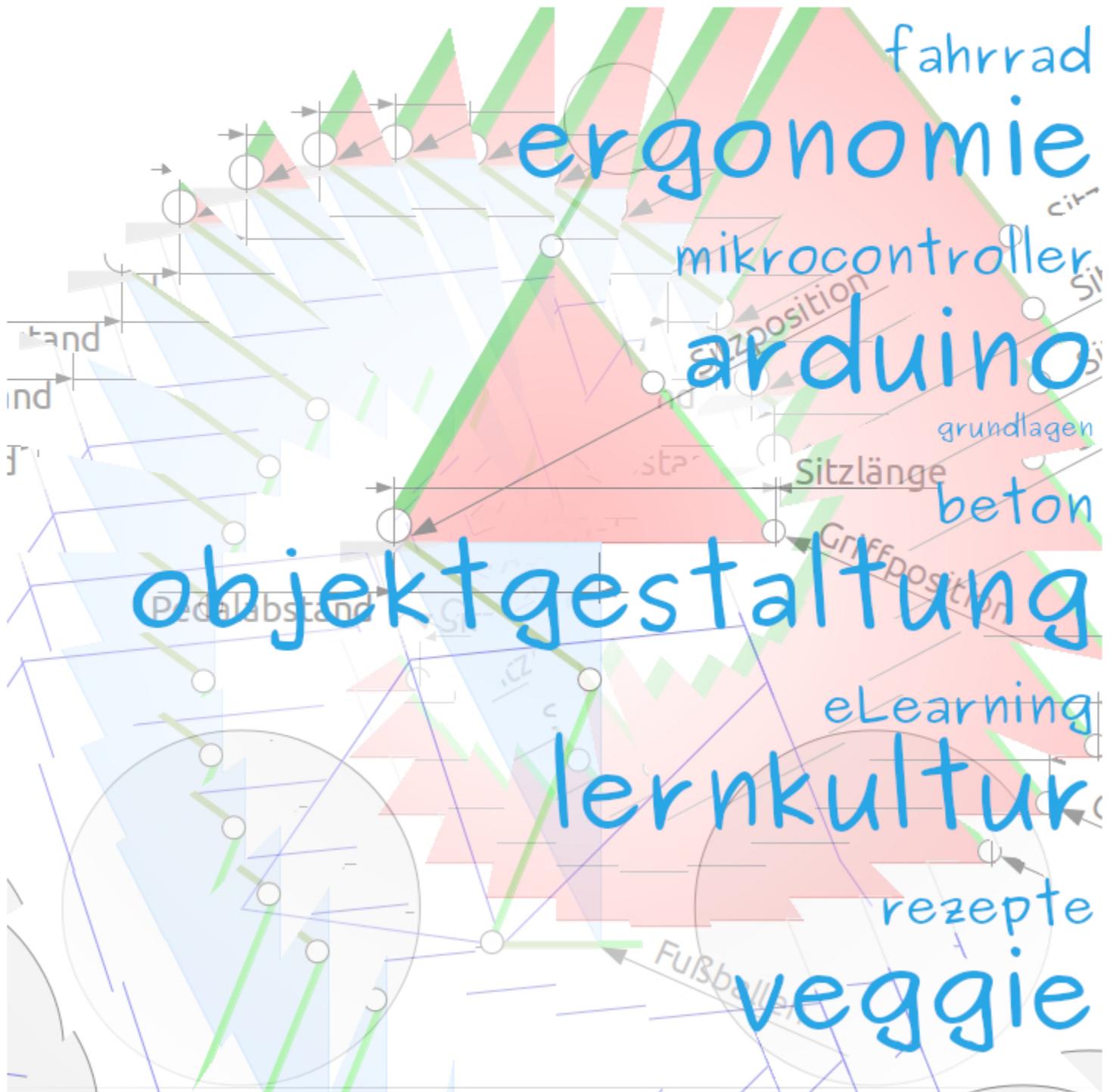


bau es selbst
best

mit

t

01.8



Wen es interessiert

Es ist die Technikbegeisterung einerseits und die Freude an präzisen Formulierungen (nicht nur) technischer Zusammenhänge andererseits, die mich zu diesem Projekt getrieben hat. Die Anfänge zum (elektronischen) Publizieren gehen auf das Jahr 1986 zurück, wie im [Impressum](#) bei [tolaris know-how](#) dokumentiert.

[bes.t](#) startet mit der Ausgabe 01.01 im Juli 2017 nach knapp 3 Jahren Vorbereitungszeit. Mit der Ausgabe 01.8 vom Februar 2022 liegt die 8. überarbeitete Version vor. Das eMag wird auch nach seiner Veröffentlichung wie eine Software ständig gepflegt. Die bunte Mischung spiegelt einen Teil meiner persönlichen Interessen wider. Im Vordergrund steht das eigene Gestalten, das Konstruieren und das Bauen. Dass [bes.t](#) für [bau es selbst](#) mit [tolaris.de](#) und nicht für [bastel ...](#) steht, ist kein Zufall.

Zu den [bes.t](#)-Projekten gibt es nicht nur Anleitungen zum Bau, sondern auch Bausätze. Das [bes.t](#)-eMag ist kostenlos, ebenso ein Teil der Inhalte bei [tolaris know-how](#). Andere Inhalte erfordern einen kostenpflichtigen Zugang oder ein kostenpflichtiges eBook.

Das gesamte kostenpflichtige Angebot aus Zugängen, eBooks und Bausätzen findest du im WebShop bei [tolaris.de](#).

Eine Rechtfertigung zum [Du oder Sie](#) findest du mit dem Link.

Das gesamte kostenlose Angebot findest du unter [Download](#).

Zu jedem [bes.t](#)-eMag gibt es eine eigene Magazinseite, die einen kostenfreien Zugang erfordert, eine Anmeldung reicht. Für diese Ausgabe ist die Seite unter [bes.t 01](#) erreichbar:

- Nutzer: b01
- Passwort: best

Dem eMag liegen die eigenen [Konventionen](#) und der [Thesaurus](#) zugrunde. Meistens halte ich mich auch an meine eigenen Vorgaben

Wolfgang Henderkes

fahrrad: ergonomie

Wer das Fahrrad als Verkehrsmittel oder als Sport- und Freizeitgerät einsetzt, wird sich früher oder später mit dem Thema Fahrradergonomie beschäftigen. Also mit der Frage, wie das Fahrrad dem Fahrer optimal angepasst werden kann. Dazu zählen die Einstellungen für:

- die Rahmengröße
- die Sattelhöhe
- die Sattelposition
- die Lenkerform
- die Lenkerposition
- die Lenkerhöhe

Dazu zählen auch die richtige Wahl des Sattels sowie der Griffe oder des Lenkerbandes.

Der Sattel sollte je nach Sitzwinkel eine etwas breitere Sitzfläche haben als der Abstand der Sitzknochen des Fahrers. Der wird ermittelt, indem sich der Fahrer auf ein Stück Wellpappe setzt. Der Abstand der beiden Eindrücke entspricht der notwendigen Satteltbreite plus bis zu 3 cm bei aufrechter Sitzweise und 1 cm bei Rennradfahrern. Ob Gel oder Federung, Leder, Kunststoff etc. ist eine Frage des Geschmacks und des persönlichen Wohlempfindens. Auch eine Frage des Radtyps: der Rennradfahrer bevorzugt einen anderen Satteltyp als der Fahrer eines Stadtrades.

Griffe mit Handballenunterstützung zwingen die Hand bei richtiger Einstellung in eine Position, die vom Handrücken eine Gerade mit dem Unterarm bildet. In Kombination mit Hörnchen kann die Griffposition gewechselt werden, um ein Ermüden zu verhindern. Auch hier spielt der Radtyp eine Rolle: der Rennradfahrer wird Lenkerbänder statt Griffe verwenden.

In diesem Beitrag geht es jedoch nur um die richtige Einstellung der Komponenten.

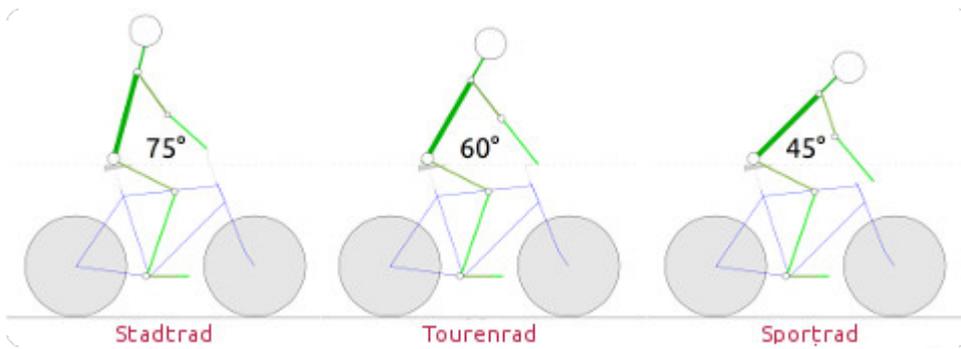
Eine falsche Haltung führt zu Problemen wie Kribbeln, Taubheitsgefühlen, Verspannungen oder Schmerzen. Mit der richtigen Einstellung der vorhandenen Komponenten oder ggf. dem Austausch einzelner Komponenten lässt sich das Fahrrad mit keinem oder geringen Aufwand richtig anpassen.

Maßfahrrad oder Maßrahmen?

Eine Frage, die sich allenfalls bei einer Neuanschaffung stellt. Und ein vorhandenes Fahrrad – kann das an den Fahrer angepasst werden? Die Antwort lautet fast immer „ja“, sofern der Fahrer nicht länger als 190 cm ist und normale Körperproportionen besitzt. Kopffüßler sind also ausgenommen.

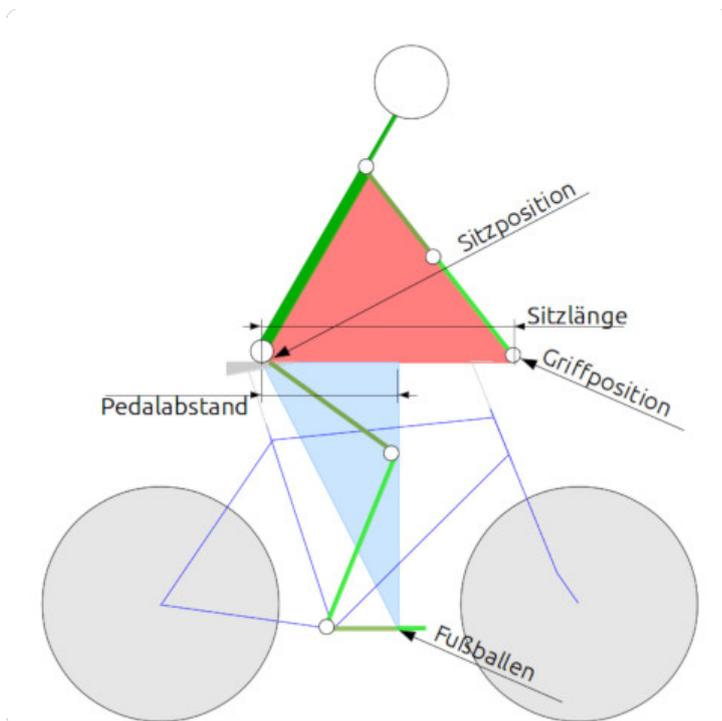
Die erste Frage auf dem Weg zum Maßfahrrad lautet: wie will ich sitzen? In [§12](#) sind drei typische Sitzpositionen gezeichnet. Das Stadtrad hat eine eher aufrechte Haltung mit einem Sitzwinkel von ca. 75° – die Griffposition liegt über der Sitzposition. Bei dieser Haltung besteht die Gefahr eines Rundrückens. Das Tourenrad mit einem Sitzwinkel von 60° gilt als ergonomisch.

Rennradfahrer haben einen kleineren Sitzwinkel und die Griffposition liegt unter der Sitzposition. Diese Haltung ist dem geringeren Windwiderstand geschuldet, führt früher oder später aber zu Problemen im Nackenbereich. Kurz: wenn möglich, sollte das Fahrrad als Tourenrad zum Maßfahrrad angepasst werden.



☺12 Sitzwinkel der verschiedenen Radtypen.

Und das wird über das Oberkörper- und das Beindreieck definiert.



☺13 Oberkörper- und Beindreieck.

In ☺13 sind die beiden Dreiecke rot und blau gezeichnet. Interessant sind darin die Sitzlänge, der Pedalabstand und die Sitzposition, deren Abstand zum Boden die Sitzhöhe bestimmt. Die wird erzielt über die Kombination aus Sitzrohr und Sattelstütze – die häufig genannte Rahmenhöhe ist nicht die wichtigste Größe.

Die Rahmenhöhe ist kein genormter Begriff, viele Hersteller verstehen darunter etwas anderes. Und: wichtiger als die Rahmenlänge ist die Sitzlänge – in dem Abstand aus Sitz- und Griffposition ist die Rahmenlänge nur eine Größe.

Anders ausgedrückt ist es möglich, einen Rahmen anzupassen mit Hilfe der richtigen:

- Sitzhöhe durch eine entsprechend lange Sattelstütze
- Sitzposition durch die passende Kombination und Ausrichtung von Sattel und Sattelstütze
- Griffposition durch die passende Kombination und Ausrichtung von Vorbau und Lenker

Fazit: ein kleiner Rahmen kann durch die Anpassung „vergrößert“ und ein großer Rahmen „verkleinert“ werden.

Bei sportlich ambitionierten Rennradfahrern und bei MTBs stößt die Anpassung an ihre Grenzen. Rennradfahrer werden einen passenden Rahmen benötigen, um sportliche Leistungen zu erbringen. Bei MTBs können starke Steigungen und Gefälle dazu führen, dass ein angepasster „vergrößerter“ Rahmen nach hinten oder nach vorne kippt.

Bei normalen Fahrrädern mit oder ohne Motor ist ein Maßrahmen nur selten erforderlich. Der Weg zum Maßfahrrad führt entweder über das Ausprobieren, also die Montage und Ausrichtung der fraglichen Komponenten oder über die Vermessung des Ichs und der mathematischen Ermittlung des Maßfahrrads.

Um die eigenen Körpermaße zu erfassen, hat **tolaris know-how** das sogenannte Egometer entwickelt.

Egometer

Egometer – das gibt's ja gar nicht!

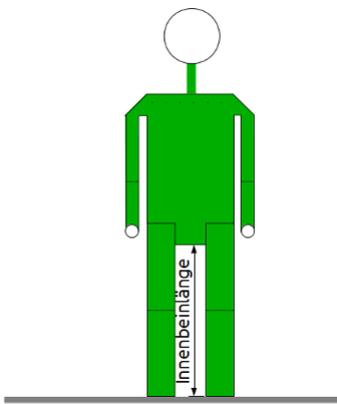
Zumindest nicht als Begriff in der deutschen Sprache. Das **tolaris know-how** Egometer ist ein Verfahren zur Vermessung des Ichs, allerdings in rein physischer Form. Das Ziel: ein Maßfahrrad – und nicht ein Maßrahmen. Benötigte Hilfsmittel: 1 Maßband und 1 Buch. Das Buch bzw. eine Buchkante dient zur Bestimmung der einen Körperposition, die andere wird durch eine Tür, Lehne etc. ermittelt.

💡 Deine Messungen trägst du in das [Formular](#) ein!

Die blauen Größen in den Grafiken werden aus deinen Angaben errechnet. Die folgenden Schritte beschreiben die einzelnen Messungen.

Messungen

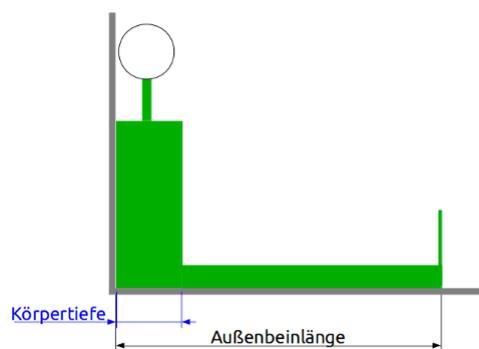
👉 Messung der Innenbeinlänge:



🚲14 Messung der Innenbeinlänge.

1. 🔑 Lege dich auf den Rücken mit den Füßen an der Tür.
2. 🔑 Schiebe das Buch bis zum Anschlag am Schambein.
3. 🔑 Messe den Abstand Buchkante ⇔ Tür.
4. 🔑 Notiere die Innenbeinlänge.

👉 Messung der Außenbeinlänge:

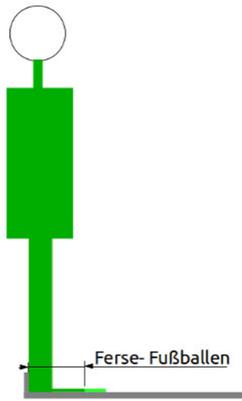


🚲15 Messung der Außenbeinlänge.

1. 🔑 Setze dich auf den Boden.
2. 🔑 Schiebe den Po und den Rücken an die Tür, halte die Füße aufrecht.
3. 🔑 Positioniere am Fußende das Buch.
4. 🔑 Messe den Abstand Buchkante ⇔ Tür.
5. 🔑 Notiere die Außenbeinlänge.



Messung Ferse ⇔ Fußballen:

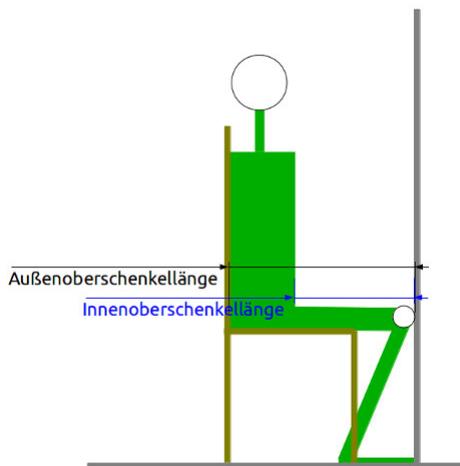


1.  Positioniere die Ferse an der Tür.
2.  Positioniere das Buch unter dem Fußballen.
3.  Messe den Abstand Tür ⇔ Buchkante und notiere den Wert.

🚲16 Messung Ferse ⇔ Fußballen.



Messung der Außen Oberschenkel länge:

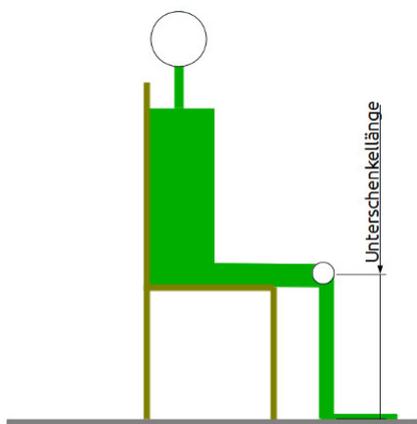


1.  Setze dich auf einen Stuhl, den Rücken aufrecht an der Lehne.
2.  Schiebe den Stuhl zur Tür, bis das Knie die Tür berührt.
3.  Messe Abstand Tür ⇔ Lehne.
4.  Notiere die Außen Oberschenkel länge.

🚲17 Messung der Außen Oberschenkel länge.



Messung der Unterschenkel länge:

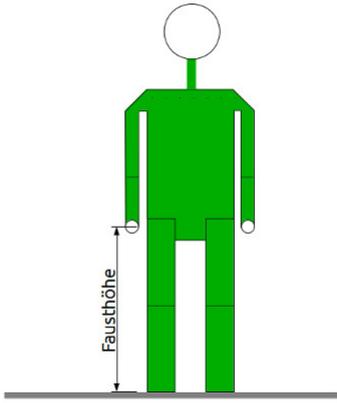


1.  Setze dich auf einen Stuhl, halte die Unterschenkel vertikal.
2.  Messe den Abstand Mitte Kniescheibe ⇔ Boden.
3.  Notiere die Unterschenkel länge.

🚲18 Messung der Unterschenkel länge.



Messung der Fausthöhe:

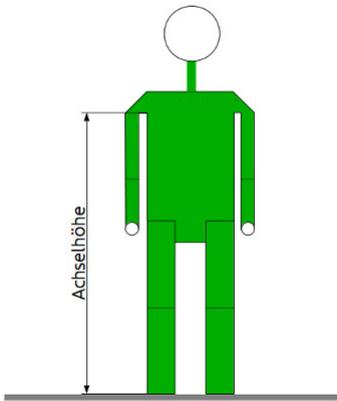


1.  Lege dich auf den Rücken mit den Füßen an der Tür.
2.  Lege die Hände parallel zum Körper.
3.  Positioniere das Buch in einer geöffneten Hand.
4.  Messe den Abstand Buchkante ⇔ Tür.
5.  Notiere die Fausthöhe.

🚲19 Messung der Fausthöhe.



Messung der Achselhöhe:

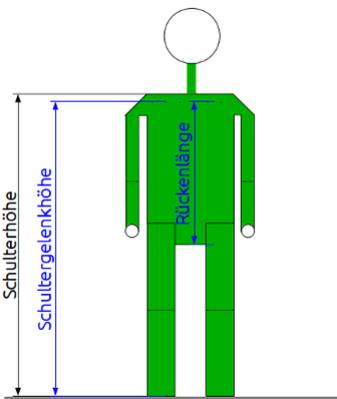


1.  Lege dich auf den Rücken mit den Füßen an der Tür.
2.  Lege die Hände parallel zum Körper.
3.  Positioniere das Buch in einer Achsel.
4.  Messe den Abstand Buchkante ⇔ Tür.
5.  Notiere die Achselhöhe.

🚲10 Messung der Achselhöhe.



Messung der Schulterhöhe:

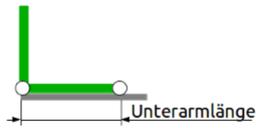


1.  Lege dich auf den Rücken mit den Füßen an der Tür.
2.  Lege die Hände parallel zum Körper.
3.  Positioniere das Buch auf Schulterhöhe.
4.  Messe den Abstand Buchkante ⇔ Tür.
5.  Notiere die Schulterhöhe.

🚲21 Messung der Schulterhöhe.



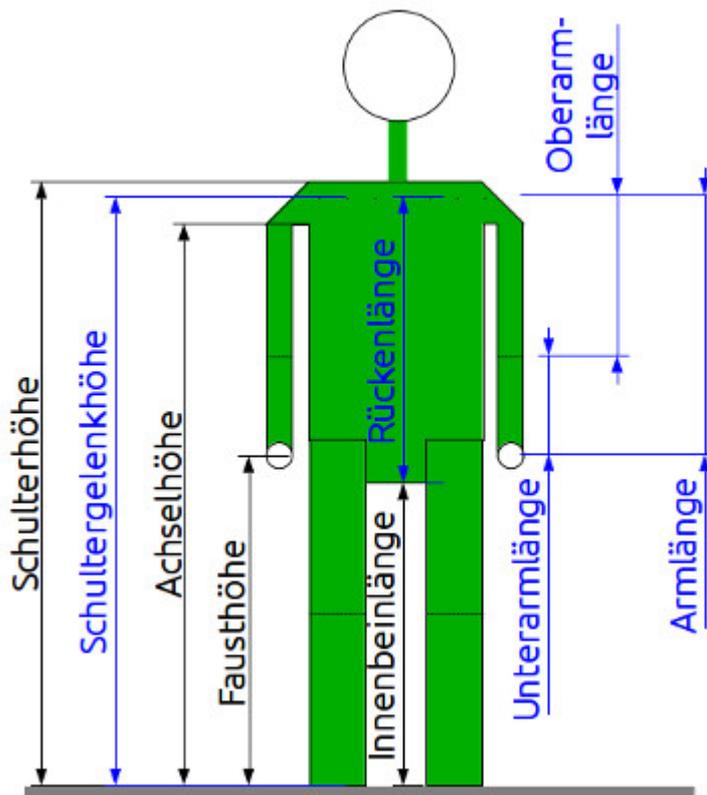
Messung der Unterarmlänge:



1.  Lege den Unterarm auf einen Tisch.
2.  Positioniere den Ellenbogen über der Tischkante.
3.  Halte das Buch in einer geöffneten Hand.
4.  Messe den Abstand Buchkante \leftrightarrow Tischkante.
5.  Notiere die Unterarmlänge.

 Messung der Unterarmlänge.

Übersicht der Messungen



Wiederhole alle Messungen und vergleiche sie mit den vorherigen Werten, bis die Differenz ± 1 cm beträgt.

Maßfahrrad oder Maßrahmen

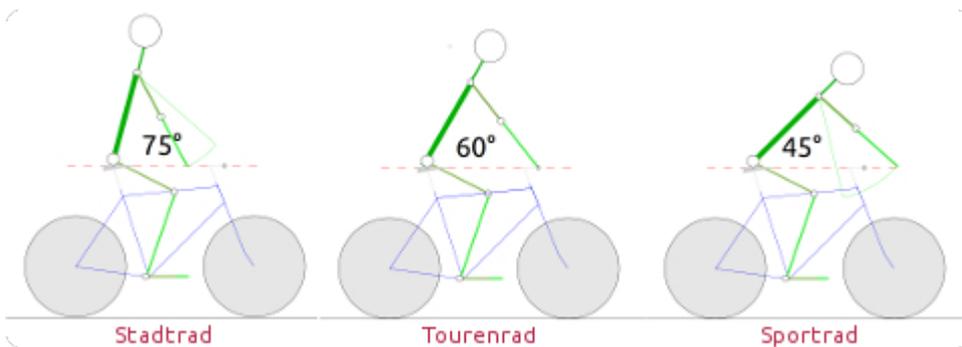
Das Ergebnis ist nicht die Angabe einer Rahmenhöhe, sondern die Berechnung deines Oberkörper- und Beindreiecks. Darin sind die wichtigsten Größen:

1. die „Sitzlänge“ in Abhängigkeit des Sitzwinkels
2. der Pedalabstand in Abhängigkeit deiner Proportionen
3. der erzielbare Pedalabstand in Abhängigkeit des Sitzrohrwinkels und der Kurbellänge

Der Sitzwinkel ist die Größe, die du frei bestimmen kannst. Die 60° sind ein Vorschlag, der sich an den Erkenntnissen der Fahrradergonomie orientiert: diese moderate Haltung ist für den Körper gut geeignet und gilt als ergonomisch. Die 60° sind ein Richtwert, kein Dogma. Eine aufrechtere Sitzhaltung kann ebenso "gut" sein, wenn der Rücken gerade gehalten wird. Eine Haltung wie auf dem Rennrad ist nicht mehr ergonomisch, für den Sportler jedoch Voraussetzung, um hohe Geschwindigkeiten zu erreichen.

☞24 zeigt drei Beispiele, welche Haltung aus dem Sitzwinkel folgt:

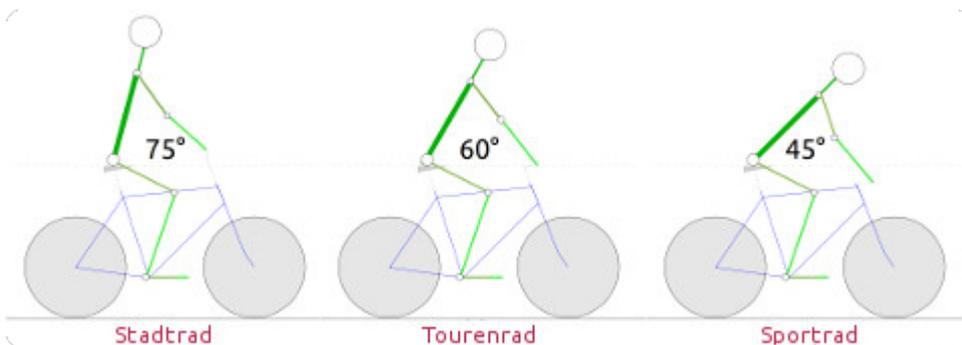
1. Stadtrad mit einem Sitzwinkel von 75°
2. Tourenrad mit einem Sitzwinkel von 60°
3. Sportrad mit einem Sitzwinkel von 45°



☞24 Sitzwinkel der verschiedenen Radtypen mit der Sitz- und Griffposition auf einer Horizontalen.

Du musst also wissen, wie du bzw. welchen Radtyp du fahren möchtest. Aus dem gewünschten Sitzwinkel und deinen Proportionen folgt die Sitzlänge, die mit deiner Rücken- und der Armlänge dein Oberkörperdreieck bildet.

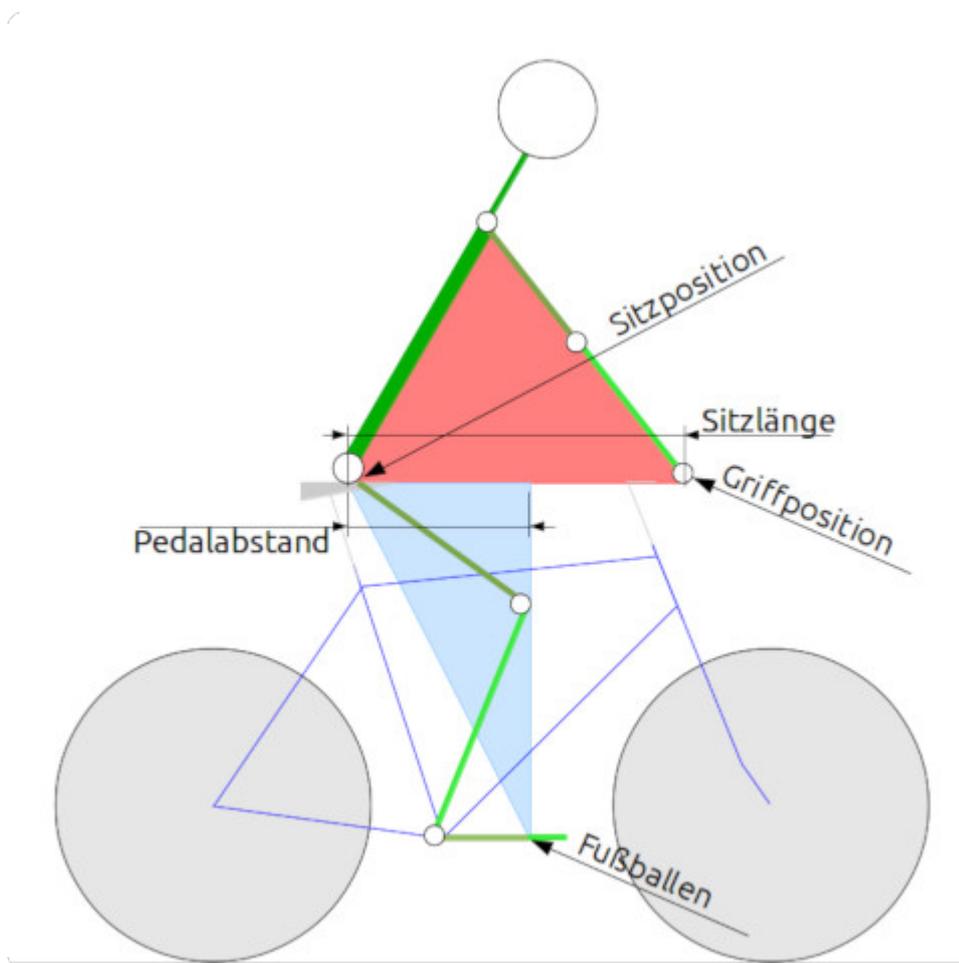
Verändere den Sitzwinkel und starte jeweils eine neue Berechnung. Dann bekommst du ein Gespür dafür, welche Auswirkungen damit verbunden sind. Grundlage der berechneten Sitzlänge ist die Annahme, dass Sitz- und Griffposition wie in ☞24 auf einer Horizontalen liegen. Diese Annahme ist ein Bezugspunkt, um unterschiedliche Sitzwinkel zu betrachten.



☞12: Sitzwinkel der verschiedenen Radtypen mit realistischer Sitz- und Griffposition.

In der Realität werden bei den Radtypen Stadtrad und Sportrad die Griff- und Sitzposition nicht auf einer Horizontalen liegen und der Ellenbogen wird gewinkelt sein, wie in ☞12 dargestellt.

Der Begriff Sitzlänge (und nicht Rahmenlänge) ist Absicht, weil die Sitzlänge in Abhängigkeit des Sitzwinkels den Abstand zwischen der Sitz- und der Griffposition angibt. Diese Sitzlänge kann über die Rahmenlänge beeinflusst werden, aber auch über die Kombination von Sattelstütze und Sattel sowie über die Kombination von Vorbau und Lenker.



13 Oberkörper- und Beindreieck.

Die Sitzposition bildet mit dem Fußballen bei horizontaler Kurbelposition das Beindreieck, das den Pedalabstand bestimmt. Du kannst den Sitzrohrwinkel verändern und die Auswirkungen direkt betrachten. Standardrahmen liegen in dem Bereich zwischen 72° und 74° , für eine exakte Berechnung musst du also den Winkel des Rahmens kennen. Die Kurbellänge von 17 cm (= 170 mm) ist ebenfalls Standard. Die Auswahl an kleineren Kurbeln (165 mm) und größeren Kurbeln (175 mm) ist überschaubar – Kurbeln > 175 mm sind die Ausnahme.

Diese beiden Dreiecke definieren den gewünschten Radtyp mit dem Fahrer und seinen Proportionen. Das Ergebnis soll eine Empfehlung für ein Maßfahrrad sein, nicht für einen Maßrahmen. Menschen mit normalen Körperproportionen (also keine Kopffüßler) bis 190 cm Länge werden aus dem Standardangebot von Rahmen und Zubehör ein passendes Fahrrad konfigurieren können.

Wenn du nur an dem Ergebnis interessiert bist, kannst du die Ausführungen zur Rahmengenometrie und Fahrradergonomie überspringen und bei [der richtigen Interpretation der Berechnung](#) weiterlesen. Wenn dich das Warum interessiert, dann lies bitte hier weiter...

Rahmengenometrie

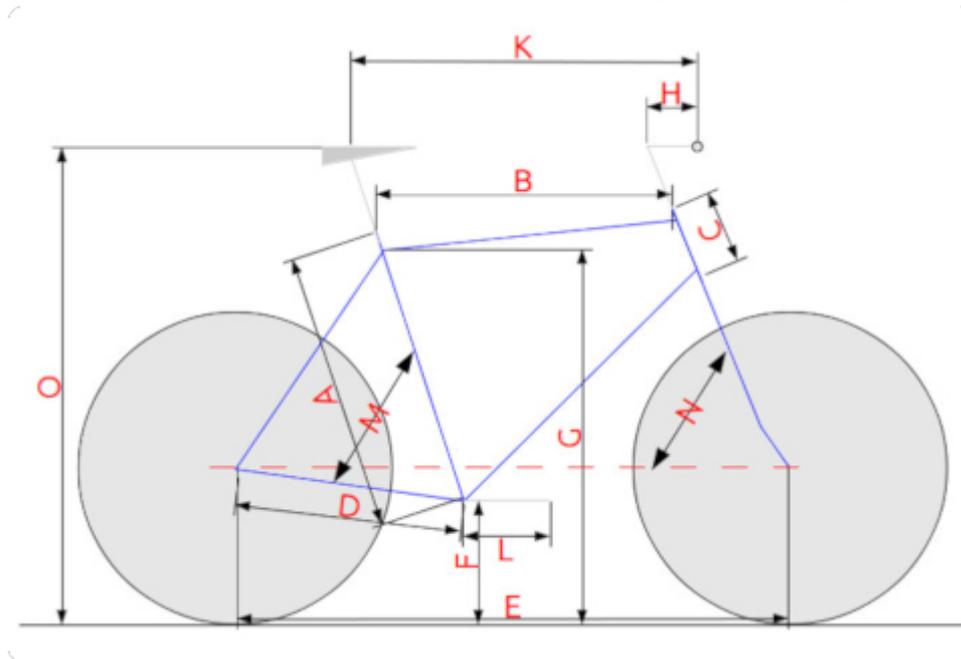
Bei den Rahmen werden in der Regel die Rahmenhöhen (auch Rahmengröße oder Sitzrohrlänge genannt) angegeben – warum verwendet **tolaris know-how** diesen Begriff nicht? Bei der Beantwortung hilft ein Blick auf die Rahmengenometrie mit einem Kommentar zu den wichtigsten Größen.

Die Rahmenhöhe ist kein genormter Begriff, viele Hersteller verstehen darunter etwas anderes. Selbst die Angabe der Sitzrohrlänge ist wenig aussagekräftig: ein zu kurzes Sitzrohr wird bei Bedarf mit einer Sattelstütze verlängert – und ergibt damit die korrekte Sitzhöhe. In $\phi 25$ ist die Sitzrohrlänge (A) definiert als Abstand Tretlagermitte bis Rohrende – aber wie bereits geschrieben: diese Definition ist nicht genormt.

Wichtig ist die erzielbare Sitzhöhe (O) aus Sitzrohr und Sattelstütze, nicht die Rahmenhöhe.

Eine wichtige Rahmeneigenschaft ist jedoch das Oberrohr, falls vorhanden. Und ob es horizontal verläuft oder geneigt ist. Denn der Abstand Rohroberseite – Boden sollte an der Standposition (also etwas vor dem Sattel) kleiner als die Innenbeinlänge sein – damit der Fahrer bequem stehen kann.

In $\phi 25$ ist die Oberrohrhöhe (G) als der Abstand vom Boden bis zum Schnittpunkt des Oberrohrs und Sitzrohrs definiert. Diese Definition ist zwar praxisgerecht – hängt jedoch nicht nur von der Rahmen-



geometrie ab, sondern auch von der Bereifung: ein breiterer Reifen ist höher und damit steigt die Oberrohrhöhe.

Ob das Oberrohr gewinkelt ist oder nicht und ob es überhaupt vorhanden ist oder nicht – für die Rahmengeometrie ist die Rahmenlänge (B) interessant. Und die lässt sich auch ohne Oberrohr ermitteln.

$\phi 25$ Rahmengeometrie.

Größen der Rahmengeometrie in $\phi 25$:

- A = Sitzrohrlänge
- B = Rahmenlänge
- C = Steuerrohrlänge
- D = Kettenstrebenlänge
- E = Radabstand
- F = Tretlagerhöhe
- G = Oberrohrhöhe
- H = Vorbaulänge
- K = Sitzlänge
- L = Kurbellänge
- M = Sitzrohrwinkel
- N = Steuerrohrwinkel
- O = Sitzhöhe

Anders ausgedrückt ist es möglich, einen Rahmen anzupassen mit Hilfe der richtigen:

1. Sitzhöhe durch eine entsprechend lange Sattelstütze
2. Sitzposition durch die passende Kombination und Ausrichtung von Sattel und Sattelstütze
3. Griffposition durch die passende Kombination und Ausrichtung von Vorbau und Lenker

Durch die Sattelstütze wird der Rahmen „höher“, durch die Sitz- und Griffposition „länger“ oder „kürzer“.

Wenn mit diesen Maßnahmen der Rahmen nicht angepasst werden kann, sollte ein Rahmen mit einem anderen Sitzwinkel gesucht werden. Wenn der nicht lieferbar sein sollte, macht die Anfertigung des Maßrahmens Sinn.

Die Kettenstrebendlänge (D) und der Steuerrohrwinkel (N) beeinflussen neben der Rahmenlänge den Radabstand (E) – und der hat Auswirkungen auf das Fahrverhalten:

1. mit einem kleineren Radabstand ist das Fahrrad wendiger
2. mit einem längeren Radabstand hat das Fahrrad bessere Geradeauslaufeigenschaften

Industriell hergestellte Rahmen werden in der Regel in verschiedenen Größen angeboten. Für die mir bekannten Rahmenbaureihen gilt: eine größere Rahmenlänge bedeutet ein längeres Sitzrohr. Anders ausgedrückt ist ein längerer Rahmen auch größer. Ein größerer Rahmen verlangt eine kürzere Sattelstütze und eine kleinere Vorbaulänge. Bei einigen Rahmenbaureihen ist der Sitzrohrwinkel bei allen Größen gleich, bei anderen wird der Sitzrohrwinkel kleiner mit wachsender Rahmengröße. Der Sitzrohrwinkel wird bei Maßrahmen den Proportionen des Fahrers angepasst, damit der Pedalabstand stimmt.

Die Tretlagerhöhe (F) ist einerseits bedingt durch den für den Rahmen vorgesehenen Felgendurchmesser, verändert sich andererseits mit der Reifenhöhe. Anders ausgedrückt sollten (und meistens können) auf einem Rahmen für 622-er Felgen (28") keine 559-er Felgen (26") gefahren werden. Mit breiteren = höheren Reifen kann aus einem 28"-er jedoch ein 29"-er werden, sofern der Reifen noch genügend Platz hat.

Für die meisten Fahrer ist die Tretlagerhöhe jedoch nicht interessant. Für Fahrer über 190 cm Länge ist evtl. eine größere Kurbellänge (L) als 170 mm erforderlich. Damit die Kurbel in Kurven keinen Bodenkontakt bekommt, ist evtl. ein höher liegendes Tretlager erforderlich oder eben ein höherer Reifen.

Die Vorbaulänge (H) entsteht aus der Kombination von Vorbau und Lenker und bestimmt die Griffposition. Die Vorbaulänge ist in [§25](#) als "Rahmenverlängerung" ausgeführt, die Griffposition verschiebt sich bei einem geraden Lenker weiter nach vorne. Ein kurzer Vorbau kombiniert mit einem Stuttgarter oder Toulouse Lenker wirkt dagegen als "Rahmenverkürzung". Die Vorbaulänge kann demnach einen positiven oder negativen Wert annehmen.

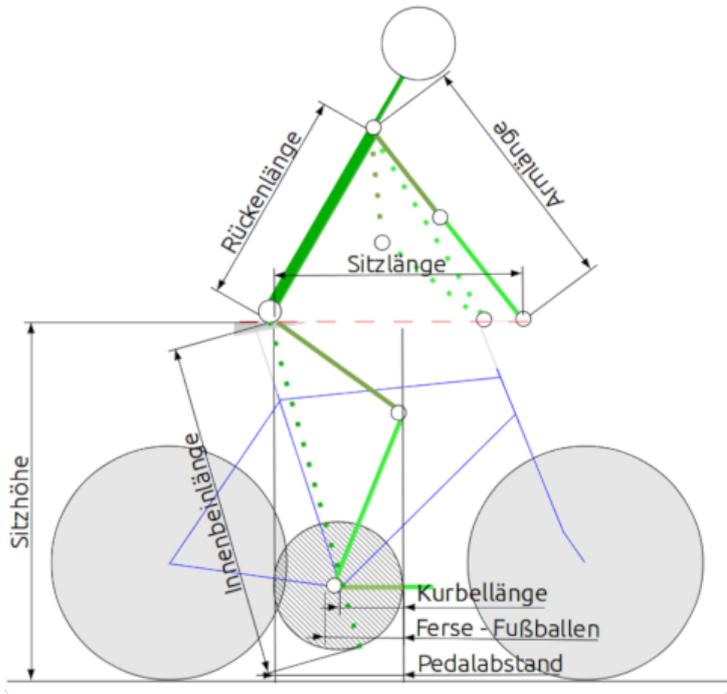
Ausschlaggebend für die Sitzlänge (K) sind die Sitzposition, die Rahmenlänge und die Griffposition. Die Sitzposition ist in [§25](#) der Schnittpunkt der Sitzrohrgeraden mit der Satteldecke. Die Sitzposition kann jedoch erheblich wie in [§26](#) davor oder dahinter liegen.

Der hier gezeigte Diamantrahmen dient als Beispiel, weil er weit verbreitet ist. Die Rahmengenometrie ist jedoch nicht an eine Rahmenform gebunden, sondern gilt universell. Bis auf die Oberrohrhöhe verfügt jede Rahmenform über die in [§25](#) angegebenen Werte.

Fahrradergonomie

Da die Maschine dem Menschen dienen soll, wird die optimierte Anordnung der verwendeten Komponenten betrachtet: die Fahrradergonomie. Dazu zuerst einige Definitionen der verwendeten Begriffe:

1. Die Rückenlänge stammt aus der Vermessung des Ichs.
2. Ebenso die Armlänge, die bei gestrecktem Arm gemessen wird. In der Praxis ist der Arm kaum durchgestreckt, der Ellenbogenwinkel ist also $< 180^\circ$ – bei der Tourenradposition weniger, bei der Stadt- und Sportposition mehr. Je nach Ellenbogenwinkel ist die "nutzbare" Armlänge kleiner, dargestellt durch die gestrichelte Linie.
3. Die Sitzlänge folgt aus der Rücken- und Armlänge sowie dem Sitzwinkel. [§26](#) zeigt die Auswirkung eines gewinkelten Ellenbogens auf die Sitzlänge.
4. Der Pedalabstand entspricht dem Abstand von der Sitzposition bis zur Pedalachse mit der Kurbel in der 3 Uhr Position. Auf der Pedalachse sollte der Fußballen (!) liegen. Der richtige Pedalabstand ist erreicht, wenn die Kniescheibe senkrecht über der Pedalachse liegt, wie in [§26](#) zu sehen.
5. Die Kurbellänge ist mit ihrem Durchmesser durch den Kreis dargestellt.
6. Der Abstand Ferse - Fußballen ist ca. 1/3 kleiner als die Fußlänge.
7. Als Sitzposition gilt die Position des Schambeins, also das obere Ende der Innenbeinlänge.
8. Das untere Ende der Innenbeinlänge entspricht der Kurbelposition im Sitzrohrwinkel mit der der Ferse (!) auf dem Pedal. Die Innenbeinlänge ist durch die gestrichelte grüne Linie am Sitzrohr dargestellt.



☛26 Ergonomie am Tourenrad.

☛26 zeigt einen Fahrer in der Tourenradposition mit einem Sitzwinkel von 60° .

Die Sitz- und Griffposition sind auf gleicher Höhe.

Die Knievorderseite ist im Lot über der Pedalachse.

Der Arm ist gestreckt und annähernd im rechten Winkel zum Rücken.

Diese Haltung gilt bei Sportwissenschaftlern als ergonomisch. Auch eine Griffposition über der Sitzposition gilt als ergonomisch. Als besonders empfehlenswert gelten Touren-Lenker, die unterschiedliche Griffpositionen und -höhen erlauben.

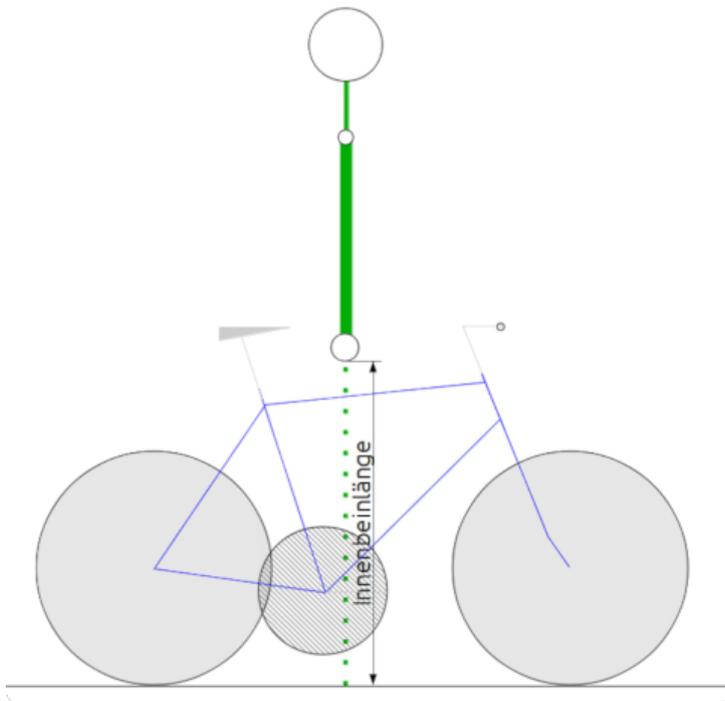
Für die Ergonomie ist nicht nur das Oberkörperdreieck mit der Sitzlänge wichtig, sondern ebenso das Beindreieck für den Pedalabstand. Das wird beeinflusst durch:

1. die Sitzhöhe
2. den Sitzrohrwinkel
3. die Sattelposition
4. die Ausrichtung und Form der Sattelstütze

Die Sitzhöhe resultiert aus der Innenbeinlänge, sie sollte nur gering verändert werden und taugt zur Einstellung des Pedalabstands wenig.

Der Sitzrohrwinkel ist durch den Rahmen vorgegeben. Viele Rahmenhersteller verbauen das Sitzrohr mit einem Winkel zwischen 72° und 74° . Häufig wird der Rahmen in verschiedenen Größen (meistens als Rahmenhöhe bezeichnet) angeboten. Je nach Rahmen und Hersteller ist der Sitzwinkel bei allen Rahmenhöhen gleich, oder der Winkel wird kleiner mit der Rahmenhöhe. (Und damit der Pedalabstand größer, was den längeren Beinen längerer Menschen entgegenkommt.)

Die Auswirkungen sind überschaubar: bei einer Innenbeinlänge von 90 cm verändert sich der erzielbare Pedalabstand zwischen 72° und 74° nur um ca. 2,5 cm.



🚲27 Ergonomie auch stehend.

Das Sattelgestell erlaubt eine Verschiebung der Sattelposition. Je nach Hersteller und/oder Sattelkonstruktion lässt sich die Sitzposition um 3,5 cm verschieben. Durch die geeignete Sattelstütze kann die Sitzposition um weitere 8 cm verschoben werden – zusammen also 11,5 cm.

Da mit Ausnahme von Parallelogrammsattelstützen die Sattelbefestigung nach vorne oder nach hinten zeigend eingebaut werden kann, beträgt der Einstellungsbereich $\pm 11,5$ cm.

Längere Kurbeln (von 170 mm auf 175 mm) bringen für den Pedalabstand nicht viel. Noch längere Kurbeln sind einerseits kaum zu bekommen, andererseits führen sie in Kurven eher zu einem ungewollten Bodenkontakt.

Die richtige Interpretation der Berechnung

Da die Berechnung keine Rahmengröße sondern zwei Körperdreiecke liefert, soll die Interpretation anhand der bereits verwendeten drei Sitzwinkel erfolgen. Bei den errechneten Sitzlängen sind die Sitz- und Griffposition auf einer horizontalen Geraden mit gestrecktem Arm. Im Idealfall sollte der Oberarmwinkel 90° betragen. In den Grafiken unten wäre dieser Fall in einem Sitzwinkel zwischen Touren- und Sportposition erreicht. Daraus folgt nicht, dass alle anderen Haltungen nicht eingenommen werden sollten. Die Griffposition kann durch einen gewinkelten Ellenbogen angepasst werden.

Die Grafiken zeigen einen Fahrer mit seinen Proportionen und einen Rahmen in maßstabsgerechter Darstellung mit den möglichen Einstellungsbereichen.

Fahrer:

1. Rückenlänge: 59 cm
2. Armlänge: 64,5 cm
3. Innenbeinlänge: 91 cm
4. Unterschenkellänge: 53 cm
5. Innenoberschenkellänge: 41 cm
6. Ellenbogenwinkel: 180°

Beispielrahmen:

1. Radabstand: 110 cm
2. Rahmenlänge: 58 cm
3. Sitzrohrlänge: 57 cm
4. Sitzrohrwinkel: 72°
5. Steuerrohrwinkel: 68°

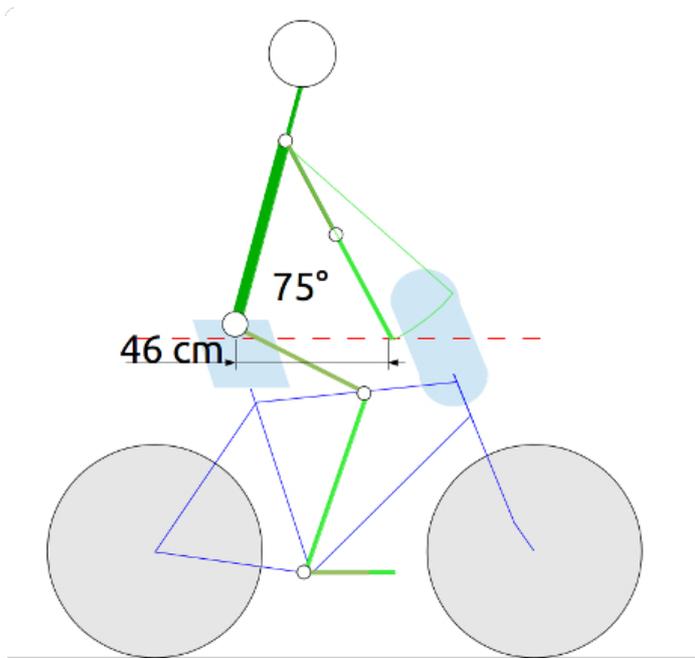
Einstellungsbereich:

1. Sitzerrhöhung: 0 - 20 cm
2. Sitzposition: $\pm 11,5$ cm
3. Grifferrhöhung: 0 - 20 cm
4. Griffposition: $\varnothing 20$ cm

Der Einstellungsbereich für die Sattelposition ist mit dem hellblauen Parallelogramm gekennzeichnet. Innerhalb des Bereichs kann jede Sitzposition eingenommen werden. Der Einstellungsbereich für die Griffposition ist ebenfalls hellblau dargestellt – auch hier kann jede Position eingenommen werden.

Voraussetzung ist in beiden Fällen die entsprechende Kombination der Komponenten.

Stadtrad



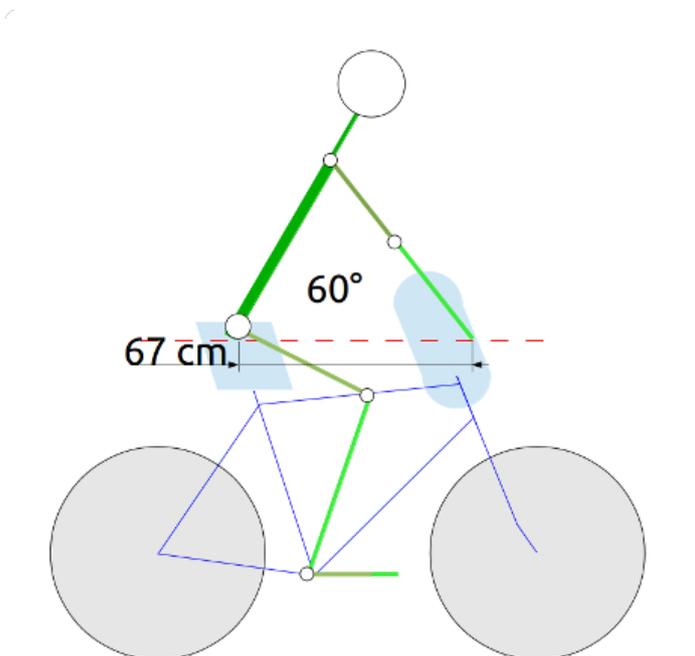
28 Sitzwinkel mit 75°.

Dieser Fahrer kann mit seiner Armlänge und dem Sitzwinkel von 75° nicht mit der Sitz- und Griffposition auf einer horizontalen Geraden fahren. Der Fahrer muss den gestreckten Arm entweder höher heben oder den Arm anwinkeln, um in den Einstellungsbereich zu gelangen.

Die höhere Griffposition passt zu der eher aufrechten Sitzweise. Beide Haltungen sind ergonomisch.

Nachteil: bei diesem Sitzwinkel neigt der Fahrer eher dazu, einen Rundrücken zu bilden.

Tourenrad



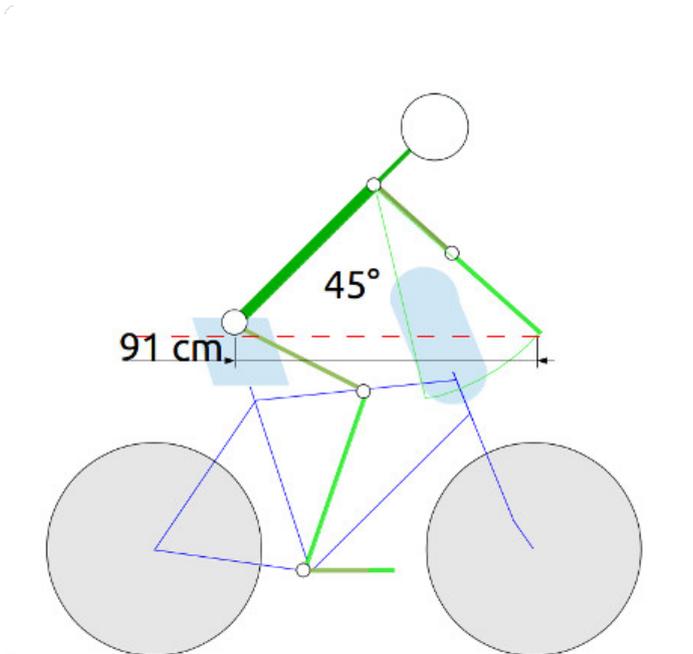
29 Sitzwinkel mit 60°.

Die Tourenradposition hat mit 60° einen moderaten Rückenwinkel, die Faust liegt fast am Steuerrohr und der Oberarmwinkel kommt dem Ideal von 90° recht nah.

Hier reicht ein kleiner Vorbau, wenn der Fahrer wirklich mit durchgestrecktem Arm fahren will. Dieser Sitzwinkel gilt als optimal.

Als ideale Ergänzung gilt ein Lenker, der verschiedene Griffpositionen erlaubt.

Sportrad



🚲30 Sitzwinkel mit 45° .

Fazit

Die Betrachtungen der drei Radtypen zeigen, wie ein Rahmen bzw. eine Rahmengröße an den Fahrer angepasst und zum Maßfahrrad wird. Der Bedarf für einen Maßrahmen wird nur in seltenen Fällen entstehen. Wem das optische Erscheinungsbild gleich ist, kann zu große Rahmen „verkleinern“ und umgekehrt.

Link

- [Formular](#)

Bei einer sportlichen Haltung liegt die Faust weit vor dem Steuerrohr, wenn die Sitz- und Griffposition auf einer Horizontalen liegen sollen. Diese Haltung ist mit Triathlon-Lenkern möglich: wer so fahren will, benötigt einen langen Vorbau.

Alternativ kann der Fahrer seinen Ellenbogen winkeln und die Faust näher am Körper positionieren. Je nach Armwinkel liegt die Griffposition unter der Sitzposition.

Nachteil: Der Sitzwinkel bedingt einen angewinkelten Kopf. Diese Haltung ist nicht ergonomisch. Langjährige Rennradfahrer berichten über Probleme im Nacken.

fahrrad: werkstatt

Bremsen

Hydraulische Bremsen sind die besten Bremsen, die es gibt. Besonders gut sind Scheibenbremsen mit möglichst großem Radius. Das ist der Tenor, wenn man dem Fachhandel und der Fachpresse Glauben schenken darf.



🚲35 Hydraulische Scheibenbremse.

Richtig ist, dass die Bremswirkung einer hydraulischen Bremse nicht zu überbieten ist. Scharf eingestellt, kann mit einem Finger am Bremshebel das Rad zum Blockieren gebracht werden. Anders ausgedrückt sollte der Fahrer das Bremsen üben, um einen Abstieg über den Lenker zu vermeiden.

Richtig ist auch, dass insbesondere hydraulische Scheibenbremsen in der Werkstatt die Schwächen deutlich aufzeigen. Wird als Bremsflüssigkeit DOT verwendet, ist besondere Vorsicht geboten. Die Praxis zeigt Fälle, in denen die Bremse total blockierte oder völlig versagte, im letzteren Fall weil Bremsflüssigkeit am Hebel ausgetreten war und den Lack weg ätzte. DOT Flüssigkeiten gibt es in verschiedenen Versionen, die nicht alle kompatibel sind. Eine falsche Flüssigkeit kann zu defekten Dichtungen führen. Bei der Arbeit mit DOT sollte Schutzkleidung (Brille und Handschuhe) getragen werden. Wer mit seinem Rad längere Touren plant, sollte sich vor der Tour Gedanken machen, wer das Rad wo reparieren kann, falls nötig.

Bremsen mit Mineralöl sind in der Handhabung weniger kritisch.

Eine typische Begleiterscheinung aller hydraulischer Scheibenbremsen ist die langsam nachlassende Bremswirkung, weil Luft im System ist. Entlüften ist in der Werkstatt keine große Aufgabe, unterwegs wegen der benötigten Teile unschön – vorsichtig formuliert.

Nervig wird es, wenn die Scheibenbremse quietscht. Dafür gibt es mehrere Ursachen, wie:

- verglaste Bremsbeläge
- verformte Bremsscheiben
- Schwingungen am Rad

Die Werkstattpraxis zeigt, hydraulische Felgenbremsen machen weniger Ärger als hydraulische Scheibenbremsen. Auch die Bremswirkung hydraulischer Felgenbremsen ist über jeden Zweifel erhaben. Die Anzahl deren Anbieter ist allerdings übersichtlich, in Deutschland ist Magura mehr oder weniger alleine vertreten. Wer unterwegs einen Defekt hat, steht vor den gleichen Problemen wie bei den Scheibenbremsen.

Warum also überhaupt Scheibenbremsen?

In den Fällen, wo die Fahrt zu einer starken Verschmutzung der Felgen führen kann, z.B. am Strand oder im matschigen Gelände. Ob es eine hydraulische Bremse sein muss, ist aber eine andere Frage. Eine Alternative sind die mechanischen Scheibenbremsen. Deren Züge und Bremsbeläge können auch auf langen Touren mitgeführt werden. Mit dem entsprechenden Wissen über die Montage bleibt der Fahrer unabhängig von der Werkstatt.

Hydraulische Bremsen sind dort erforderlich, wo ein zu langer Bremszug die Bremswirkung reduziert. Z.B. am Hinterrad eines Tandems. Und dann würde ich einer hydraulischen Felgenbremse den Vorzug geben.



🚲34 Mechanische Felgenbremse, hier als V-Brake.

Ansonsten empfehle ich mechanische Felgenbremsen am Fahrrad. Moderne Zangenbremsen oder V-Bremsen erzielen eine völlig ausreichende Wirkung, sofern der Fahrer mit der Hand eine normale Kraft entfalten kann. Bremsbeläge und Züge kosten wenig, sind leicht zu beschaffen und können ebenso leicht selbst montiert werden. Einziger Nachteil: Felgenbremsen führen zu einem Verschleiß der Bremsflanke, sodass die Felge nach der Abnutzung ersetzt werden muss.

mikrocontroller: arduino grundlagen

Erste Schritte

 Bitte beachte die **Konventionen** dieser Dokumentation und die zusätzlichen Abkürzungen und Bezeichnungen:

- Basisverzeichnis ist das Verzeichnis des Terminals nach dem Start
- Installationsverzeichnis ist das Verzeichnis der Arduino-App

 Grundkenntnisse im Umgang mit einem Computer.

 Ein Computer und ein USB-Kabel. Außerdem folgende Teile, die in dem [tolaris.de | Bausatz Arduino 1](#) enthalten sind:

- Abstandssensor (Ultraschallsensor)
- Arduino Uno
- Drahtbrücken
- LEDs
- LED-Stick
- Messgerät
- Steckbrett
- Taster
- Widerstände

 Einstieg in die Programmierung von Arduino.

 Anfänger von 10-100.

Testsystem

- Computer: x86 64 Bit
- Betriebssystem: Linux Mint 20.3 Una - Mate (kurz Mint)
- Browser: Opera 83.0 / Firefox 96
- Nutzer: [tolaris](#)

 [tolaris](#) ist der Name des in dieser Dokumentation verwendeten Nutzers, den du durch deinen Nutzernamen ersetzen musst.

 Das Ausführen der Links in das **Programmierung: Lexikon** von [tolaris know-how](#) erfordert einen kostenpflichtigen Zugang, der in dem Bausatz enthalten ist.

Wer sich noch nie mit Mikrocontrollern beschäftigt hat, verspürt vermutlich eine Hemmschwelle, den Anfang zu wagen. Programmieren gilt vielen als ein Buch mit sieben Siegeln. Zu Unrecht: Programmieren ist (nur) eine formale Logik, die in diesem Fall eher einfach ist. Ein Vergleich? Gitarre spielen zu lernen ist (für mich) schwieriger. Diese Dokumentation richtet sich an Anfänger, die neugierig sind und Spaß daran haben, etwas Neues zu probieren. Mit den Eigenbauprojekten von [tolaris know-how](#) gelingt der spielerische Einstieg. Das Ziel ist der kreative Umgang mit der Technik. Als Mikrocontroller kommt hier der Arduino Uno zum Einsatz. Der Uno ist ein Mitglied einer Baureihe mit unterschiedlichen Ausführungen. Wer den Umgang mit dem Uno erlernt hat, kann auch mit den anderen Arduinos umgehen: die Programmiersprache ist die gleiche. Programme für Arduinos werden Sketches genannt.

Wer Programmieren erlernt hat, findet sich auch in einer anderen Hardware (z.B dem Raspberry) und einer anderen Programmiersprache leichter zurecht. Ziel dieses [bes.t-Projektes Arduino Uno 1](#) ist nicht der Einsatz möglichst vieler Sensoren und anderer elektronischer Bauelemente, sondern der Einstieg in die Programmierung und in die Elektronik. Wer Spaß an der Elektronik hat, wird weiter machen. Wer

nicht weiter machen will oder kann, erhält mit diesem Bausatz trotzdem eine dauerhaft bleibende und überall einsetzbare Anwendung in Form des LED-Sticks als Leuchte mit einem Näherungssensor als Schalter.

 Alle Schaltungen funktionieren mit dem Steckbrett und den Steckverbindungen ohne Löten.

Die Reihenfolge der Beispiele in diesem **bes.t**-Projekt ist kein Zufall: als Anfänger solltest du dem roten Faden folgen, also die Reihenfolge einhalten. Grundsätzliche Handhabungen sind nur einmal erklärt.

Auf dem Computer muss ein Programm laufen, das die Verbindung zwischen dem Computer und dem Mikrocontroller herstellt. Dieses Programm ist eine Entwicklungsumgebung, Arduino-IDE genannt. Zur IDE gehört ein Editor, eine einfache Textverarbeitung, die das Programm – Sketch genannt – für den Mikrocontroller anzeigt.

Anders ausgedrückt ist die IDE ein Programm auf dem Computer, das zur Erstellung von Programmen für den Mikrocontroller dient.

 Wenn du die IDE nicht installieren willst, kannst du den Webeditor nutzen, der unter <https://create.arduino.cc/> zur Verfügung steht und ebenfalls lokal installiert wird. Die Nutzung setzt einen Zugang voraus. In dieser Dokumentation wird die IDE benutzt.

 Wenn du die IDE bereits installiert hast, lies bitte trotzdem den folgenden Abschnitt, weil dort die Nutzung des Verzeichnisses für Arduino-Programme dokumentiert ist.

Installation der IDE unter Mint/Ubuntu

Für Linux, Windows und den Mac steht die IDE unter der Adresse [arduino.cc | Download](https://www.arduino.cc/en/software) bereit. Hier ist die Installation für Mint beschrieben.

 Die Installation ist über die Paketquellen von Ubuntu möglich, wegen der veralteten Version aber nicht zu empfehlen.

 Lade die IDE (Februar 2022: [arduino-1.8.19-linuxXX.tar.xz](#)) auf deinen Computer.

 XX steht für 32 oder 64 Bit. Wähle die Variante, die deinem Computer entspricht. In dieser Dokumentation kommt die [arduino-1.8.19-linux64.tar.xz](#) zum Einsatz.

 Entpacke die Datei in dem gewünschten Verzeichnis – in dieser Dokumentation [Apps](#).

 [Apps](#) liegt in dem (persönlichen Ordner genannten) Verzeichnis, bei mir [tolaris](#) genannt. Diesen Namen musst du durch den Namen deines persönlichen Ordners ersetzen.

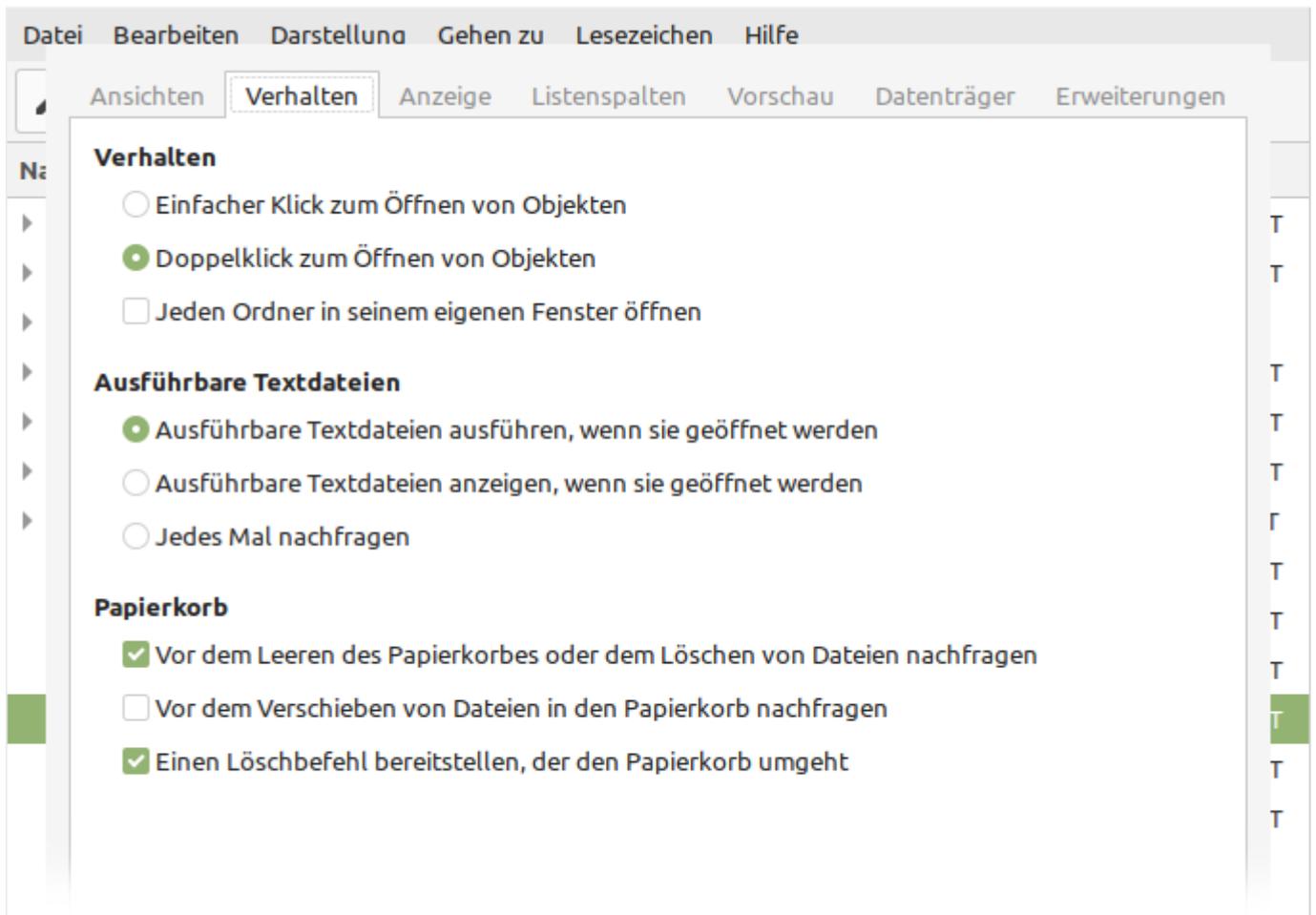
 [Apps](#) enthält ein Verzeichnis [arduino-1.8.19](#).

Name	Größe	Typ	Änderungsdatum
▶ arduino-1.8.19	13 Elemente	Ordner	Mo 20 Dez 2021 17:13:14 CET
▶ Hydrogen	19 Elemente	Ordner	Do 05 Aug 2021 15:28:18 CEST
▶ Ressourcen	7 Elemente	Ordner	Sa 29 Jan 2022 16:49:57 CET
▶ Rosegarden	16 Elemente	Ordner	Di 27 Jul 2021 12:14:33 CEST
▶ supercollider	39 Elemente	Ordner	Mo 16 Aug 2021 18:00:53 CEST
▶ yoshimi	21 Elemente	Ordner	Fr 03 Dez 2021 14:37:52 CET
arduino-1.8.19-linux6...	128,6 MB	Tar-Archiv (XZ-komprimiert)	Sa 29 Jan 2022 16:42:59 CET
Audacity	48,3 MB	AppImage-Anwendungspaket	Mo 22 Nov 2021 14:27:30 CET
MuseScore	160,3 MB	AppImage-Anwendungspaket	Di 27 Jul 2021 12:18:55 CEST
Qsynth	29,1 MB	AppImage-Anwendungspaket	Di 27 Jul 2021 12:24:34 CEST

📁AU1 Das Installationsverzeichnis arduino-1.8.19 (oben). | 📁AU2 Die Installationsdatei (unten).

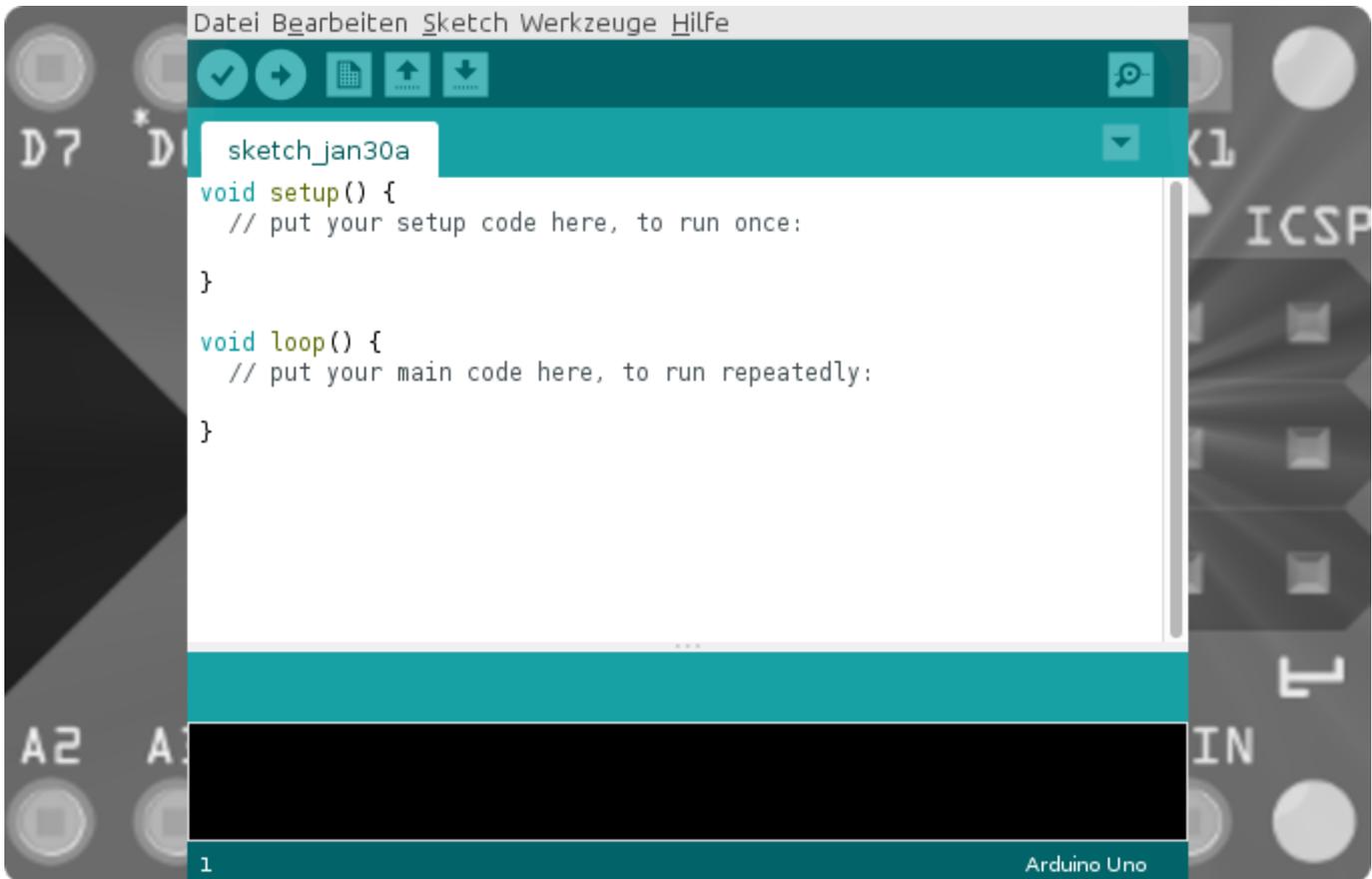
Name	Größe	Typ	Änderungsdatum
▶ examples	11 Elemente	Ordner	Mo 20 Dez 2021 17:12:39 CET
▶ hardware	4 Elemente	Ordner	Mo 20 Dez 2021 17:13:30 CET
▶ java	11 Elemente	Ordner	Di 11 Feb 2020 13:00:22 CET
▶ lib	66 Elemente	Ordner	Mo 20 Dez 2021 17:12:44 CET
▶ libraries	19 Elemente	Ordner	Mo 20 Dez 2021 17:12:41 CET
▶ tools	3 Elemente	Ordner	Mo 20 Dez 2021 17:12:41 CET
▶ tools-builder	2 Elemente	Ordner	Fr 14 Mai 2021 14:43:53 CEST
arduino	882 Bytes	Shell-Skript	Mo 20 Dez 2021 17:12:43 CET
arduino-builder	26,5 MB	Programm	Mo 20 Dez 2021 17:13:13 CET
arduino-linux-setup.sh	5,9 kB	Shell-Skript	Mo 20 Dez 2021 17:12:43 CET
install.sh	11,1 kB	Shell-Skript	Mo 20 Dez 2021 17:12:43 CET
revisions.txt	96,2 kB	Einfaches Textdokument	Mo 20 Dez 2021 17:12:34 CET
uninstall.sh	86 Bytes	Shell-Skript	Mo 20 Dez 2021 17:12:43 CET

-  Lösche die Datei [arduino-1.8.19-linux64.tar.xz](#).
-  Starte die Datei [install.sh](#) in [arduino-1.8.19](#).
-  Der Dateimanager muss so eingestellt sein, dass er Textdateien ausführt (und nicht anzeigt). Die Einstellung erfolgt unter Mint in [Bearbeiten > Einstellungen > Verhalten](#).



[AU3](#) Die Einstellungen unter Mint.

-  Die Installation erzeugt ein Symbol auf dem Desktop und einen Menüeintrag.
-  Starte die IDE.
-  Die IDE erzeugt ein Verzeichnis [Arduino](#) in [tolaris](#) für die Sketches. Die IDE ist bereit und zeigt einen leeren Sketch mit dem Tagesdatum wie in [AU4](#).



☞AU4 Die Entwicklungsumgebung (IDE) für Arduino mit der Voreinstellung für den Arduino Uno.

- 💡 Aus Gründen der Übersichtlichkeit wird in dieser Dokumentation das Verzeichnis [Arduino](#) in [sketchbook](#) umbenannt und in das Verzeichnis [Apps](#) verschoben. Anschließend erfolgt die Anmeldung in der IDE.
- 🖋 Nenne das Verzeichnis [Arduino](#) in [sketchbook](#) um.
- 🖋 Verschiebe das Verzeichnis [sketchbook](#) in [Apps](#).
- 🖋 Datei > Voreinstellungen > Einstellungen > Durchsuchen.
- 🖋 Navigiere zu dem Verzeichnis [sketchbook](#).
- 💻 Der Eintrag in [Sketchbook-Speicherort](#) lautet jetzt `/home/tolaris/Apps/sketchbook`.

Einstellungen Netzwerk

Sketchbook-Speicherort:

Editor-Sprache: (erfordert Neustart von Arduino)

Editor-Textgröße:

Oberflächen-Zoomstufe: Automatisch (erfordert Neustart von Arduino)

Thema: (erfordert Neustart von Arduino)

Ausführliche Ausgabe während: Kompilierung Hochladen

Compiler-Warnungen:

Zeilennummern anzeigen Zeilennummern anzeigen

Code nach dem Hochladen überprüfen Code nach dem Hochladen überprüfen

Beim Start nach Updates suchen Beim Start nach Updates suchen

Use accessibility features

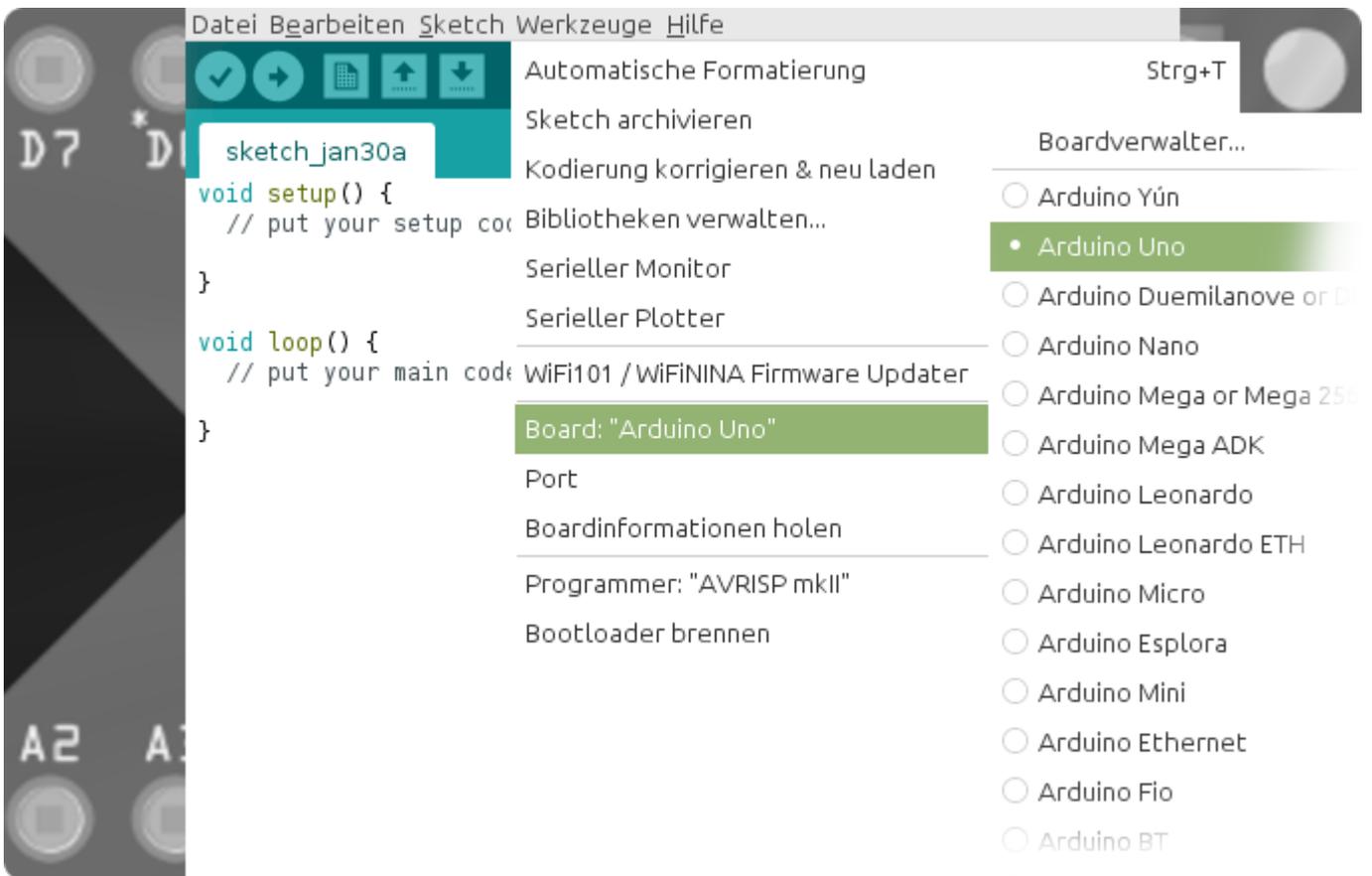
Zusätzliche Boardverwalter-URLs:

AU5 Pfadangabe zum Verzeichnis mit den Sketches.

Datei Bearbeiten Darstellung Gehen zu Lesezeichen Hilfe

Name	Größe	Typ	Änderungsdatum
▶ arduino-1.8.19	13 Elemente	Ordner	Mo 20 Dez 2021 17:13:14 CET
▶ Hydrogen	19 Elemente	Ordner	Do 05 Aug 2021 15:28:18 CEST
▶ Ressourcen	7 Elemente	Ordner	Sa 29 Jan 2022 16:49:57 CET
▶ Rosegarden	16 Elemente	Ordner	Di 27 Jul 2021 12:14:33 CEST
▶ sketchbook	1 Element	Ordner	Sa 29 Jan 2022 19:12:20 CET
▶ supercollider	39 Elemente	Ordner	Mo 16 Aug 2021 18:00:53 CEST
▶ yoshimi	21 Elemente	Ordner	Fr 03 Dez 2021 14:37:52 CET
Audacity	48,3 MB	ApplImage-Anwendungspaket	Mo 22 Nov 2021 14:27:30 CET
MuseScore	160,3 MB	ApplImage-Anwendungspaket	Di 27 Jul 2021 12:18:55 CEST
Qsynth	29,1 MB	ApplImage-Anwendungspaket	Di 27 Jul 2021 12:24:34 CEST

AU6 Apps enthält die Verzeichnisse für die Sketches und die IDE.



📖 AU8 Board bestimmen.

✎ Bestimme den Arduino-Port mit **Werkzeuge > Port > /dev/ttyACM0 (Arduino Uno)**.



📖 AU9 Port bestimmen.

Um den Port unter Mint zu nutzen, musst du die Konfiguration anpassen:

 Starte ein Terminal.

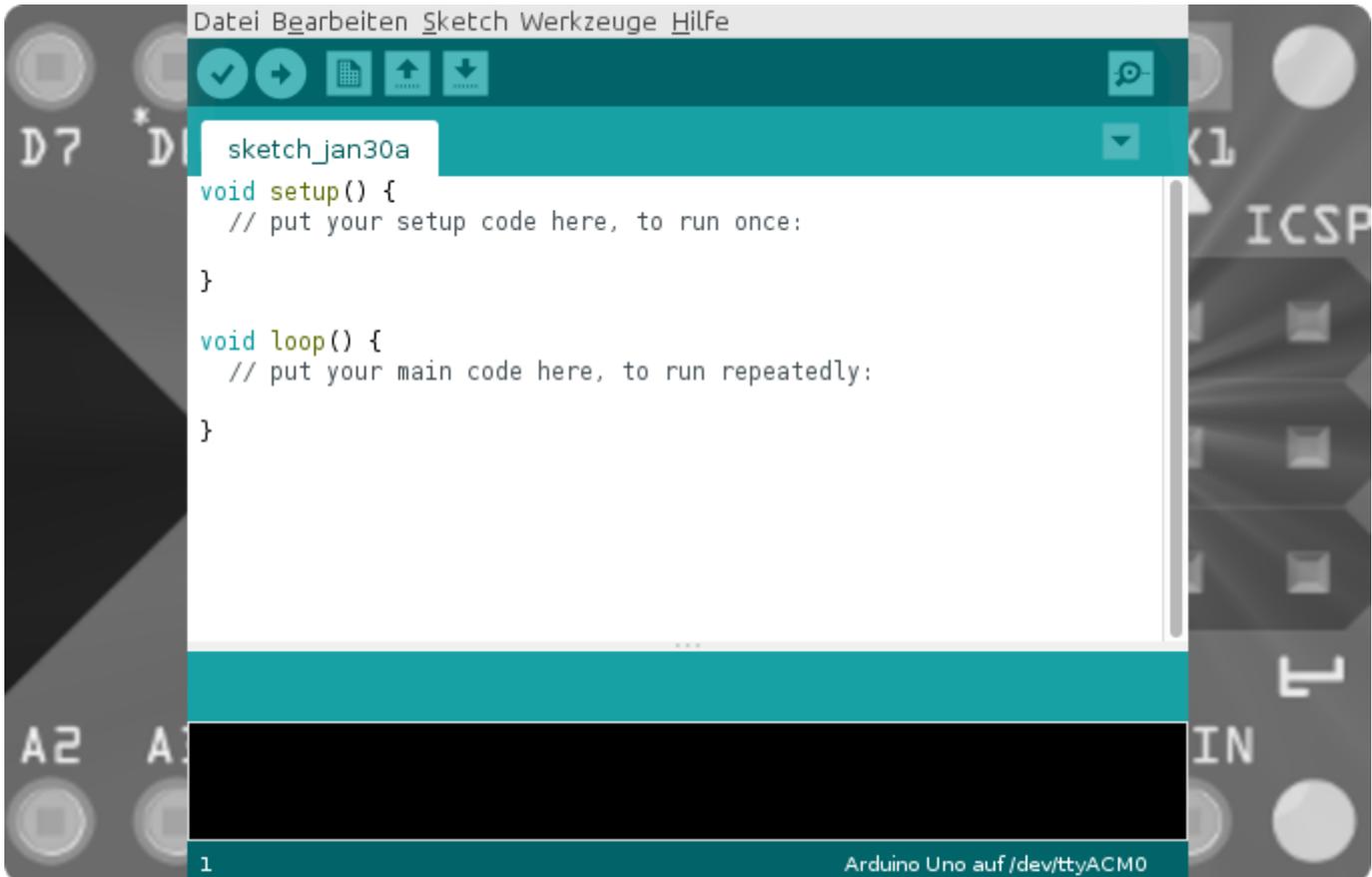


```
sudo usermod -a -G tty tolaris
sudo usermod -a -G dialout tolaris
```

 Fahre den Computer herunter und starte ihn erneut.



Starte die Arduino-IDE.



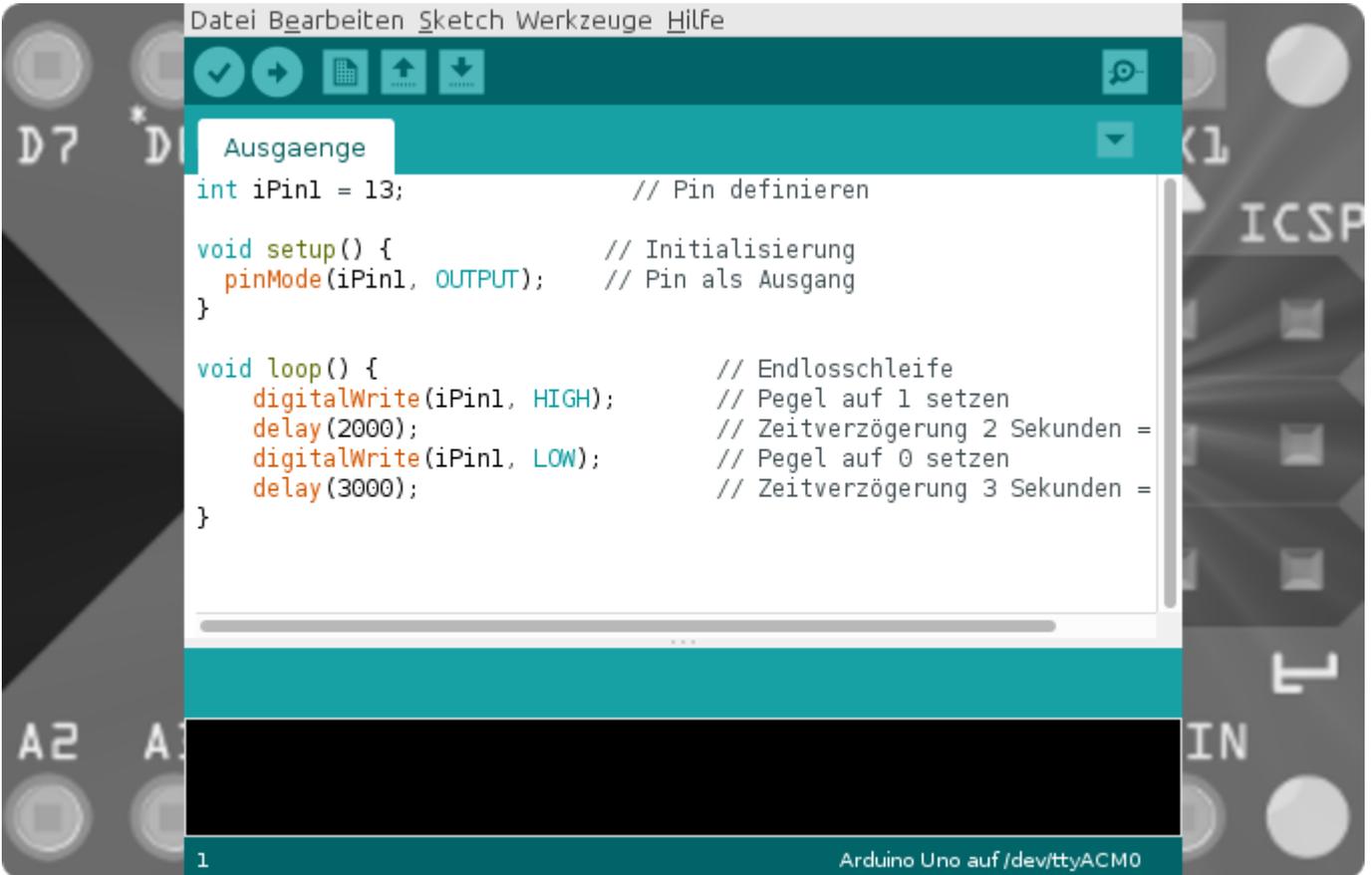
 AU10 Anschluss Computer ⇌ Arduino Uno mit dem Port (am unteren Bildrand).

Sketch laden

-  Erzeuge in dem Verzeichnis [sketchbook](#) ein neues Verzeichnis [Uno](#).
-  Lade die Datei [↓ausgaenge.zip](#), speichere sie in dem Verzeichnis [Uno](#) und entpacke sie.
-  Die [Ausgaenge.ino](#) liegt dann in dem Unterverzeichnis [Ausgaenge](#), der komplette Pfad lautet [/home/tolaris/Apps/sketchbook/Uno/Ausgaenge/Ausgaenge.ino](#).
-  Du kannst die [Ausgaenge.ino](#) direkt starten, oder wie folgt über die IDE.
-  Starte den Sketch mit [Datei > Sketchbook > Uno > Ausgaenge](#).

 Alternativ nutze Datei > Öffnen > /home/tolaris/Apps/sketchbook/Uno/Ausgaenge > Ausgaenge.ino.

 Die IDE zeigt den Sketch.



The screenshot shows the Arduino IDE interface. The menu bar at the top includes 'Datei', 'Bearbeiten', 'Sketch', 'Werkzeuge', and 'Hilfe'. Below the menu bar is a toolbar with icons for saving, running, uploading, and downloading. The main editor window displays the following C++ code for a sketch named 'Ausgaenge':

```
int iPin1 = 13;           // Pin definieren

void setup() {           // Initialisierung
  pinMode(iPin1, OUTPUT); // Pin als Ausgang
}

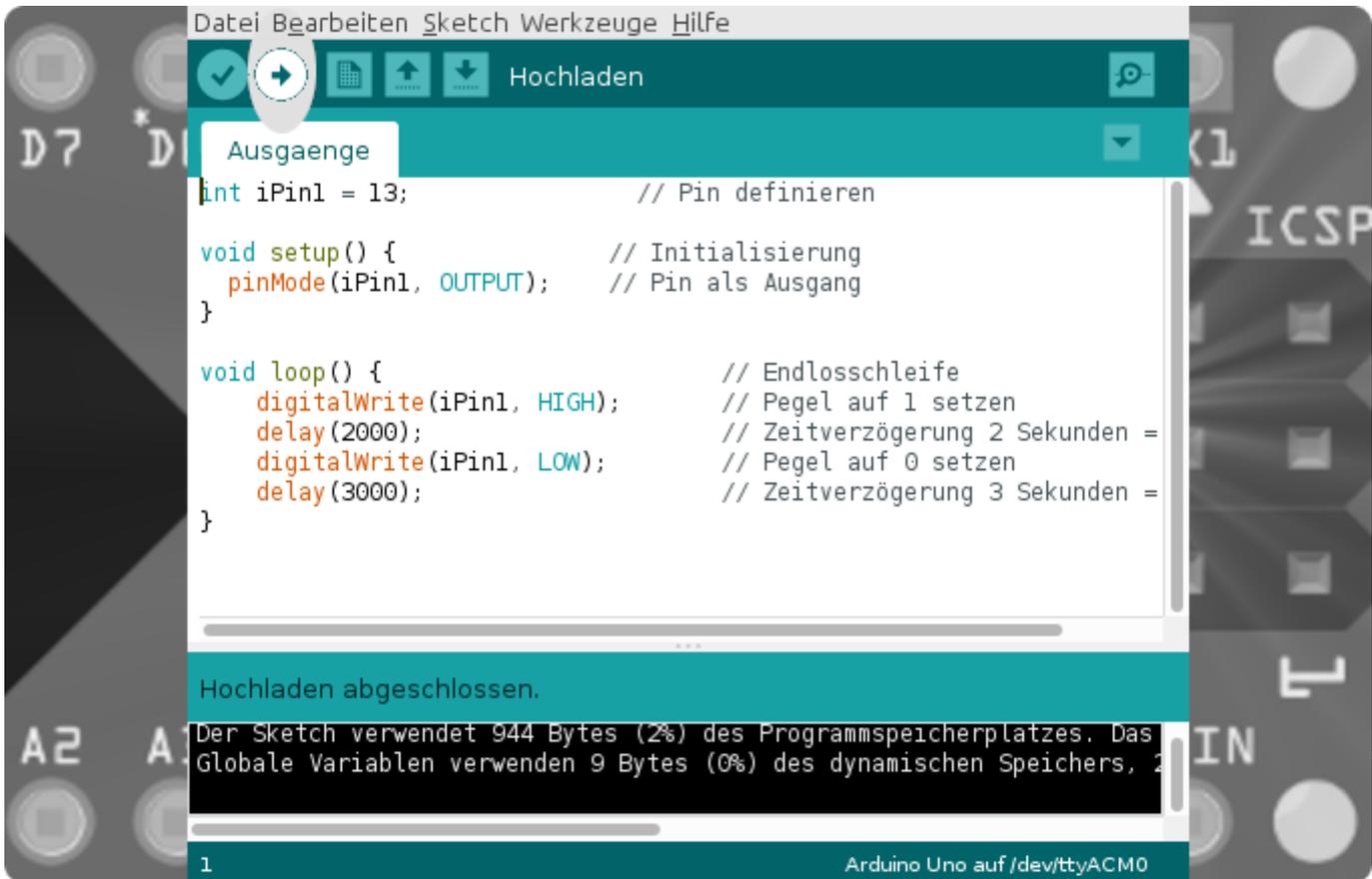
void loop() {            // Endlosschleife
  digitalWrite(iPin1, HIGH); // Pegel auf 1 setzen
  delay(2000);           // Zeitverzögerung 2 Sekunden =
  digitalWrite(iPin1, LOW);  // Pegel auf 0 setzen
  delay(3000);           // Zeitverzögerung 3 Sekunden =
}
```

At the bottom of the IDE, the status bar shows '1' on the left and 'Arduino Uno auf /dev/ttyACM0' on the right.

 AU11 Geladener Sketch.

 Drücke den Knopf **Upload**.

 Die IDE meldet "Upload abgeschlossen".



■AU12 Der Upload Knopf (oben) und die Erfolgsmeldung (unten).



Die LED auf dem Mikrocontroller blinkt.

Türen

Von außen betrachtet verhält sich der Mikrocontroller wie eine Burg mit zahlreichen Türen. Jede Tür kann grundsätzlich als Eingang oder als Ausgang dienen. Zur optimalen Verkehrssteuerung ist es sinnvoll, Ein- und Ausgänge zu trennen. Bei den alten Burgen war das vielleicht noch nicht üblich, bei den Mikroprozessoren schon: ein Ausgang kann nicht als Eingang benutzt werden – und umgekehrt.



■MC19 Türen und Tore.

Die seitlichen Türen dienen entweder als Ein- oder als Ausgang. Das Hauptportal dient zur Versorgung, vergleichbar mit der Spannungsversorgung beim Arduino.

Welche Tür Ein- oder Ausgang ist, wird in dem Sketch genannten Programm bestimmt. In der Welt der Mikrocontroller ist nicht nur alles streng geregelt, sondern auch in Sein oder Nichtsein unterteilt. Entweder du bist oder du bist nicht. Die Lampe leuchtet oder nicht. Das sind alles die gleichen Aussagen: ein Mikrocontroller (wie ein Computer) kann nur zwischen 0 und 1 unterscheiden. Diese beiden Zustände werden durch eine elektrische Spannung ausgedrückt:

- 0 entspricht einer Spannung von 0 V
- 1 entspricht einer Spannung von ca. 5 V, also einer Toleranz über und unter 5 V

Ein Ausgang stellt eine Spannung von ca. 5 V oder 0 V zur Verfügung, je nach Sketch. Ein Eingang erwartet eine Spannung von 5 V oder 0 V, ebenfalls abhängig von dem Sketch. Ein Strom fließt jeweils, wenn der Stromkreis geschlossen ist.

In der realen Welt ist nicht alles 0 und 1, aber fast alles kann in Nullen und Einsen ausgedrückt werden. Dazu sind entsprechend viele davon erforderlich, die zu Gruppen (z.B. 8 Stück nebeneinander) zusammengefasst werden. Wenn etwas nur den Zustand 0 oder 1 annehmen kann, wird es als Bit bezeichnet. 8 Bit als Gruppe ergibt ein Byte. Unsere alphanumerischen Zeichen (Buchstaben a-z, Sonderzeichen und Ziffern) werden jeweils durch ein Byte ausgedrückt. Das Byte 0000 0101 entspricht der Ziffer 5 in unserem Dezimalzahlensystem. Wenn du alle möglichen Kombinationen von 0-en und 1-en in einem Byte erstellst, erhältst du 256. Platz genug für unsere überschaubare Anzahl an Zeichen.

Ein Computer verarbeitet nicht nur alphanumerischen Zeichen, sondern allgemein Daten. Schneller ist besser. Deswegen benötigt ein Computer viele Türen, die aktuellen Modelle haben 64 davon. Ein Mikrocontroller ist die Sparversion des Computers, er hat weniger Türen. Das Prinzip ist das gleiche. Die Türen des Mikrocontrollers werden als Ports bezeichnet.

Der Arduino Uno kann die Ports D0 bis D13 entweder als digitalen Ein- oder Ausgang verwenden. Die Definition der Ports erfolgt in dem verwendeten Sketch.

In dem Sketch `Ausgaenge.ino` und den folgenden Versuchen dienen D0 und D13 als Ausgänge.

Schaltplan

Der Schaltplan zeigt eine reduzierte Darstellung dessen, was wir gerade benötigen. Es geht an dieser Stelle nicht darum, den Mikrocontroller detailgetreu abzubilden, sondern nur um Funktionsweisen. Also wer macht was?



 MC15 Digitale Ports.

D0 bis D13 sind mögliche digitale Ausgänge, also insgesamt 14.

Bauteile:

- MC: Arduino Uno



Die Spannungsversorgung des Arduino stammt aus dem USB-Anschluss. Ohne USB-Verbindung ist für den Betrieb des Arduino eine externe Gleichspannung erforderlich, z.B. in Form eines 9 V Steckernetzteils.

Sketch



`Ausgaenge.ino`

- D13 arbeitet in dem Sketch `Ausgaenge.ino` als Ausgang, der den Pegel zwischen LOW und HIGH in einer Endlosschleife wechselt.
- D0 ist (noch) nicht definiert!
- Die interne LED des Mikrocontrollers leuchtet, wenn der Pegel am Ausgang D13 HIGH ist.
- Die Spannung am Ausgang D13 beträgt ca. 4,7 V bei HIGH und 0 V bei LOW.
- Den Sketch hast du bereits auf deinen Computer geladen, wenn du [an der Strippe](#) warst.

Ausgaenge.ino

```
int iPin1 = 13;           // Pin definieren

void setup() {           // Initialisierung
  pinMode(iPin1, OUTPUT); // Pin als Ausgang
}

void loop() {            // Endlosschleife
  digitalWrite(iPin1, HIGH); // Pegel auf 1 setzen
  delay(2000);             // Zeitverzögerung 2 Sekunden = 2000 ms
  digitalWrite(iPin1, LOW);  // Pegel auf 0 setzen
  delay(3000);             // Zeitverzögerung 3 Sekunden = 3000 ms
}
```

Funktionsweise

Programm, Sketch und Code sind verschiedene Worte für die gleiche Sache. [Sketch](#) ist der von den Arduino Schöpfern gewählte Begriff für ein Programm.

Der Sketch ist in Anweisungen und Kommentare unterteilt.

Der dreiteilige Programmaufbau ist typisch für die Arduino-Familie: Definition (Wer ist dabei?), Initialisierung (Was macht wer?) und Ausführung (Wie wird es gemacht?). Das Programm läuft nach dem Start vom Anfang bis zum Ende und dann wieder von vorne, also in einer Schleife. Wie jede gesprochene Sprache kennt eine Programmiersprache eine Anzahl an Wörtern mit fester Bedeutung sowie Regeln für den Satzaufbau. In einer Programmiersprache werden Wörter als Schlüsselwörter und Sätze als Anweisungen bezeichnet. Jede Anweisung wird mit einem Semikolon abgeschlossen.

1. Die erste Anweisung

```
int iPin1 = 13;
```

definiert den verwendeten Port. Darin ist `int` das dafür vorgesehene Schlüsselwort, `iPin1` ein frei wählbarer Name und `13` der zugewiesene Port.

2. Die Initialisierung erfolgt mit dem Schlüsselwort `void` und der Funktion² `setup()`

```
void setup() {}
```

sowie der zweiten Anweisung

```
pinMode(iPin1, OUTPUT);
```

zwischen den geschweiften Klammern `{}`, die Port 13 als Ausgang bestimmt.

3. Die Ausführung erfolgt mit dem Schlüsselwort `void` und der Funktion² `loop()`

```
void loop() {}
```

in einer Endlosschleife, ebenfalls mit den Anweisungen zwischen den geschweiften Klammern `{}`.

```
digitalWrite(iPin1, HIGH);
```

setzt den Pegel von Port D13 auf HIGH. Dann liegen an diesem Ausgang knapp 5 V an, aber nur für 2 Sekunden.

```
delay(2000);
```

ist eine Zeitverzögerung, ausgedrückt in Millisekunden.

```
digitalWrite(iPin1, LOW);
```

setzt den Pegel auf LOW, also auf 0 V, aber nur für 3 Sekunden.

```
delay(3000);
```

Nach dieser Zeitverzögerung ist das Programm durchlaufen und startet von vorne.

Versuche

-  Ändere in dem Sketch den Ausgang von 13 auf 0.
-  Lade den Sketch in den Mikrocontroller.
-  Die interne LED des Mikrocontrollers leuchtet ohne Unterbrechung.
-  Überprüfe mit einem Spannungsmeßgerät den Pegel am Ausgang D0.
-  Die Spannung am Ausgang D0 beträgt ca. 4,7 V bei HIGH und 0 V bei LOW.

Wie die Profis

Programmcode zu lesen und zu verstehen ist gewöhnungsbedürftig. Ein gut strukturierter Code mit einem Namensschema für alle Programmteile erleichtert den Umgang und reduziert die Hemmschwelle, sich mit dem Code zu befassen.

Was für die redaktionellen Inhalte die Konventionen von [tolaris know-how](#) sind, sind für die Codebeispiele die [Programmierkonventionen](#).

Weiter ist unter den Profis ein Schema für die Abgrenzung von Codeteilen mit Einrückungen ungeschriebenes Gesetz, wie in dem Sketch `Ausgaenge.ino` verwendet.

Eine Erleuchtung

... oder die Antwort auf die Frage: Wie bringe ich eine LED zum Leuchten?

Eine Möglichkeit besteht darin, eine 4,5 V Batterie zu nehmen und die LED mit einem Vorwiderstand daran anzuschließen. Der Vorwiderstand ist zur Strombegrenzung nötig, damit die LED nicht durchbrennt. Willst du mehr Komfort, baust du noch einen Schalter dazwischen. Willst du einen Effekt (z.B. blinkende LED), schaltest du den Schalter an und aus. Und wieder an und aus. Und so weiter. In

diesem Fall bist du mit deiner Batterie und dem Schalter der große Controller, der bestimmt, wo es lang geht. Der eine Spannung an die Kombination LED/Widerstand ausgibt – oder eben nicht. Licht an kannst du als Zustand 1 definieren, Licht aus als Zustand 0. Das ist das Prinzip der Digitaltechnik.

Mit dieser Tätigkeit als großer Controller wird in dir bald der Wunsch nach einer Veränderung aufkeimen. Einerseits kriegst du nichts anderes mehr gebacken, andererseits führt das endlose Schalten zu Krämpfen in der Hand und auf Dauer zur völligen geistigen Verblödung. Wegen der Krämpfe wird der Zeitabstand zwischen Licht an und Licht aus unregelmäßig – in der Folge schickt dir dein Publikum noch eine Klage ins Haus wegen Produktmangels.

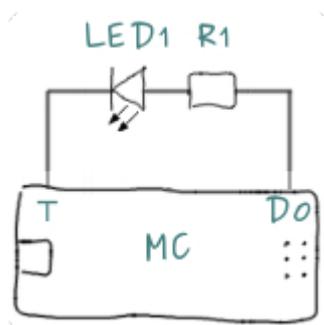
Also deligierst du den Job an eine Blech-Emma, oder politisch korrekt: an einen Blech-Emmerich wie den Arduino. Egal. Der Arduino ist zwar nur ein Mikrocontroller, aber der macht das schon.

Und wie?

Du musst ihm sagen, wann er wo und wie lange eine 1 (also eine Spannung) oder eine 0 (also keine Spannung) ausgibt. Das Wo ist die Tür (der Port), ausgeben bedeutet: ein Ausgang muss her. LED und Vorwiderstand schließt du zwischen dem Ausgang und der Masse an, damit der Stromkreis geschlossen ist und ein Strom fließen kann, wenn zwischen Ausgang und Masse eine Spannung anliegt. Dann ist der Ausgang High, also 1. Wenn du dann noch den richtigen Widerstand erwischst hast und die LED richtig angeschlossen ist, leuchtet die LED, wie in dem Sketch festgelegt.

Schaltplan

Der Schaltplan zeigt einen Mikrocontroller, der aus dem Port `D0` eine LED über einen Vorwiderstand ansteuert. Der dazu passende Sketch heißt `LED_blinken_1.ino`.



MC16a Schaltplan zum Sketch `LED_blinken_1.ino`.

Bauteile:

- MC: Arduino Uno
- LED1: LED 5 mm
- R1: 220 Ω

Die verwendete LED ist für 20 mA ausgelegt. Diese Angabe kannst du dem Datenblatt des Bauelements entnehmen. Jedes elektronische Bauelement hat technische Spezifikationen, die entweder auf dem Bauelement aufgedruckt sind oder in Form eines Datenblatts verfügbar sind.

Bei einer Spannung von 5 V am Ausgang `D0` folgt aus [SER1.1](#) ein Wert für $R1 = 250 \Omega$. Da die Spannung des Arduino in der Regel unter 5 V liegt, reicht ein kleinerer Widerstand, der als Standardwert mit 220 Ω verfügbar ist. Standardwert bedeutet: dieses Bauelement kannst du mit diesem Wert kaufen. Sollte der Strom (wegen des kleineren Widerstands) geringfügig höher sein, wird die LED keinen Schaden nehmen. Der Widerstandswert ist also nicht kritisch. (Bei kritischen Anwendungen musst du den Widerstand aus einzelnen Bauelementen kombinieren.)

$$R = \frac{U}{I} = \frac{5V}{20mA} = \frac{5 \times 1000V}{20A} = 250\Omega$$

§ER1.1 Widerstandsberechnung

Jetzt geht es an deinen ersten Aufbau einer elektronischen Schaltung.



Du benötigst folgende Teile:

- 1 Mikroprozessor: Arduino Uno
- 1 Widerstand: 220 Ω
- 1 LED: rot
- 1 Steckbrett
- 3 Drahtbrücken
- 1 Computer mit installierter Arduino-IDE
- 1 USB-Kabel
- 1 Messgerät

Wenn du dich noch nie mit elektronischen Bauteilen befasst hast, brauchst du noch Antworten auf einige Fragen:

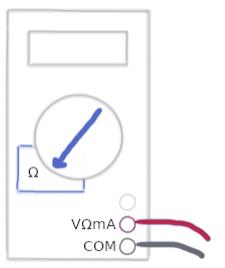
- Wie kann ich den Wert eines Widerstands ermitteln?
- Wie schlieÙe ich die LED an?

Der Widerstand ist an dem aufgedruckten Wert oder an dem Farbcode zu erkennen. Die meisten Widerstände tragen 3 oder 4 farbige Ringe, die Reihenfolge dieser Ringe von links nach rechts gibt den Wert in Ohm [Ω] an. Rechts ist die Seite mit dem zusätzlichen silbernen oder goldenen Ring. Silber bedeutet 10% und gold 5% Genauigkeit.



E3.220 Mit 3 Farbringen entspricht dieser Widerstand einem Wert von 220 Ω .

Hast du die Farbcodes nicht im Kopf (oder keinen Zugriff auf die Informationen), kannst du mit einem Messgerät wie in E22 den Widerstand ermitteln. Stell dein Messgerät auf den Bereich Ω , der gewählte MeÙbereich muss über dem Widerstandswert liegen. Im Zweifel wähle den größten Bereich 2000 k. Als Anschlüsse dienen auf dem Messgerät V Ω mA und COM. Halte die rote Strippe an die eine Seite und die schwarze Strippe an die andere Seite des Widerstands. Wo welche Strippe liegt ist (hier) egal. Das Messgerät sollte irgendwas anzeigen. Wähle einen kleineren Bereich, bis du den Wert so genau wie möglich lesen kannst.



E22 Multimeter mit der Einstellung zur Widerstandsmessung.

Die LED hat entweder ein abgeknicktes Beinchen oder ein längeres und ein kürzeres Beinchen. Das abgeknickte sowie das längere Beinchen gibt jeweils den Pluspol an.



E21 Rote LED.

Sketch

 LED_blinken_1.ino

- D0 arbeitet als Ausgang, der die LED in einer Endlosschleife an- und ausschaltet.

 Alle Beispiele in dieser Dokumentation sind mit dem kompletten Code notiert. Der Link mit dem Sketchnamen über dem Code führt jeweils den Download aus.

 Lade den Sketch LED_blinken_1.ino und speichere ihn in deinem Sketch-Verzeichnis.

 Starte die Arduino-IDE.

 Öffne die Datei [LED_blinken_1.ino](#) und führe den Download auf deinen Arduino aus.

 Die LED blinkt.

LED_blinken_1.ino

```
int iLED1 = 0;                // LED-Pin

void setup() {                // Initialisierung
  pinMode(iLED1, OUTPUT);    // LED-Pin als Ausgang
}

void loop() {                 // Endlosschleife
  digitalWrite(iLED1, HIGH); // LED einschalten
  delay(2000);                // Zeitverzögerung 2000 ms
  digitalWrite(iLED1, LOW);  // LED ausschalten
  delay(3000);                // Zeitverzögerung 3000 ms
}
```

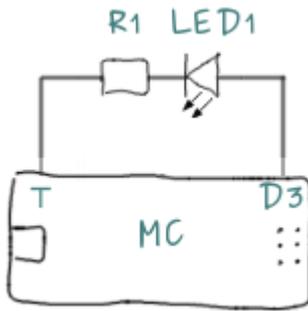
Gestottert

Wenn du Roller fährst, kannst du zum Bremsen mit deinem Schuh mehrfach kurz den Boden berühren. Je nach Feingefühl stottert das Ding mehr oder weniger. Und es ruiniert deine Schuhe. Das Antiblockiersystem beim Auto macht das eleganter: ein Stottern ist nicht wahrzunehmen, wenn stärker oder schwächer gebremst wird.

Die Helligkeit einer Beleuchtung wie der LED am Arduino kannst du über ein Stottern des Stroms erreichen. Schaltest du den Strom sehr schnell unterschiedlich lang an und aus, erreichst du einen sanften Wechsel von hell bis dunkel. Aufgrund ihrer Trägheit nehmen die Augen das Stottern (in der IT Pulse genannt) als ein dauerndes Sinken und Steigen der Helligkeit wahr.

Warum dieser Umstand? Weil der Arduino keinen analogen Spannungsausgang hat.

Schaltplan



Bauteile:

- MC: Arduino Uno
- LED1: LED 5 mm
- R1: 220 Ω

MC17 Schaltplan zum Sketch LED_dimmen.ino.



Wenn du schon **keine Erleuchtung** hattest, weißt du auch, wie der Widerstand berechnet wird.

Sketch



LED_dimmen.ino

- D3 ist ein pulsweitenfähiger Ausgang, der mit dem Sketch LED_dimmen.ino die LED mit unterschiedlich langen Pulsen ansteuert.



Lade den Sketch LED_dimmen.ino und speichere ihn in deinem Sketch-Verzeichnis.



Starte die Arduino-IDE.



Öffne die Datei LED_dimmen.ino und führe den Download auf deinen Arduino aus.



Die LED leuchtet kontinuierlich langsam auf und erlischt wieder.

LED_dimmen.ino

```
int iLED1 = 3;           // LED-Pin

void setup() {
}

void loop() {
  for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5) { // in 5er
    Schritten die Leuchtkraft erhöhen
    analogWrite(iLED1, fadeValue); // Wert ausgeben
    delay(30); //
    Zeitverzögerung 30 ms
  }

  for(int fadeValue = 255 ; fadeValue >= 0; fadeValue -=5) { // in 5er
    Schritten die Leuchtkraft senken
    analogWrite(iLED1, fadeValue); // Wert ausgeben
    delay(30); //
    Zeitverzögerung 30 ms
  }
}
```



Für Programmieranfänger ist die Visualisierung auf der Magazinsseite [bes.t 01](#) gedacht.

beton: objektgestaltung

Ein interessanter Werkstoff. Und ein sensibler Werkstoff. Hart wie Stein, nicht ganz so zerbrechlich wie ein Keks, aber näher dran als gewünscht. Die Rede ist von Beton. Diesen Werkstoff gibt es vielen unterschiedlichen Ausführungen. **tteV** verwendet einen Schnellguss-beton, der filigrane Strukturen abbildet und schnell trocknet. Insbesondere die Gussfähigkeit erlaubt die Gestaltung von allen Objekten, für die eine Form erstellt werden kann. Denkbar ist auch eine Verbindung von mehreren Objekten mit einem entsprechenden Kleber. Die folgenden Bilder zeigen einige Beispiele.



Abdeckung eines Schallplattenregals.



Die Natur ist nicht symmetrisch.



Der Grauton entsteht durch Beimischung von Pigmenten.



Aussparungen für durch die Platte verlaufenden Säulen.



Eine Betonscheibe als Träger für eine Leuchte.



Farbe des Betons ohne Pigmente.

💡 Die Fassung ist nicht aus Beton gegossen.

Wer sich über die Formen wundert: die Natur ist auch nicht symmetrisch. Hast du schon mal eine eckige Frau gesehen? Warum müssen alle Platten rechteckig sein?

💡 Ich weiß, dass die Fotos schlecht sind.

eLearning: lernkultur

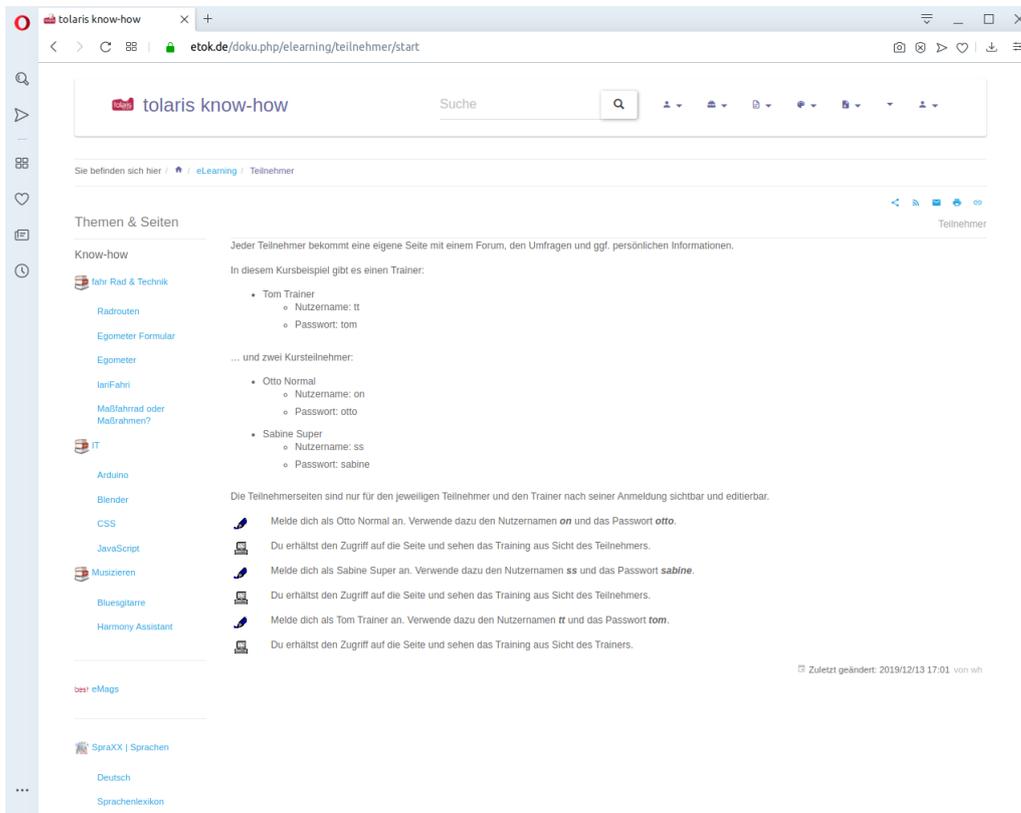
eLearning ersetzt das Lernen an einem physischen Ort durch eine virtuelle Umgebung. Der Unterricht, die Vorlesungen oder die Vorträge sind nicht mehr zeitgebunden, sondern stehen als Video oder in Form einer Bilderschau jederzeit zur Verfügung. Fragen & Antworten werden zeitunabhängig in Foren diskutiert. Übungsstunden finden entweder in einem virtuellen Raum nach vereinbarten Terminen oder in Foren statt, entweder personenbezogen oder in Gruppen.

eLearning erfordert:

1. gut aufbereitetes Trainingsmaterial,
 1. online wie z.B. bei der Einführung in den Mikrocontroller [Arduino](#).
 2. online und offline als eBooks nach dem Konzept von [Corporate Publishing](#).
 3. in Form der QuiXX genannten [interaktiven Visualisierungen](#).
 4. als Video oder als [KiXX-Bilderschau](#).
2. eine Infrastruktur auf Basis der Open-Source-Software wie bei [tolaris know-how](#).
3. eine Lernkontrolle in Form von [Umfragen](#).
4. die Kommunikation in geschlossenen Foren.
5. ggf. die Kommunikation in einem [virtuellen Raum](#).

Auf [tolaris know-how](#) ist mit [eLearning](#) eine Anwendung mit zwei Teilnehmern und einem Trainer realisiert. Teilnehmer und Trainer haben jeweils eigene Zugänge, mit denen die Anwendung aus unterschiedlichen Blickwinkeln betrachtet werden kann. Die Nutzernamen und Passwörter stehen unter der Adresse zur Verfügung.

Trainingsumgebung



The screenshot shows a web browser window with the URL `etok.de/doku.php/elearning/teilnehmer/start`. The page title is "tolaris know-how" and it features a search bar. The main content area is titled "Themen & Seiten" and "Know-how". It lists various topics such as "fahrrad & Technik", "Musizieren", and "Sprachlexikon". The "fahrrad & Technik" section is expanded, showing a list of participants and their login details:

- Jeder Teilnehmer bekommt eine eigene Seite mit einem Forum, den Umfragen und ggf. persönlichen Informationen.
- In diesem Kursbeispiel gibt es einen Trainer:
 - Tom Trainer
 - Nutzernamen: tt
 - Passwort: tom
- ... und zwei Kursteilnehmer:
 - Otto Normal
 - Nutzernamen: on
 - Passwort: otto
 - Sabine Super
 - Nutzernamen: ss
 - Passwort: sabine

Below this list, there are instructions for logging in as each user, such as "Melde dich als Otto Normal an. Verwende dazu den Nutzernamen on und das Passwort otto." The page also includes a footer with the text "Zuletzt geändert: 2019/12/13 17:01 von wh".

Die Trainingsumgebung nach der Anmeldung von Sabine Super.

Sie befinden sich hier / eLearning / Teilnehmer / Sabine Super

Sabine Super

Umfragen

Umfrage 1 – Variablen in JavaScript

100% verstanden? * Ja

ABSENDEN

Umfrage 2 – Funktionen in JavaScript

100% verstanden? * Ja

ABSENDEN

Frage	Name	100% verstanden?
Variablen in JavaScript	Sabine	Nein
Funktionen in JavaScript	Sabine	Ja

Wie bewertest du den Trainer?

Voller Name	sehr gut	gut	befriedigend	ausreichend	mangelhaft	ungenügend
Summe:	0	0	0	0	0	0
Sabine Super (ss)	<input type="checkbox"/>					

ABSTIMMEN

Die Sicht von Sabine Super.

Sie befinden sich hier / eLearning / Umfragen

Wissensstand

Umfragen dienen dazu, den Wissensstand jedes Teilnehmers zu einzelnen Trainingsinhalten zu erfragen. Dazu bereitet der Trainer eine Liste von Fragen vor, die auf jeder Teilnehmerseite aufgeführt sind und im Laufe des Trainings beantwortet werden. Die einfache Frage lautet jeweils:

Ist das Thema 100% verstanden?

Die Frage kann nur mit Ja oder Nein beantwortet werden. Beispiele für die Umfragen sind auf den Teilnehmerseiten von Sabine Super und Otto Normal enthalten.

Jeder Teilnehmer kann nur seine eigenen Antworten sehen. Die Antworten aller Teilnehmer sind nur für den Trainer sichtbar. Ein Beispiel ist auf der Trainerseite von Tom Trainer enthalten.

Um den Inhalt zu sehen, melde dich als Tom Trainer an:

- Verwende dazu den Nutzernamen **tt** und das Passwort **tom**.
- Du erhältst den Zugriff auf die Seite und sehen das Training aus Sicht des Trainers.

Allgemeine Umfragen/Abstimmungen

Umfragen/Abstimmungen können öffentlich sichtbar erfolgen.

Voller Name	Terminvorschlag Zusatztraining:		
	Sa 10:00 Uhr	Sa 14:00 Uhr	Sa 18:00 Uhr
Otto Normal (on)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sabine Super (ss)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Tom Trainer (tt)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Summe:	1	1	3

Zuletzt geändert: 2019/12/13 17:01 von wh

Wissensstandsabfrage.

Visualisierungen

Die interaktiven Visualisierungen **QuiXX** und die **KiXX**-Bilderschaufen sind ebenfalls Open-Source-Software auf Basis von jQuery und laufen unabhängig von **tolaris know-how** in einem Browser offline und online. Die **KiXX**-Bilderschau wirkt unscheinbar, ist aber als Alternative zu einem kurzen Video sehr

brauchbar. Die Idee: statt eines Videos bei Youtube eine Bilderserie als Schau laufen zu lassen. KiXX kommt in drei Varianten:



KiXX1 Die synchronisierte Variante der Bilderschau.



KiXX5 Bilderschau ohne Tonsynchronisation.



KiXX8 Die Bilderschau in Stereo.

Die interaktiven Visualisierungen mit **QuiXX** sollen spielerisch, aber seriös Wissen vermitteln. Ein Beispiel aus einer Lernanwendung für Kinder (und alle, die noch staunen können) ist auf der eMag-Seite [bes.t 01](#) dieser Ausgabe zu sehen. Für den Zugriff ist eine Anmeldung erforderlich:

- Nutzer: b01
- Passwort: best

The image shows a screenshot of the QuiXX interface. At the top left is a small icon of a graduation cap. Below it is the text 'Liste A'. The main area contains a grid of interactive cards. Each card has a text label and a corresponding icon. The cards are arranged in three rows and four columns. The first row contains: 'Wer?' with a group of people icon, '13' with a person icon, 'Was->' with a castle icon, and '13' with a castle icon. The second row contains: '<-Was' with a castle icon, 'Wie>' with a globe icon, '13' with a person icon, and '2000' with an hourglass icon. The third row contains: '13' with a castle icon, '3000' with an hourglass icon, '<Wie' with a globe icon, and an empty space. Below this grid is a horizontal line, followed by the text 'Liste B'. Under 'Liste B', there are three rows of empty boxes corresponding to the text labels in List A: 'Wer?' and 'Was->' in the first row, '<-Was' and 'Wie>' in the second row, and '<Wie' in the third row.

Eine Visualisierung aus „Eine unglaubliche Geschichte“.

Die Lernanwendung ist eine Einführung in die Elektronik allgemein und die Handhabung und Programmierung von Arduino in Gestalt eines eBooks mit Verlinkung auf Inhalte bei [tolaris know-how](#). Konzeptionell ist das eBook mit diesem eMag vergleichbar, kommt jedoch als Märchen daher. Die Fertigstellung ist für Ende 2019 geplant, einen Eindruck vermittelt die erste Seite aus "Eine unglaubliche Geschichte".

... für alle, die noch staunen können.

Das ist die Geschichte von dem Königssohn Fixbit und seiner Entdeckungsreise, in der er das königliche Schloss zu einem Mikrocontroller umbaut. Dabei begleiten ihn seine Eltern Arduino und Genuina.

Das ist auch die Geschichte von der Entdeckung des elektrischen Stroms und des kleinen Unterschieds zwischen Jungs und Mädchen.

Es war einmal

... ein Königssohn mit dem Namen Felix Friedrich Ferdinand Freiherr von Bitburg zu Drachenfels auf Nimmerland in Uino – und da sich kaum jemand diesen langen Namen merken konnte, wurde er kurz Fixbit genannt.

Sein Vater, der König, hieß mit Vornamen Ard und ließ sich entweder mit „Seine Herrlichkeit“ oder einfach mit Arduino anreden. Seine Gemahlin wurde kurz Genuina genannt, weil sie auf die geniale Idee gekommen war, den Apfelkuchen zu salzen statt ihn zu zuckern. Der Apfelkuchen war Arduinos Lieblingsspeise. Arduino und Genuina waren ein prächtiges Paar, das sich bestens vertrug. Außer, wenn Genuina mal wieder den Apfelkuchen versalzen hatte. Aber das passierte nicht oft, nur einmal täglich.

Der wahrhaft königliche Name konnte nicht darüber hinwegtäuschen, dass das Königshaus mit finanziellen Problemen zu kämpfen hatte. Das ist so, als wenn dein Taschengeld mal wieder nicht reicht. Vornehm ausgedrückt und dem majestätischen Haus angemessen: man war dem Leben zugetan und das kostet eben. Allein den Wein auf den Drachenfels zu schleppen – auf dem Fels wächst ja nix. Und erst die Heizkosten, die Bude – also das Schloss – hatte noch keine dämmenden Fassaden und durch die Fenster blies ein Wind, dass Fixbit seine Nachtmütze anschnallen musste.

Außerdem wollte der Adel bei Laune gehalten werden, denn die sollten darauf achten, dass das Volk nicht macht, was es will – sondern was Arduino will. Und der musste sich deswegen immer neue Vorschriften ausdenken, das macht hungrig. Denn nur wer satt und zufrieden ist hat Lust, anderen Vorschriften zu machen. Also gab es regelmäßig Versammlungen, bei denen besprochen wurde, was zu tun ist. Und das klappte am besten mit ´ner Party. Und da Arduino wirklich ein großzügiger Mensch war, durften alle am Hofe an der Party teilnehmen und mit ihm essen und trinken. Und das zeigte schnell Erfolge, die Bäume wuchsen. So war das Wort Wachstum erfunden. Das wurde schnell zur Strategie: alles was Wachstum versprach, war gut. Je mehr Parties, desto mehr Wachstum.

Wenn du meinst, der König mit seinem Hof hatte ein leichtes Leben, dann irrst du aber. Raus aus dem Bett, rein in die Klamotten und zu Tisch, Essen ordern, würdevoll gucken, Essen essen, Wein trinken, wieder in die Kammer, raus aus den Klamotten, Schläfchen halten usw. Jeden Tag. Das ist schon ein harter Job. Und als König kannst du nicht einfach mal eben in die Jeans springen. Schon mal was von Etikette gehört?

Das ist heute ungefähr das, was jede Flasche im Supermarkt auf dem Bauch kleben hat. Jedenfalls braucht ein König ein königliches Gewand und nicht nur eines. Das kann aber nur ein Schneider nähen und der ist teuer. Und wegen des Wachstums brauchte der König auch noch einen Änderungsschneider, der die Gewänder dem Wachstum anpasst. Wegen der vielen Gewänder mehrere Schneider. Die anderen bei Hofe konnten schlecht unbekleidet durch die Gegend rennen – noch mehr Gewänder und noch mehr Schneider.

Reisekosten, Gewänder, Essen und Trinken. Das ist noch immer nicht alles. Soll der König etwa zu Fuß gehen? Nein, das könnte ja das Wachstum hemmen. Also braucht der ´ne Kutsche. Da die Bitburg hoch oben auf dem Drachenfels steht, brauchte er starke Pferde. Wenn du schon mal einen Schlitten einen Berg hochgezogen hast, weißt du, was das für eine Maloche ist. Also viele Pferde. Die wollen auch fressen und versorgt werden, also sind Knechte und Mägde nötig und so weiter. Kosten, überall Kosten und das selbstverständlich nur zum Wohle des Volkes. Um es kurz zu machen: der König war pleite. Die ganze Kohle auf den Kopf gehauen. Ein Plan war nötig. Ohne das Wachstum zu gefährden. So war es damals, so ist es heute. Nur haben wir heute keinen König und Hof mehr, sondern viele Könige. Die wählen dann und wann einen Oberkönig, der darf dann auch mal sagen: „Ich bin aber der Oberkönig.“ Dann machen entweder alle so weiter wie vorher oder sie wählen einen neuen Oberkönig. Früher war König sein noch viel schöner, da hatten die noch Kronen.

Die wurden aber nur zu besonderen Anlässen getragen. Heute haben die Könige rote Nasen, die sie aber nur einmal im Jahr tragen – selbst darauf ist kein Verlass. Pferdekutschen gibt es nicht mehr, statt dessen fliegen die Könige mit Raumschiffen über ihr Volk, das verschafft eine bessere Übersicht. Aus dem Adel wurden Beamte. Und alle sind immer noch damit beschäftigt, Vorschriften zu machen und auf das Wachstum zu achten. Damit das arme Volk weiß, dass es nicht allein ist. Pleite sind die Könige immer noch. Aber das macht nix mehr. Da heute keine knappen Goldstücke zum Zahlen genutzt werden, sondern bunt bedrucktes Papier, wird einfach mehr buntes Papier gedruckt – heute Geld genannt – und leiht sich das Geld bei denen, die noch gar nicht geboren sind. Wenn du meinst das geht nicht, liegt das daran, dass du eben nicht zum König taugst. Könige wissen, wie das geht.

Heute gibt es sogar Königinnen. Ob die Raumschiff oder Besen fliegen, ist nicht bekannt.

Der Text oben markiert den Anfang, in der Fortsetzung wird es zunehmend technischer. Die Leichtigkeit des Lernens soll jedoch erhalten bleiben, dazu dienen die diversen Visualisierungen, z.B. auf der eMag-Seite [bes.t 01](#). Mit der richtigen Anordnung der Grafiken wird der Programmablauf dargestellt und der entsprechende Code ausgegeben.

rezepte: veggie

Diese Rezepte sind kein Glaubensbekenntnis: ich bin kein Vegetarier. Ich esse Fleisch, allerdings wenig und bisweilen wochenlang gar nicht. Grundsätzlich Fleisch zu essen schließt nicht aus, vegetarische oder vegane Küche in den Speiseplan zu integrieren.

Bulgursalat

Zutaten:

- 250 g Bulgur
- 80 g Zwiebeln
- 50 g Stangensellerie
- 125 g rote oder gelbe Paprika
- 4 EL byodo Lemon Balsam, alternativ Condimento Bianco mit dem Saft einer Limette
- 1 TL Gemüsebrühe
- Salz
- Pfeffer
- Estragon (Sonnentor oder Lebensbaum), alternativ Zitronenverbene

Zubereitung:

- Bulgur im Wasserbad spülen, bis das Wasser klar ist
- Gemüse abwaschen
- Bulgur in Wasser mit der Gemüsebrühe kurz aufkochen, bis das Wasser aufgesogen ist
- Bulgur abkühlen lassen
- Zwiebeln klein schneiden
- Paprika in Würfel schneiden
- Stangensellerie in sehr feine Scheiben schneiden
- Essig unterrühren und Gewürze nach Geschmack hinzugeben



3: Bulgursalat im Einkochglas.

Rote Bete mit Pastinaken

Zutaten:

- 1 Glas (ca. 300 gr) eingelegte rote Beete
- 1 -2 Pastinaken, je nach Größe

- ca. 250 gr Naturjoghurt
- Salz
- Pfeffer
- Garam Masala (Sonnentor)



Je nach gewünschter Geschmacksrichtung entweder mehr Pastinaken oder mehr rote Bete zugeben. Bei dem Naturjoghurt lohnt das Probieren. Mein persönliche Vorliebe ist derzeit der Joghurt nach griechischer Art mit 2 % oder besser mit 10 % Fett.

Zubereitung:

1. rote Beete würfeln
2. Pastinaken schälen und bis zum harten Kern fein raspeln
3. rote Beete, Pastinaken und Joghurt vermischen und würzen
4. Mischung kühl stellen
5. vor dem Verzehr einen Tag ziehen lassen



1: ge"Weck"t bleibt er einige Tage haltbar.



Der Kern ist weniger zart als seine umgebende Hülle. Wer ihn verwenden will, sollte ihn je nach Zustand entweder dünsten oder länger ziehen lassen.



Herstellerangaben: wenn ein Geschmack präzise beschrieben werden soll, geht das nicht ohne die verwendeten Zutaten (die je nach Hersteller unterschiedlich schmecken) und je nach Rezept auch nicht ohne Mengenangaben.

Zu den Gewürzen:

- Estragon schmeckt je nach Hersteller reichlich unterschiedlich, deswegen zwei Angaben von Lieferanten mit vergleichbarem Geschmack.
- Die Chance, Zitronenverbene im Supermarkt zu finden ist ähnlich hoch wie die Chance auf einen Volltreffer im Lotto. Ich habe meine Zitronenverbene beim Wurzelsepp in Nürnberg gekauft, die (nicht nur dort) fälschlicherweise als Eisenkraut benannt ist.

Links

- [tolaris.de | Bausatz Arduino I](#)
- [tolaris.de | eLearning](#)

Impressum

Verantwortlich für den Inhalt:

Dipl.-Ing. Wolfgang Henderkes | tolaris technik e.V. | Eppenhauser Str. 85 | 58093 Hagen

Internet: <https://www.tolaris.de>

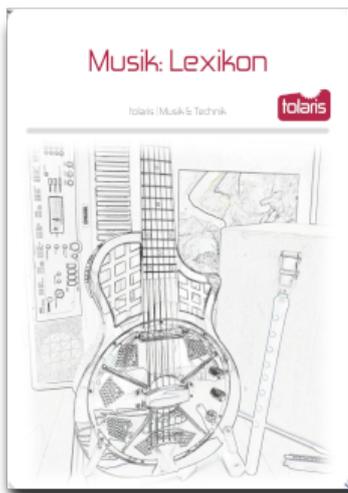
E-Mail: info@tolaris.de

Creative Commons Lizenz



Diese Publikation von [tolaris know-how](#) ist lizenziert unter einer [Creative Commons Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International](#) Lizenz.

Literaturhinweise



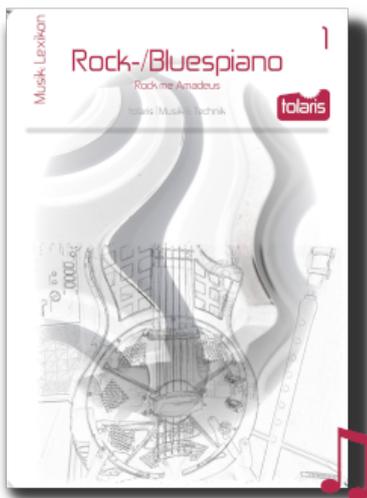
Musik: Lexikon

2022 - I

PDF-Dokument als Soloausgabe:

- mit dem Hintergrundwissen zur Musiktheorie in verständlicher Form
- Beispiele mit der **tolaris-Tasten-Organisation (toTO)** und **tolaris-Akkord-Organisation (tAO)**
- gezielter Einsatz von Farben als Mittel zur Informationsaufnahme
- für Gitarre und Piano
- für die Musik des 20. und 21. Jahrhunderts

Musik: Lexikon ist auch als Verbundausgabe mit anderen Publikationen lieferbar.



Rock-/Bluespiano 1

Rock me Amadeus

Musik: Lexikon

PDF-Dokument als Verbundausgabe [Rock-/Bluespiano 1](#) + [Musik: Lexikon](#) mit Verlinkung. Wenn du wissen willst, was du da machst.

Alle Musikprojekte werden mit und ohne [Musik: Lexikon](#) angeboten. Jeder Verbundausgabe liegt ein Gutscheincode bei, mit dem eine weitere Verbundausgabe preiswerter wird.



Rock-/Bluespiano 1

Rock me Amadeus

PDF-Dokument als Soloausgabe mit:

- 10 Licks in **A** und den gleichen Licks transponiert in **E**
- 10 Musikstücken mit mehr als 25 Varianten und einer Anleitung, eigene Varianten zu erstellen
- Link zum Download der MuseScore-Dateien

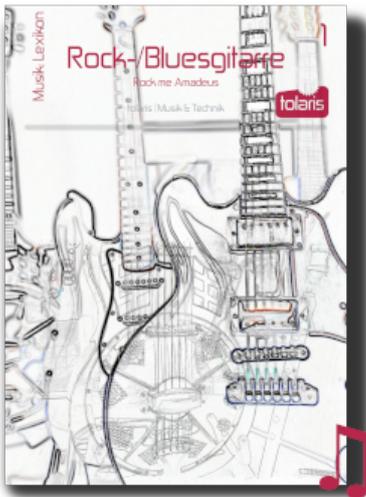
[Rock-/Bluespiano 1](#) ist auch als Verbundausgabe mit dem [Musik: Lexikon](#) lieferbar.



Rock-/Bluespiano 1

Notenblätter

PDF-Dokument mit den Notenblättern zum Ausdruck oder zur Ansicht am Bildschirm.



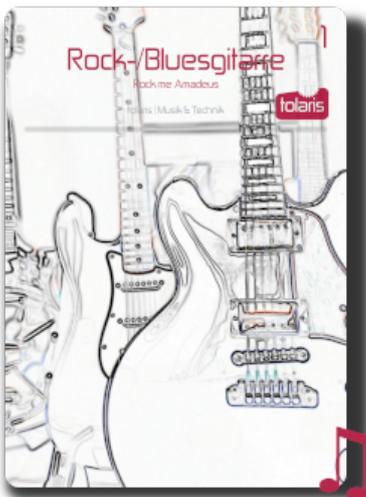
Rock-/Bluesgitarre 1

Rock me Amadeus

Musik: Lexikon

PDF-Dokument als Verbundausgabe [Rock-/Bluesgitarre 1](#) + [Musik: Lexikon](#) mit Verlinkung. Wenn du wissen willst, was du da machst.

Alle Musikprojekte werden mit und ohne [Musik: Lexikon](#) angeboten. Jeder Verbundausgabe liegt ein Gutscheincode bei, mit dem eine weitere Verbundausgabe preiswerter wird.



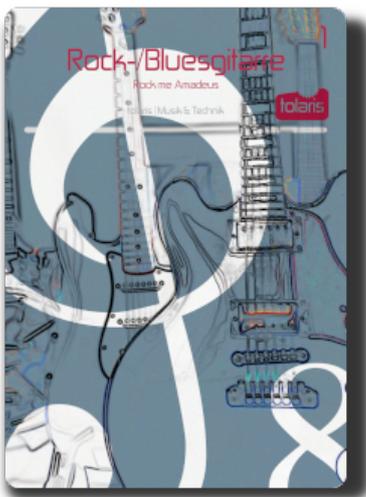
Rock-/Bluesgitarre 1

Rock me Amadeus

PDF-Dokument als Soloausgabe mit:

- 10 Licks in **A** und den gleichen Licks transponiert in **E**
- 10 Musikstücken mit mehr als 25 Varianten und einer Anleitung, eigene Varianten zu erstellen
- Link zum Download der MuseScore-Dateien

[Rock-/Bluesgitarre 1](#) ist auch als Verbundausgabe mit dem [Musik: Lexikon](#) lieferbar.



Rock-/Bluesgitarre 1

Notenblätter

PDF-Dokument mit den Notenblättern zum Ausdruck oder zur Ansicht am Bildschirm.

best

01.8



tolaris technik e.V.
Eppenhauser Str. 85 | 58093 Hagen
www.tolaris.de