beachten Sie auch, dass Sie die Formeln bei uns in einem Java-Script anwenden können. » HIER – ACHTUNG: JAVASCRIPT NOCH NICHT AN METHODE VERSION 2005 ANGEPASST!**

Programmversionen unserer Leser

Zahlreiche Leser haben sich durch unsere kleine Formelsammlung mit dem Problem der Berechnung des Sonnenlaufs beschäftigt. Dabei wurden diese Formeln in zahlreiche Programmiersprachen übertragen. Einige dieser Beispiele möchten wir hier ungeprüft(!) weiterreichen.

MS Excel von Axel Lopata:

SunsetTabelle.xls: einfache Berechnung in Excel und

Sunrise_Sunset.xl: mit kleiner Orte-Datenbank und Berechnung über 28 Tage.

Adobe Flash von Michael Arntz:

SunRiseSet_Code2010.html: Online-Beispiel

SunRiseSet_Code2010 fla: Flash-Code

SunRiseSet_CodePreview.html: Formeln-Code.

Fragen

Vor dem Kontakt zum Autor: Bevor Sie ein E-Mail mit dem Betreff „alle Formeln falsch“ schreiben, sollten Sie wissen, dass der untenstehende Quelltext das richtige Resultat liefert. Auch sind andere Leute offenbar damit klar gekommen.

Haftungsausschluss: Der Autor übernimmt keine verbindlichen Garantien für die Anwendbarkeit und Richtigkeit der in diesem Artikel angegebenen Formeln. Man vergleiche die Ergebnisse auf jeden Fall zuerst mit anderen Berechnungen, z.B. mit den online abfragbaren Auf- und Untergängen von CalSky!

Einfaches Quelltextbeispiel in C++

```
#include <iostream>

#include <cmath>

using namespace std;

double pi2=6.283185307179586476925286766559;

double pi=3.1415926535897932384626433832795;

double RAD = 0.017453292519943295769236907684886;

double JulianischesDatum ( int Jahr, int Monat, int Tag,

                           int Stunde, int Minuten, double Sekunden )

// Gregorianischer Kalender

{

int Gregor;

if (Monat<=2)

{
```

```
    Monat=Monat +12;

    Jahr = Jahr -1;

}

Gregor = (Jahr/400)-(Jahr/100)+(Jahr/4); // Gregorianischer Kalender

return 2400000.5+365.0*Jahr - 679004.0 + Gregor

+ int(30.6001*(Monat+1)) + Tag + Stunde/24.0

+ Minuten/1440.0 + Sekunden/86400.0;

}

double lnPi(double x)

{

int n = (int)(x/pi2);

x = x - n*pi2;

if (x<0) x +=pi2;

return x;

}

double eps(double T) // Neigung der Erdachse

{

return RAD*(23.43929111 + (-46.8150*T - 0.00059*T*T + 0.001813*T*T*T)/3600.0);

}
```

```
double BerechneZeitgleichung(double &DK, double T)

{

double RA_Mittel = 18.71506921

+ 2400.0513369*T +(2.5862e-5 - 1.72e-9*T)*T*T;

double M = lnPi(pi2 * (0.993133 + 99.997361*T));

double L = lnPi(pi2 * ( 0.7859453 + M/pi2

+ (6893.0*sin(M)+72.0*sin(2.0*M)+6191.2*T) / 1296.0e3));

double e = eps(T);

double RA = atan(tan(L)*cos(e));

if (RA<0.0) RA+=pi;

if (L>pi) RA+=pi;

RA = 24.0*RA/pi2;

DK = asin(sin(e)*sin(L));

// Damit 0<=RA_Mittel<24

RA_Mittel = 24.0*lnPi(pi2*RA_Mittel/24.0)/pi2;

double dRA = RA_Mittel - RA;

if (dRA < -12.0) dRA+=24.0;

if (dRA > 12.0) dRA-=24.0;
```



```
dRA = dRA* 1.0027379;

return dRA ;

}

int main()

{

double JD2000 = 2451545.0;

double JD;

JD = JulianischesDatum(2005,9,30,12,0,0); // Testdatum

/*

// manuelle Dateneingabe

int year, month, day;

cout << „Berechnung Sonnenufgang und -untergang“ << endl;

cout << „Jahr (YYYY): „; cin >> year;

cout << „Monat (MM): „; cin >> month;

cout << „Tag (DD): „; cin >> day;

JD = JulianischesDatum(year,month,day,12,0,0);

*/

double T = (JD - JD2000)/36525.0;
```

```
double DK;

double h = -50.0/60.0*RAD;

double B = 50.0*RAD; // geographische Breite

double GeographischeLaenge = 10.0;

//double Zeitzone = 0; //Weltzeit

//double Zeitzone = 1; //Winterzeit

double Zeitzone = 2.0; //Sommerzeit

double Zeitgleichung = BerechneZeitgleichung(DK,T);

double Minuten = Zeitgleichung*60.0;

double Zeitdifferenz = 12.0*acos((sin(h) - sin(B)*sin(DK)) / (cos(B)*cos(DK)))/pi;

double AufgangOrtszeit = 12.0 - Zeitdifferenz - Zeitgleichung;

double UntergangOrtszeit = 12.0 + Zeitdifferenz - Zeitgleichung;

double AufgangWeltzeit = AufgangOrtszeit - GeographischeLaenge /15.0;

double UntergangWeltzeit = UntergangOrtszeit - GeographischeLaenge /15.0;

double Aufgang = AufgangWeltzeit + Zeitzone; // In Stunden

if (Aufgang<0.0) Aufgang +=24.0;

else if (Aufgang>=24.0) Aufgang -=24.0;

double Untergang = UntergangWeltzeit + Zeitzone;

if (Untergang<0.0) Untergang +=24.0;
```

```
else if (Untergang>=24.0) Untergang -=24.0;

int AufgangMinuten = int(60.0*(Aufgang - (int)Aufgang)+0.5);

int AufgangStunden = (int)Aufgang;

if (AufgangMinuten>=60.0) { AufgangMinuten-=60.0; AufgangStunden++; }

else if (AufgangMinuten<0.0) {

AufgangMinuten+=60.0; AufgangStunden--;

if (AufgangStunden<0.0) AufgangStunden+=24.0;

}

int UntergangMinuten = int(60.0*(Untergang - (int)Untergang)+0.5);

int UntergangStunden = (int)Untergang;

if (UntergangMinuten>=60.0) { UntergangMinuten-=60.0; UntergangStunden++; }

else if (UntergangMinuten<0) {

UntergangMinuten+=60.0; UntergangStunden--;

if (UntergangStunden<0.0) UntergangStunden+=24.0;

}

cout << „Aufgang „<< AufgangStunden << „:“;

if (AufgangMinuten<10.0) cout << „0“;

cout << AufgangMinuten << endl;

cout << „Untergang „<< UntergangStunden << „:“;

if (UntergangMinuten<10.0) cout << „0“;

cout << UntergangMinuten << endl;
```

// Vergleich mit CalSky.com

// Aufgang : 7h18.4m Untergang : 19h00.6m

Unsere Teleskopguides

- [Teleskop](#)
- [Teleskop Test](#)
- [Teleskop kaufen](#)
- [Teleskop für Einsteiger](#)
- [Teleskop für Kinder](#)

AFFILIATELINKS/WERBELINKS

Die mit Sternchen (*) gekennzeichneten Links sind sogenannte Affiliate-Links. Wenn Sie auf einen dieser Affiliate-Links klicken und über diesen einkaufen, erhalten wir von dem betreffenden Online-Shop oder Anbieter eine Provision. Für Sie verändert sich der Preis nicht und wir können die Seite weiter kostenlos zur Verfügung stellen.

Über Uns

[Kontakt](#)
[Datenschutzerklärung](#)
[Impressum](#)

© Copyright 2022, Alle Rechte vorbehalten | Wir ♥ Astronomie!

 Deutsch