



Beispielkonzept für das Lernfeld 7

Ausbildungsberuf	Informationselektroniker
Fach	Systemkonfiguration und Programmierung
Lernfeld	LF 7: Computersysteme konfigurieren und einrichten
Lernsituation	Lernsituation 1: Eine Single Board Computer gesteuerte Wildkamera realisieren
Zeitraumen	24 Unterrichtsstunden
Benötigtes Material	Tafel, Wacom Pen-Display, digitales Endgerät, Projektionstechnik, Informationsblätter, Leittexte, Schülerrechner, Mikrocontroller basierte Minicomputer, Versuchsaufbauten zur Simulation, Kameras, LEDs, Bewegungsmelder
Querverweise	LF 4

Konzeptionsmatrix für die Lernsituation 1

Konzeptionsmatrix für Lernsituation 1		Die S.u.S. informieren sich über Single Board Computer und mögliche Programmiersprachen. Sie erstellen Pseudoskripte und setzen diese in funktionsfähige Programme um. Die S.u.S. befassen sich mit Schnittstellen und erkennen den Nutzen von Ein- und Ausgängen. Sie bauen Schaltungen auf und testen Ihre Programme. Die S.u.S. vernetzen Single Board Computer und starten Ihre Skripte automatisch. Die S.u.S. erhalten einen Kundenauftrag zur Planung und Durchführung einer mit Single Board Computer gesteuerten Wildkamera. Komponenten werden eigenständig ausgewählt (z.B. IR-Sensor, IP-Kamera, Raspberry Pi) und vernetzt. Zur Visualisierung der Kamerafunktion werden über LEDs die Betriebszustände angezeigt. Kundenwünsche bezüglich Zeit der Videos bzw. Anzahl der Bilder bei Betätigung des Sensors werden berücksichtigt. Die Schaltung sowie die benötigten Bauteile werden aufgebaut und gegebenenfalls berechnet. Das gewünschte Skript wird verfasst und getestet. Die anschließende Kundenübergabe wird im Zusammenhang mit einer kurzen Beschreibung zum Programmstart bzw. Programmende durchgeführt.						
Zeit	Thema/ Beschreibung	Sachwissen	Prozesswissen	Reflexions- wissen	Aufgabe			
					Aktivitäten	Lernprodukte	Medien/ Materialien	Kontroll- und Reflexionselemente
30	Vorstellen der Problemstellung	-	Analyse des Kundenwunsches und Festlegung der benötigten Schnittstellen	-	Erstellung einer digitalen Mindmap			
					S.u.S. protokollieren den Arbeitsauftrag und diskutieren mögliche Lösungsansätze	Mindmap	<u>Präsentation:</u> Vorstellung der Lernsituation mittels Auftragsmail <u>Präsentation:</u> Arbeitsauftrag mit Blockschaltbild per PowerPoint Präsentation	S.u.S. prüfen den Kundenauftrag auf eventuelle Schwachstellen, die berücksichtigt werden müssen. Sie achten darauf, dass alle Kundenwünsche berücksichtigt worden sind.

105	Orientierung über die geforderten Ressourcen	<u>Hardware:</u> EVA – Prinzip Peripheriegeräte USB – Schnittstelle RJ – 45 – Schnittstelle HDMI – Schnittstelle CPU <u>Anwendung:</u> Programmiersprachen	Wählen gemäß dem Kundenwunsch und dem Betriebssystem passende Software aus.	-	Auswahl der benötigten Ressourcen anhand der Kundenanforderungen			
					Gruppenarbeit: Wählen Ressourcen gemäß dem Kundenauftrag aus. Beachten dabei die benötigten Schnittstellen und die erforderliche Leistung der CPU. Prüfen preisliche Aspekte und beschäftigen sich mit möglichen Programmiersprachen. Entscheiden sich für eine Netzwerkstruktur und legen diese fest.	Preis-/Leistungsverhältnis der Materialien einschätzen können. Erhalten Informationen über die Anwendung diverser Schnittstellen.	Fachzeitschriften Mobiltelefone Tablets PCs Geeignete Software	Die jeweiligen Gruppen präsentieren ihre Ergebnisse und erörtern die zugehörigen Gedankengänge. Die Gruppen erarbeiten dabei eine finale Materialliste und verwenden diese über den weiteren Verlauf der Lernsituation.
220	Informieren sich über die Funktion und den Aufbau der benötigten Schnittstellen, sowie die Grundlagen von Programmiersprachen und das passende Betriebssystem.	<u>Hardware:</u> EVA – Prinzip Peripheriegeräte USB – Schnittstelle RJ – 45 – Schnittstelle HDMI – Schnittstelle CPU <u>Betriebssystem:</u> Linux/Debian-Derivat <u>Software:</u> Python	Wählen gemäß dem Kundenwunsch und dem Betriebssystem passende Software aus.	-	Dokumentation der relevanten Bauteile und Einarbeitung in die Grundlagen der Programmierung			
					Recherche über die Funktion der benötigten Bauteile und deren Anwendung. Spielerische Einführung in die Programmierung durch Erstellen eines Skriptes für ein Autorennen unter	Übersicht über Bauteile und deren Funktion. Erstellung eines ersten Skriptes.	Informationsblätter Fach/-Tabellenbücher Internetplattformen Internet-Tutorials PCs, Tablet, etc. ...	Die Auszubildenden stellen ihre Skripte dem Plenum vor und optimieren ggf. ihre Programme. Sie wenden USB-Speichermedien an,

					Anwendung von Blockprogrammierung.			um die Skripte abzuspeichern.
155	Die Auszubildenden planen die Umsetzung des Kundenauftrages	<u>Hardware:</u> EVA – Prinzip Peripheriegeräte USB – Schnittstelle RJ – 45 – Schnittstelle HDMI – Schnittstelle CPU <u>Betriebssystem:</u> Linux/Debian-Derivat <u>Software:</u> Python-Interpreter <u>Visualisierung:</u> Technologieschema Stromlaufplan mit Schaltzeichen	Planen die korrekte Struktur zur Einrichtung von Netzwerk und Peripheriegeräten mit Hilfe von Blockschaltbildern.	Physikalische/-technische Hintergründe zu den jeweiligen Betriebssystemen, sowie die Netzwerkanbindung zu Peripheriegeräten und Hardwarekomponenten.	Erstellen einen strukturierten Projektplan mit Hilfe von Schaltplänen und Blockschaltbilder			
					Erstellen eine Planungsdocumentation mit geeigneter Software, gegebenenfalls CAD-Programme, und fügen diese in den bereits vorhandenen Projektordner mit der Materialliste hinzu.	Dokumentation der Planung und Erstellung einer sinnvollen Ordnerstruktur.	PCs inkl. geeigneter Software.	Plenumsdiskussion der individuellen Planungen, gegebenenfalls Optimierung dieser Pläne unter Verwendung kompetenter Argumentation.
200	Die Auszubildenden führen den Kundenauftrag aus und testen die Anlage am Steckbrett	<u>Hardware:</u> EVA – Prinzip Peripheriegeräte USB – Schnittstelle RJ – 45 – Schnittstelle HDMI – Schnittstelle CPU <u>Betriebssystem:</u> Linux/Debian-Derivat	-	Physikalische/-technische Hintergründe zu den jeweiligen Betriebssystemen, sowie die Netzwerkanbindung zu Peripheriegeräten	Führen den Auftrag gemäß dem Kundenwunsch durch			
					Stellen Netzwerkverbindung zum Minicomputer her. Legen eine Ordnerstruktur an und Dokumentieren die Programmschritte im Skript. Bauen die geforderte Schaltung inkl. Peripheriegeräte am	Eigenständige Umsetzens Umsetzen einer Handlungsorientierten Aufgabe.	Minicomputer; PC; Informationsblätter aus vorangegangenen Phasen; Steckbrett und Peripherie;	Die S.u.S. prüfen ihr fertiggestelltes Produkt anhand der im Vorfeld angelegten Pflichten und dokumentieren die

		<u>Software:</u> Python/Debian-Derivat <u>Visualisierung:</u> Technologieschema Stromlaufplan mit Schaltzeichen		en und Hardwarekomponenten.	Schaltbrett auf. Simulieren das fertige Produkt.		ggf. Internet für Syntaxrecherche	Bedienung der Anlage in kurzen Sätzen.
135	Die Auszubildenden präsentieren Ihre Schaltungen	<u>Hardware:</u> EVA – Prinzip Peripheriegeräte USB – Schnittstelle RJ – 45 – Schnittstelle HDMI – Schnittstelle CPU <u>Betriebssystem:</u> Linux/Debian-Derivat <u>Software:</u> Python-Interpreter <u>Visualisierung:</u> Technologieschema Stromlaufplan mit Schaltzeichen		Physikalische/technische Hintergründe zu den jeweiligen Betriebssystemen, sowie die Netzwerkanbindung zu Peripheriegeräten und Hardwarekomponenten.	Präsentation der individuell gestalteten Projektlösung			
					Stellen unter zu Hilfenahme der gegebenen Informationen eine Präsentation für das Plenum zusammen.	Sprachkompetenz; Fachkompetenz	Lehrer-PC; ggf. Dokumentenkamera; Beamer; Individuelle Schülermaterialien zur Präsentation	Präsentation des Produkts vor dem Plenum Handout zu der Präsentation
135	Optimieren das individuelle Vorgehen bei der	<u>Teamfähigkeit:</u> Diskussionen in Gruppen	Erkennen strukturiertes Vorgehen (roter Faden) von der Auftragsannahme	Optimierung der eigenen Projektbearbeitung, sowie die Struktur für	Optimierung der Anlage bzw. der Herangehensweise zur Problemstellung			
					Die Auszubildenden vergleichen in Gruppen Ihre Arbeitsweisen und optimieren zusammen das	Die S.u.S. erstellen einen Leitfaden für	Unterlagen der Auszubildenden	Vorstellung der erarbeiteten Optimierungen.



	Projektdurchführung		bis hin zur Fertigstellung	Projektumsetzung im Team.	Vorgehen, um einen möglichen Zeitgewinn oder eine Kostenersparnis zu erhalten.	die Projektumsetzung		
135	Erstellung der Rechnung für den Kunden und Übergabe mit kurzer Beschreibung		-	-	Kundenübergabe des Projektes mit kurzer Bedienungsanleitung und Erstellung der Rechnung			
					Die Auszubildenden erstellen ein normgerechtes Rechnungsschreiben und Verfassen eine Kurzanleitung für den Betrieb der Anlage.		Datenverarbeitungsprogramme; Internet	Übergeben dem Kunden die Kurzanleitung und senden per E-Mail das Rechnungsschreiben.
	EMV	<u>EMV</u> : Was ist EMV? Verbindung zu CE - Kennzeichen	EMV – Testhaus für Firmenbesichtigung	Optimierung der eigenen Projektbearbeitung, sowie die Struktur für Projektumsetzung im Team.	Erstellung einer Merkhilfe zum Thema EMV.			
					Arbeitsteilige Gruppenarbeit zur EMV mit anschließenden Präsentationen der Arbeitsergebnisse.	Merkhilfe EMV	<u>Informationsmaterial</u> <u>DIN-Norm</u>	Präsentationen Handout zu den Präsentationen
60	Schulaufgabe							

Unterlagen, Medien, Materialien



Sehr geehrte Damen und Herren,

ich benötige für die Hauseinfahrt meiner Firma ein Überwachungssystem. Dieses soll über einen Bewegungssensor angesteuert werden und sowohl tags als auch nachts einsatzbereit sein. Vorerst würde es mir genügen, wenn dieses System Bilder von der Auslösestelle liefert. Sollte ich für meine Zwecke ganze Zeitabläufe benötigen, müsste das System leicht umprogrammierbar sein. Außerdem möchte ich die Möglichkeit haben das System per Knopfdruck Ein/-Auszuschalten.

Ich habe mich im Internet über Mikrocontroller informiert und war von den diversen Einsatzbereichen und die schnelle Art der Programmierung begeistert. Daher würde ich Sie bitten mir eine Steuerung mittels dieser Technik zu initialisieren und zu installieren.

Hintergrund hierfür ist, dass ständig Tiere über meine Fahrzeuge laufen und eventuelle Schäden verhindert werden können.

Ideensammlung mit oncoo.de Mindmap (Diskussion im Plenum)



1. Orientieren Sie sich an der E-Mail-Anfrage und fassen Sie den tatsächlichen Kundenauftrag in eigenen Worten zusammen. Nehmen Sie die von Ihnen erstellte Mindmap zu Hilfe. (5 min)



2. Bilden Sie 4er-Arbeitsgruppen und erarbeiten eine stichpunktartige To-do-Liste. Informieren Sie sich in diesem Rahmen über mögliche Betriebssysteme, geeignete Mikrocomputer sowie die benötigten Schnittstellen und Netzwerkanbindungen. (25 min)

3. Sie haben sich mittlerweile die benötigten Schnittstellen für das Projekt vor Augen geführt. Der nächste Arbeitsschritt ist die Funktion sowie die Anwendung dieser Schnittstellen zu analysieren. (25 min)

• USB (Verwenden Sie die Infoblätter zum Thema Schnittstellen als Informationsmaterial)

- Nennen Sie die gängigen USB-Schnittstellen.

USB 2.0; USB 3.0 ; USB 3.1 ; USB 3.2; USB 4; USB Mini ; USB Micro (Typen A, B, C..)

- Wo liegen beim USB2 – Standard die Unterschiede zwischen 2.0, 3.0 und 3.1 Gen2?

USB 2.0 (High Speeds 480 MBit/s)

USB 3.0 (Super Speed 5 GBit/s)

USB 3.1 Gen 2 (Super Speed Plus 10 GBit/s)

- Nennen Sie die Informationen, die beim USB-Stecker durch die Farben gelb, blau und rot gegeben sind.

Blau: 3.0 Port -> Sehr schnelle Übertragungsrates für kompatible Geräte

Gelb: Port verfügt über permanente Stromversorgung

Rot: Hier kann es sich um USB 2.0 bzw. USB 3.0 handeln (Handbuch Mainboard)

• HDMI (Verwenden Sie das Internet sowie Fachbücher als Informationsquellen)

- Nennen Sie Anwendungsbereiche für HDMI-Schnittstellen.

Die HDMI – Schnittstelle wird für die digitale Bild- und Ton-Übertragung in der

Unterhaltungselektronik verwendet. Sie übertragen Bild und Ton in einer Leitung.

Vorteil gegenüber anderen Übertragungsmöglichkeiten ist die hohe Qualität.

- Geben Sie Vorteile von HDMI-Verbindungen gegenüber RGB-Verbindungen an.

Die RGB – Technik verfügt über eine Farbtiefe von 8 Bit pro Komponente.

Die HDMI – Technik kann Farbtiefen von 10 Bit, 12 Bit und 16 Bit pro

Farbkomponente unterstützen.

- DVI (Verwenden Sie das Internet sowie Fachbücher als Informationsquellen)
 - Nennen Sie die Anwendungsbereiche für DVI-Schnittstellen.

Sie findet Anwendung als Bindeglied zwischen Grafikkarte des PC's und dem

Monitor. Verwendung ist sowohl für analoge als auch digitale Signale möglich.

 - Geben Sie Vorteile von DVI – Schnittstellen an.

Sie kann als Universalschnittstelle verwendet werden, da sie analoge und digitale

Datenübertragung zulässt. Zudem sind DVI-D und DVI-I elektrisch kompatibel zu

HDMI 1.0. DVI-D kann mit passiven Adaptern an HDMI betrieben werden.

- 4. Es wurde vom Kunden gefordert ein geeignetes Betriebssystem für den Minicomputer zu wählen. Nennen Sie die drei gängigsten Betriebssysteme, die am Markt sind. Bringen Sie diese, falls möglich, mit den Herstellern in Verbindung. (10 min)
 - Windows Betriebssystem / Microsoft

 - iOS Betriebssystem / Apple

 - Linux Betriebssystem / Kernel ist Open Source – Ubuntu z.B. Cononical

 - Geben Sie nun zu jedem Betriebssystem ein Anwenderbeispiel an und begründen Ihre Entscheidungen.
 - Windows: Findet Anwendung bei Privatpersonen und Firmen, ist mit den

 - meisten Softwarekomponenten kompatibel

 - iOS: findet Anwendung bei Privatpersonen und im amerikanischen Bereich

 - in Firmen. Grund für Privatpersonen, automatische Synchronisierung aller Geräte.

 - Linux: findet Anwendung bei Privatpersonen und Firmen mit IT-Verständnis,

 - Grund hierfür ist die spezielle Oberfläche und der fast störungsfreie Betrieb.

5. Für unseren Kunden benötigen wir ein möglichst reibungsfreies Betriebssystem, mit dem es üblich ist zu programmieren. Welches der drei Systeme bietet sich hierfür an? Begründen Sie kurz Ihre Aussage. (10 min)

Unser Kunde möchte seine Anlage speziell für seine Bedürfnisse programmiert haben,

zudem möchte er eine Gewährleistung für einen möglichst reibungsfreien Betrieb.

Aus diesem Grund würde sich das Betriebssystem Linux anbieten. Hier würde der

Anwender nicht direkt mit dem System in Berührung kommen, somit kann auf die ganzen

Hilfestellungen von Windows und iOS verzichtet werden.

6. Aufgrund unserer gefundenen Fakten suchen wir den geeigneten Minicomputer aus. Geben Sie diesen an und begründen Ihre Entscheidung in Bezug auf dem Kundenwunsch. Geben Sie gegebenenfalls auch weitere Geräte an, mit denen das Projekt realisierbar ist. (5 min)

Aufgrund der Kundenanforderungen und des möglichen Ausbaues der Anlage würde

sich der Raspberry Pi anbieten, da er leicht zu programmieren ist und zudem gute

Datenübertragungsraten sowie Rechnerleistung bietet. Zusätzlich ist er klein und

im Kosten-/Leistungsbereich zu empfehlen.

Eine Alternative wäre die Lösung mit einem Arduino, dieser kann ebenfalls mit

einer bekannten Programmiersprache programmiert werden, liefert allerdings eine nicht

ganz so hohe Rechnerleistung als der Raspberry Pi.



Legen Sie sich eine geeignete Ordnerstruktur für dieses Projekt an und kennzeichnen Sie diesen Abschnitt mit „Orientierung/- Informationen zum Kundenauftrag“.

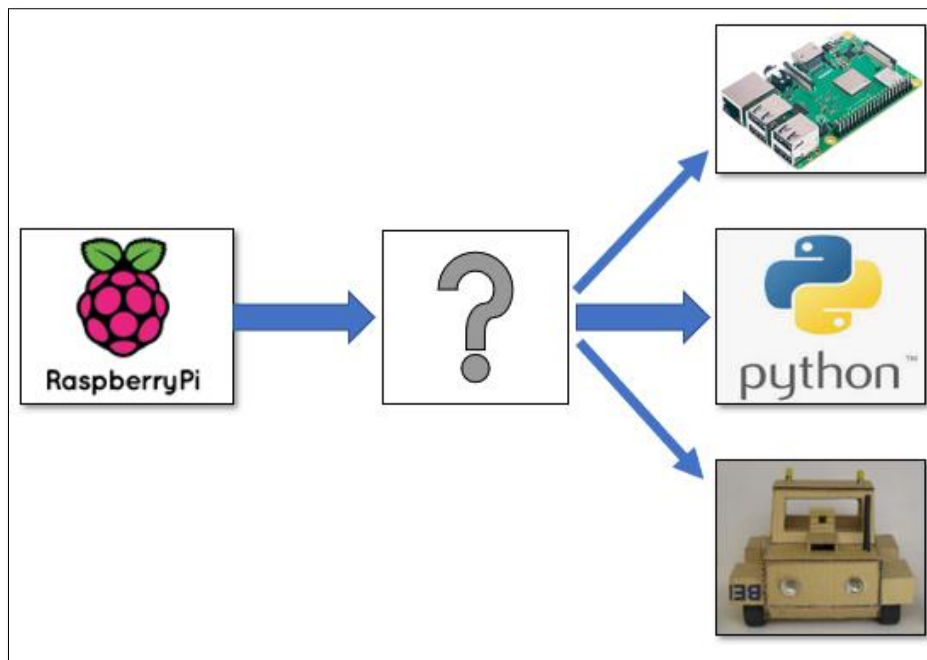


Sehr geehrte Schülerinnen und Schüler,

nachdem wir uns für den Minicomputer Raspberry Pi entschieden haben, müssen wir uns mit dem Gerät vertraut machen. Wie uns bekannt ist, können mit dem leistungsfähigen Minicomputer viele Projekte verwirklicht werden. Zudem ist in den Internetforen immer wieder der Begriff Python erwähnt worden. Bearbeiten Sie als nächstes die typischen Kundenanfragen zu diesem Thema. Senden Sie im Anschluss Ihrem Vorgesetzten eine E-Mail mit den grundlegenden Informationen zu diesem Gerät. Gehen Sie dabei sowohl auf praktische Anwendungsgebiete und technische Informationen als auch auf Hard/- Software ein.

Mit freundlichen Grüßen

Elektro GmbH



1. Orientieren Sie sich an der E-Mail Ihres Vorgesetzten und notieren sich die geforderten Informationen stichpunktartig.

(5 min)



2. Informieren Sie sich über die Merkmale eines RPi und geben Ihre Ergebnisse in eigenen Worten wieder. (30 min)

- Definition Raspberry Pi

Ein RPi ist ein extrem preiswerter, kreditkartengroßer Computer.

Es hat einen sehr geringen Stromverbrauch und eine SD-Karte als Peripherie-

Speicher. Es kann unter anderem mit der Programmiersprache Python programmiert

werden (für eine Blockprogrammierung steht Scratch zur Verfügung).

- Im Laufe Ihrer Recherche stoßen Sie auf die Schlagwörter interaktives Exponat, System mit speziellen Eingabegeräten, mobiles Gerät und Server für spezielle Aufgaben. Worin liegt der Unterschied bei diesen Hardwarekonfigurationen?

<p>Interaktives Exponat</p>  <p>Abbildung 1:[Quelle]: http://www.raspberry-pi-geek.de</p>	<p>Auf dem RPi läuft ein interaktives Programm mit graphischer Benutzeroberfläche. Das kleine Gerät ist dabei auf die Rückseite eines großen Touchscreen geklebt. (z.B. Interaktive Türschilder)</p>
<p>System mit Spezial-Eingängen</p>  <p>Abbildung2:[Quelle]:AD-Wandler www.mouser.de</p>	<p>Der RPi ist mit Sensoren verbunden, der Computer reagiert auf die Sensorsignale. Hiermit lassen sich z.B. Spiele mit Hilfe von AD-Wandlern, Widerständen, etc. programmieren.</p>
<p>Mobiles Gerät</p>  <p>Abbildung 3[Quelle]developer-blog.net</p>	<p>Aufgrund der geringen Größe und des geringen Stromverbrauchs, eignet sich der RPi gut für mobile Geräte und autonome Roboter (z.B. als IR-Messgerät).</p>
<p>Server für spezielle Aufgaben</p>  <p>Abbildung 4:[Quelle]http://www.raspberry-pi-geek.de</p>	<p>Die geringen Anschaffungskosten sowie die niedrige Nennleistung von 3,5W (Modell B) sind Faktoren, die den Einsatz des RPi für Serveranwendungen und Hotspots begünstigen.</p>

3. Führen Sie die Beschriftung der Hardwarekomponenten des Raspberry Pi Typ B/ B+ durch.

(15 min)

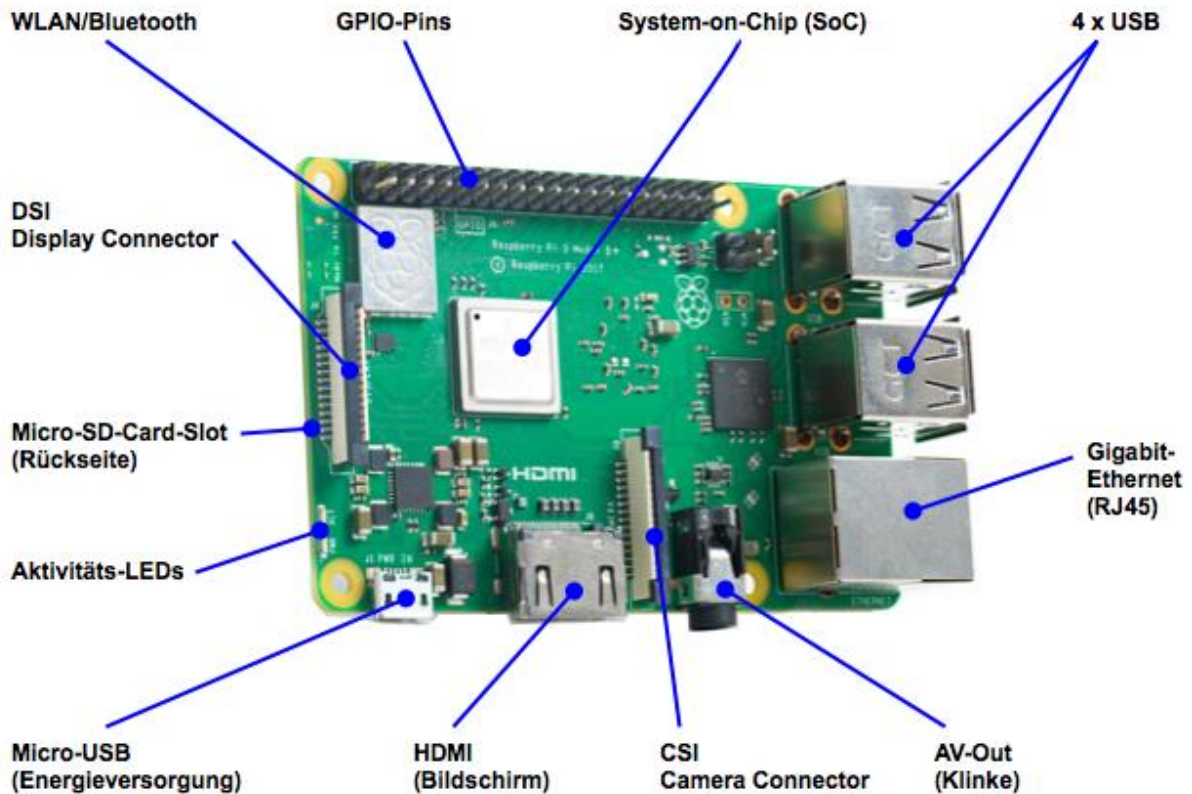


Abbildung 5:[Quelle]www.elektronik-kompodium.de

4. Neben der visuellen Programmierung „Scratch“ zum Einstieg in die Programmierwelt wird der Raspberry Pi in den meisten Fällen mit Python programmiert. Informieren Sie sich über diese Programmiersprache im Zusammenhang mit dem Raspberry Pi und beantworten die folgenden Fragen. (30 min)

- Zur Standardinstallation von Python gehört eine integrierte Entwicklungsumgebung namens IDLE, welche im Wesentlichen aus drei Komponenten besteht. Beschreiben Sie diese kurz.

Die Python Shell	Die Shell ist eine Anwendung, mit der auf der Kommando-
	zeile einzelne Python-Anweisungen eingegeben und
	ausgeführt werden können. Ein Python-Programm,
	welches eine Bildschirmausgabe liefert, gibt diese in einem
	Shell Fenster aus.

Der Programmierer	Ist eine Art Textverarbeitungsprogramm zum Schreiben von
	Programmen. Er wird vom Shell Fenster aus (File / NewFile)
	gestartet.
Der Debugger	Er dient dazu, den Lauf eines Programms zu kontrollieren
	und zu überwachen, um logische Fehler zu finden.

- Geben Sie die Bedeutung der folgenden Operatoren an.

**	Potenz, $x^{**}y = x^y$
*, /, //	Multiplikation, Division und ganzzahlige Division
%	Modulo-Operation. Der Rest einer ganzzahligen Division
+, -	Addition und Subtraktion

- Für nahezu alle Programme wird die Eingabe von Zahlen benötigt. Geben Sie die Zahlenformate folgender Bezeichnungen an.

Ganze Zahlen	integer
Gleitpunktzahlen	float
Komplexe Zahlen	complex



5. Schreiben Sie die geforderte E-Mail und verweisen Sie dabei auf das Arbeitsblatt im Anhang. Geben Sie zusätzlich Ihre eigene Meinung über die Anschaffung eines Raspberry Pi inklusive Begründung an.

Ordnen Sie diese Unterlagen unter „Informationssammlung Kundenauftrag Wildkamera_1“ ein.

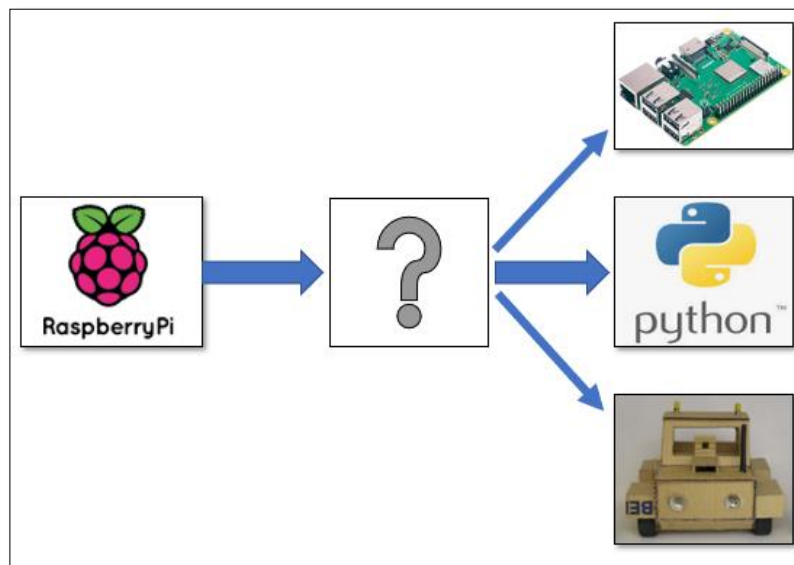
Sehr geehrte angehende Facharbeiter und Facharbeiterinnen,

der Kunde war mit Ihrer Beratung sehr zufrieden und hat sich entschieden das Projekt mit dem Auftrag freizugeben. Bereiten Sie sich auf das Programmieren mit Python vor, indem Sie einfache mathematische Berechnungen sowie die erste Programmierung einer LED durchführen.



Mit freundlichen Grüßen

Elektro GmbH



1. Orientieren Sie sich an der E-Mail des Kunden und notieren sich die geforderten Informationen stichpunktartig.

Aufgrund eines Kundenauftrages sollen wir uns in die Programmstruktur

des RPi einarbeiten (Python). Um unsere Fortschritte zu testen, wird im

Anschluss eine LED – Ein/Aus programmiert und getestet.

Versuchen Sie sich die Bearbeitungszeit so einzuteilen, dass Sie in ca. 90 min mit dem Arbeitsblatt abschließen können.

Verwenden Sie als Informationsquellen Internetforen, Fachliteratur und Fachbücher.



2. Informieren Sie sich über die Programmiersprache Python und beantworten folgende Fragen.

- Nachdem Sie sich bereits über einige Operatoren erkundigt haben, weisen Sie nun einer Variablen z.B. a den Wert 3 zu.

```
>>> a = 3
```

- Aus welchem Grund führt folgende Eingabe zu einer Fehlermeldung?

```
>>> (1+1) (3+9) : Weil der Multiplikationsoperator * fehlt (>>>(1+1)*(3+9))
```

- Welche Funktion müssen Sie eingeben, um den absoluten (positiven) Wert einer Zahl x zu erfahren?

```
>>> abs(x)
```

- Nennen Sie die Funktionen, die Ihnen ermöglichen den Maximal-/ und Minimalwert der Zahlen von $0x, \dots, xn$ auszugeben.

```
>>> max(0x, ..., xn)
```

```
>>> min(0x, ..., xn)
```

- Die Eingabe der Befehle in Python Shell startet immer hinter drei spitzen Klammern `>>>`. Geben Sie an, welche Ausgabe nach folgenden Befehlen erfolgt.

Befehl	Ausgabe	Befehl	Ausgabe
<code>>>>2+2</code>	4	<code>>>>2+</code>	Syntax Error
<code>>>>a=125</code> <code>>>>a</code>	125	<code>>>>1+2*3+4</code>	11
<code>>>>(2+3)/2</code>	2.5	<code>>>>3/2</code>	1.5
<code>>>>3//2</code>	1	<code>>>>7%3</code>	1
<code>>>>2**3</code>	8	<code>2*3**2</code>	18
<code>>>>(2*3)**2</code>	36	<code>>>>type(1.2)</code>	<class'float'>

- Um den Raspberry Pi fachgerecht zu betreiben, benötigt man die exakte Belegung des GPIO. Ermitteln Sie mit Hilfe des Internet die erforderlichen Daten des GPIO für den RPi Model 3 B.

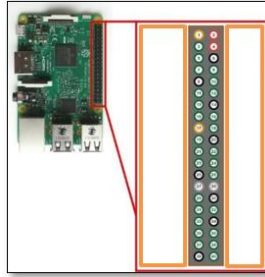


Abbildung 6: Quelle [schmidt-feldberg.de]





















Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I2C)		DC Power 5v	04
05	GPIO03 (SCL1 , I2C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)		(I2C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Abbildung 7: Quelle[raspberrypi-geek.de] GPIO Belegung

4. Nennen Sie die Bedeutung der folgenden Begriffe:

+3,3V	Spannungsquelle	(SCL)	Taktleitung
+5V	Spannungsquelle	(TXD)	UART-Empfang
GPIO 17	Ein/-Ausgänge	(RXD)	UART-Senden
GND	Ground	(MOSI)	M-Output/S-Input
(SDA)	Datenleitung	(MISO)	M-Input/S-Output
(GPCLKO)	Getakteter I/-Outp.	(SCLK)	Serial Clock
ID_SD	Datenaustausch mit Erweiterungsmodul	ID_SC	Datenaustausch mit Erweiterungsmodul

UART => Senden und Empfangen von Daten über Datenleitung; MOSI/MISO => Master – Slave Prinzip

SCLK => Wird vom Master als Synchronisation ausgegeben

5. Geben Sie die Vorgehensweise bei der Erstinbetriebnahme des Raspberry Pi an.

NOOBS auf die SD-Karte kopieren -> Peripheriegeräte an den RPi anschließen ->

RPi an Stromversorgung anschließen -> Wählen Sie Raspbian aus -> Nehmen Sie

die Spracheinstellung vor -> Betätigen Sie den Button install -> Nach Neustart,

Einstellungen -> RPi Configuration -> Boot -Option to Desktop

6. Beschreiben Sie das Vorgehen, den RPi über eine VNC – Verbindung mit dem Laptop zu verbinden. Worin liegt der Sinn einer solchen Verbindung?

In Betrieb genommenen RPi an benötigte Peripheriegeräte anschließen,

ssh aktivieren, VNC – Viewer öffnen und IP-Adresse anzeigen lassen,

VNC - Viewer am PC oder Laptop installieren, RPi per LAN mit PC verbinden,

IP-Adresse eingeben => Fernzugriff auf RPi über PC oder Laptop hergestellt



7. Schreiben Sie die geforderte E-Mail und verweisen Sie dabei auf das Arbeitsblatt im Anhang. Geben Sie zusätzlich Ihre eigene Meinung über die Anschaffung eines Raspberry Pi inklusive Begründung an.

Ordnen Sie diese Unterlagen unter „Informationssammlung Kundenauftrag Wildkamera_2“ ein.

Sehr geehrte Auszubildende,



nachdem Sie sich mittlerweile mit den mathematischen Regeln, sowie einer Auswahl der dazugehörigen Operatoren befasst haben, sollen Sie zusätzlich einen Onlinekurs an Ihrem Arbeitsplatz zum Thema RPi absolvieren. Geben Sie im Anschluss daran Ihrem Vorgesetzten per E-Mail ein kurzes Feedback über die erlernten Grundlagen.

Mit freundlichen Grüßen

Elektro GmbH

```
pi@raspberrypi:~/S0S $ gpio readall
-----Pi 3+-----
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 8 | SDA.1 | IN | 1 | 3 | 4 | | | 5v | | |
| 3 | 9 | SCL.1 | IN | 1 | 5 | 6 | | | 5v | | |
| 4 | 7 | GPIO.7 | IN | 1 | 7 | 8 | 0 | IN | TxD | 15 | 14 |
| | | | | | 9 | 10 | 1 | IN | RxD | 16 | 15 |
| 17 | 0 | GPIO.0 | IN | 0 | 11 | 12 | 0 | IN | GPIO.1 | 1 | 18 |
| 27 | 2 | GPIO.2 | IN | 0 | 13 | 14 | | | 0v | | |
| 22 | 3 | GPIO.3 | IN | 0 | 15 | 16 | 0 | IN | GPIO.4 | 4 | 23 |
| | | | | | 17 | 18 | 0 | IN | GPIO.5 | 5 | 24 |
| 10 | 12 | MOSI | IN | 0 | 19 | 20 | | | 0v | | |
| 9 | 13 | MISO | IN | 0 | 21 | 22 | 0 | IN | GPIO.6 | 6 | 25 |
| 11 | 14 | SCLK | IN | 0 | 23 | 24 | 1 | IN | CE0 | 10 | 8 |
| | | | | | 25 | 26 | 1 | IN | CE1 | 11 | 7 |
| 0 | 30 | SDA.0 | IN | 1 | 27 | 28 | 1 | IN | SCL.0 | 31 | 1 |
| 5 | 21 | GPIO.21 | IN | 1 | 29 | 30 | | | 0v | | |
| 6 | 22 | GPIO.22 | IN | 1 | 31 | 32 | 0 | IN | GPIO.26 | 26 | 12 |
| 13 | 23 | GPIO.23 | IN | 0 | 33 | 34 | | | 0v | | |
| 19 | 24 | GPIO.24 | IN | 0 | 35 | 36 | 0 | IN | GPIO.27 | 27 | 16 |
| 26 | 25 | GPIO.25 | IN | 0 | 37 | 38 | 0 | IN | GPIO.28 | 28 | 20 |
| | | | | | 39 | 40 | 0 | IN | GPIO.29 | 29 | 21 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
-----Pi 3+-----
```

Abbildung 8: Resultat des Befehls `gpio readall`



1. Orientieren Sie sich an der E-Mail des Kunden und notieren sich die geforderten Informationen stichpunktartig.

Um den richtigen Umgang mit RPi zu erlernen, sollen die folgenden Aufgaben

des „Online-Kurses“ bearbeitet werden. Im Anschluss daran muss dem

Vorgesetzten ein kurzes Feedback gegeben werden.

Planen Sie für die Aufgaben eins und zwei ca. 50 min ein. Die verbleibende Zeit sollten Sie der praktischen Anwendung und der Präsentation zukommen lassen. Verwenden Sie als Informationsquellen das Internet und die Fachliteratur.



2. Ihre Aufgabe ist es, ein Python-Skript zu erstellen. Informieren Sie sich in diesem Zusammenhang über das richtige Vorgehen.
- Geben Sie die Definition des Ausdrucks „Python-Skript“ an.

Allgemein bestehen Programme aus einer oder mehreren Anweisungen, die vom Computer (Raspberry Pi) nach dem EVA-Prinzip ausgeführt werden können. Diese Programme werden auch häufig als Skripte bezeichnet.

- Nennen Sie mögliche Bestandteile eines vollständigen Skriptes.

Eingabebefehle, Ausgabebefehle, Einbindung von Bibliotheken, Schleifen, Felder, Variablen, Belegung von Ein- und Ausgängen, Operatoren, Speicherort

- Geben Sie die Funktion folgender Eingabebefehle wieder.

import	Vorgefertigte Programme o.ä. können importiert werden.
import RPi.GPIO as GPIO	Das Modul RPi.GPIO wird importiert und dessen Inhalt ist über das Kürzel GPIO.<...> im Code erreichbar.
GPIO.setmode(GPIO.BOARD)	Modus Pin – Nummerierung wird zugewiesen
GPIO.setup(10, GPIO.OUT)	Der Pin 10 des GPIO wird als Ausgabekanal deklariert.
GPIO.output(10, GPIO.HIGH)	Der Pin 10 Ausgang wird mit True ausgegeben, das heißt es liegen +3,3 Volt an.
GPIO.output(10, GPIO.LOW)	Der Pin 10 Ausgang wird mit False ausgegeben, das heißt es liegen 0 Volt an.
sudo idle3	Python - Shell (IDLE) wird geöffnet.
print(" ")	Standardfunktion zur Ausgabe (hier Zeichenkette (String)) am Bildschirm.
print()	Standardfunktion zur Ausgabe von Ergebnissen in den Standardstream.
input([Prompt])	Der Benutzer soll eine Eingabe am Bildschirm vornehmen, welche im Skript verarbeitet wird.
d = int ()	Der Variablen d wird ein Wert vom Typ integer zugewiesen bzw. 1.2 in 1 umgewandelt.



3. Erstellen Sie ein Skript mit folgenden Vorgaben:

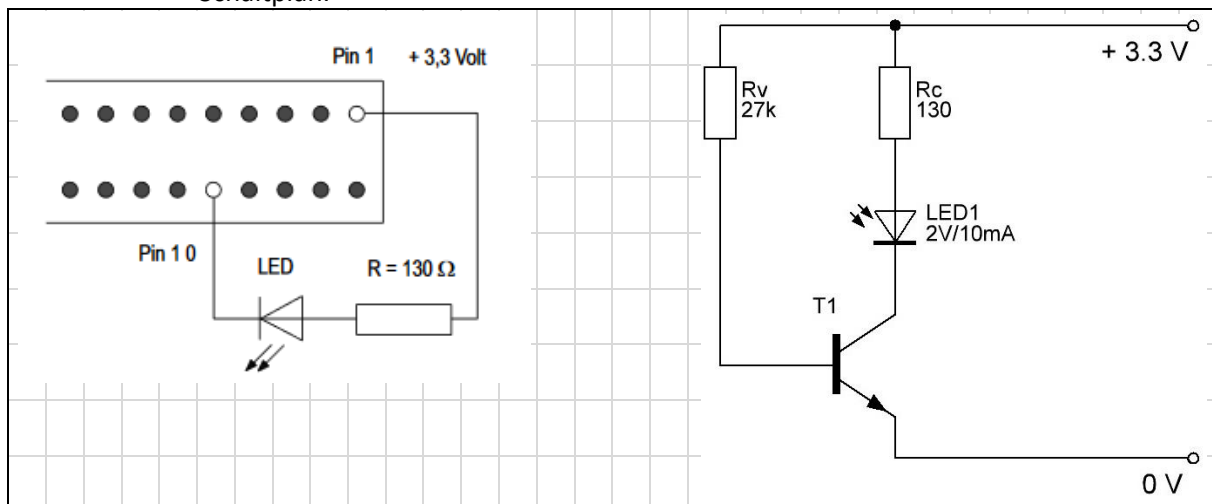
- Öffnen Sie die Programmieroberfläche und erstellen Sie einen Ordner mit dem Namen Ihrer Klasse (*mkdir Klasse*).
- Testen Sie, ob der Ordner angelegt wurde (*ls*).
- Öffnen Sie den Ordner (*cd Klasse*) und anschließend den Editor mit dem Namen Ihres Programms (*nano Ohmsches.py*)
- Lassen Sie dem Benutzer einen Spannungswert in Volt eingeben.
 - i. Weisen Sie der Variable U diesen Wert in Form eines float types zu.
- Lassen Sie dem Benutzer einen Stromwert in Ampere eingeben.
 - i. Weisen Sie der Variable I diesen Wert in Form eines float types zu.
- Berechnen Sie den Wirkwiderstand R
- Geben Sie das Ergebnis in folgender Form aus:
 - i. Der von Ihnen benötigte Widerstand R beträgt den Wert, Wert in Ω .
- Notieren Sie sich das funktionsfähige Programm im folgenden Feld (mit Kommentaren)

```

V = input ("Geben Sie einen Spannungswert ein: ")
A = input ("Geben Sie einen Stromwert ein: ")
U = float (V)
I = float (A)
R = U/I
print "Der benoetigte Widerstandswert betraegt: ", R , "Ohm"
```

4. Planen Sie eine Schaltung mit der Sie eine LED (rot) mit Hilfe des RPi ein- bzw. ausschalten können. Verwenden Sie dafür den Ausgabepin 10.

- Nennen Sie die benötigten Bauteile der Schaltung und zeichnen Sie im Anschluss den Schaltplan.



- Die LED soll mit einer Spannung von 2 V und einen Strom von 10 mA betrieben werden. Legen Sie den Wert des benötigten Vorwiderstandes R unter Verwendung des zuvor erstellten Python – Skriptes fest. Nutzen Sie dabei Ihr Wissen über den Spannungswert des Pin 10 GPIO!

Der benötigte Vorwiderstand R beträgt 130 Ω

- Bauen Sie die Schaltung auf der Steckplatine auf (beachten Sie dabei die richtige Polung der LED (*Anode (+) etwas länger als Kathode (-)*), programmieren Sie den RPi (orientieren Sie sich dabei an Aufgabe 2 des Arbeitsblattes) und notieren sich das korrekt entstandene Skript.

Achtung: Stellen Sie keine Verbindung (Kurzschluss) mit den 5V Pins her!!!

```
import RPi.GPIO as GPIO      # RPi.GPIO wird importiert
GPIO.setmode(GPIO.BOARD)    # Modus Pin-Nummerierung GPIO ein
GPIO.setup(10, GPIO.OUT)     # Pin 10 Ausgabekanal
GPIO.output(10, GPIO.HIGH)   # Am Pin 10 liegt eine Spannung von 3,3V gegen Masse an
GPIO.output(10, GPIO.LOW)   # Am Pin 10 liegt eine Spannung von 0V gegen Masse an
```

- Testen Sie Ihr Programm und präsentieren es Ihrem Vorgesetzten. Bereiten Sie sich dabei auf folgende Fragen vor:

i. Aus welchem Grund ist die LED im Zustand LOW in Betrieb?
Spannung entsteht durch Potentialunterschied, 3,3V am

Ausgang und 0V (LOW) entspricht 3,3 V Potentialunterschied.



ii. Warum muss auf die richtige Polung der LED geachtet werden?
Wenn die Polung nicht korrekt ist, wird die LED in

Sperrichtung betrieben.

Ordnen Sie diese Unterlagen unter „Informationssammlung Kundenauftrag Wildkamera_3“ ein.

Sehr geehrte Schülerinnen und Schüler,



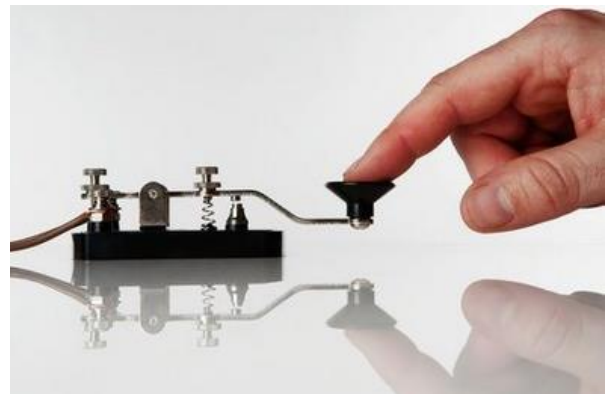
Sie haben in den letzten Wochen große Erfahrungen bezüglich Hard/- Software des RPi errungen. Nachdem es Ihnen sowohl gelungen ist ein Skript zu verfassen, mit dessen Hilfe der Widerstand berechnet werden kann, als auch einen Verbraucher mittels Softwarebefehlen ein- bzw. auszuschalten, ist es an der Zeit die Funktionen zu automatisieren. Bearbeiten Sie die folgenden Aufgaben Ihres Vorgesetzten schrittweise ab. Ziel ist es, eine LED in einem Morsecode blinken zu lassen. Senden Sie Ihrem Vorgesetzten die entstandenen Skripte per E-Mail zu.

Mit freundlichen Grüßen

Elektro GmbH



Abbildung 9: <https://www.conrad.de>



<https://www.telespiegel.de>



1. Orientieren Sie sich an der E-Mail Ihres Vorgesetzten und notieren sich die geforderten Informationen stichpunktartig.

2. Ihre erste Aufgabe wird es sein einen „Pseudocode“ zu erstellen. Dabei sollen Sie in eigenen Worten aufschreiben, welche Programmabläufe Schritt für Schritt durchgeführt werden müssen, um eine LED mit einer halben Sekunde Pause blinken zu lassen.
- Erstellen Sie den Pseudocode.

Eine LED soll für 0.5 Sekunden mit Spannung versorgt werden, anschließend für 0.5 Sekunden spannungsfrei sein. Diese Prozedur soll mindestens 5-mal durchlaufen werden.

- Nachdem Sie den Pseudocode erstellt haben, legen Sie die benötigten PINS am GPIO fest. Geben Sie zudem an, ob die PINS als Eingänge oder Ausgänge deklariert sein müssen. (Die PINBELEGUNG des RPi ist unter dem Befehl `gpio readall` zu finden)

Ich verwende für die benötigte LED den PIN 10 (weil es ein GPIO PIN ist) und einen GND (0V). Der PIN 10 ist als Ausgang zu deklarieren.

- Wandeln Sie den von Ihnen erstellten Pseudocode in ein funktionsfähiges Python-Skript um. Bauen Sie die Schaltung auf und testen diese auf Funktionalität. Hilfestellung: Für die Zeitfunktion müssen sie im Header die Bibliothek `time` importieren (`import time`), die Zeitfunktion an sich wird mit der Syntax (`time.sleep()`) abgerufen. Um die gesamte Funktion mehrmals ablaufen zu lassen, benötigen wir eine for-Schleife (`for x in range():`).

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(10, GPIO.OUT)
GPIO.output(10, GPIO.LOW)
for x in range(4):
    GPIO.output(10, GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(10, GPIO.LOW)
    time.sleep(0.5)
GPIO.cleanup()
```

Achtung, das Einrücken der Befehle ist sehr wichtig!!!

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(10, GPIO.OUT)
GPIO.output(10, GPIO.LOW)

dauer=0.5

for x in range(10):
    GPIO.output(10, GPIO.HIGH)
    time.sleep(dauer)
    GPIO.output(10, GPIO.LOW)
    time.sleep(dauer)

GPIO.cleanup()
```


- Nachdem Sie Ihr Programm getestet haben, erklären Sie in eigenen Worten, welche Funktion die for-Schleife in Ihrem Programm erfüllt.

Die for-Schleife wiederholt alle Befehle, die eingerückt sind, so oft ich es in der Klammer angegeben habe. Hier z.B. 3-mal (for x in range (3):)

3. Im Anschluss an diese Aufgabe, sollen Sie die LED mit einem Morsecode beschalten.
 - Recherchieren Sie im Internet über den Morsecode für S O S und geben diesen an.

3 mal kurz, 3 mal lang, 3 mal kurz

- Erstellen Sie den dafür benötigten Pseudocode.

Die LED soll für 0.3 Sekunden mit Spannung versorgt werden, anschließend für 0.3 Sekunden spannungsfrei sein. Diese Prozedur muss dreimal durchlaufen werden. Anschließend 0.4 Sekunden Pause.

Die LED soll für 0.5 Sekunden mit Spannung versorgt werden, anschließend für 0.5 Sekunden spannungsfrei sein. Diese Prozedur muss dreimal durchlaufen werden. Anschließend 0.4 Sekunden Pause.

Die LED soll für 0.3 Sekunden mit Spannung versorgt werden, anschließend für 0.3 Sekunden spannungsfrei sein. Diese Prozedur muss dreimal durchlaufen werden. Anschließend 1.0 Sekunden Pause.

Das gesamte Programm soll 4-mal durchlaufen werden.



Abbildung 10:<https://schaechter.asmblog.org>

- Bauen Sie die Schaltung auf der Steckplatine auf (beachten Sie dabei die richtige Polung der LED (Anode (+) etwas länger als Kathode (-)), programmieren Sie den RPi (orientieren Sie sich dabei an dem Pseudocode) und notieren sich das entstandene Skript.

Achtung: Stellen Sie keine Verbindung (Kurzschluss) mit den 5V Pins her!!!

<pre>import RPi.GPIO as GPIO import time GPIO.setmode(GPIO.BOARD) GPIO.setup(10, GPIO.OUT) GPIO.output(10, GPIO.LOW) for x in range(4): for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(0.3) GPIO.output(10, GPIO.LOW) time.sleep(0.3) for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(0.5) GPIO.output(10, GPIO.LOW) time.sleep(0.5) for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(0.3) GPIO.output(10, GPIO.LOW) time.sleep(0.3) time.sleep(0.5) GPIO.cleanup()</pre>	<pre>import RPi.GPIO as GPIO import time GPIO.setmode(GPIO.BOARD) GPIO.setup(10, GPIO.OUT) GPIO.output(10, GPIO.LOW) k = 0.3 #kurze Pause l = 0.5 #lange Pause b = 0.8 #Zeit zwischen Buchstaben w = 1.5 #Zeit zwischen Wort for x in range(5): for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(k) GPIO.output(10, GPIO.LOW) time.sleep(k) time.sleep(b) for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(l) GPIO.output(10, GPIO.LOW) time.sleep(l) time.sleep(b) for x in range(3): GPIO.output(10, GPIO.HIGH) time.sleep(k) GPIO.output(10, GPIO.LOW) time.sleep(k) time.sleep(w) GPIO.cleanup() #Setzt die GPIO PINS in dem Ausgangszustand zurueck</pre>
--	--

- Testen Sie Ihr Programm und präsentieren es Ihrem Vorgesetzten. Verwenden Sie den nachfolgenden Platz für persönliche Notizen.



4. Schreiben Sie Ihrem Vorgesetzten die geforderte E-Mail.

Ordnen Sie diese Unterlagen unter „Zeitfunktion und for-Schleife Wildkamera“ ein.

Sehr geehrte angehende Facharbeiterinnen und Facharbeiter,



bei einem weiteren Kundengespräch wollte der Auftraggeber wissen, ob es denn die Möglichkeit gäbe, den Betriebszustand der Anlage in Form von Lichtsignalen anzeigen zu lassen. Eventuell möchte er die Wildkamera so programmiert haben, dass Sie nur zu bestimmten Zeiten im Betriebszustand ist. Um diese Frage zu beantworten, erstellen Sie bitte ein Testskript, das die Anforderungen des Kunden erfüllt. Hierfür bietet sich eine Ampelsteuerung an.

Befassen Sie sich mit der Funktion der Ampelschaltung, erstellen ein passendes Skript und simulieren die Schaltung am Steckbrett. Geben Sie im Anschluss Ihrem Vorgesetzten Bescheid, ob die Kundenanforderungen erfüllbar sind.



Mit freundlichen Grüßen

Elektro GmbH

Abbildung 11:[<https://www.nzg.de/baustellenampel.html>]

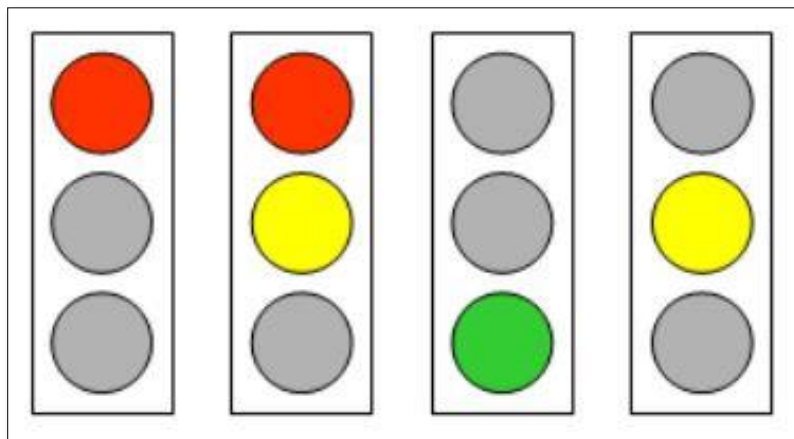


Abbildung 12: Ablauf Ampelfunktion [<http://schuljahr.inf-schule.de>]

Platz für Notizen:



1. Orientieren Sie sich an der Kundenanfrage und notieren sich die geforderten Informationen stichpunktartig.

2. Geben Sie stichpunktartig Ihr Vorgehen an, um diese Aufgabe erfolgreich zu absolvieren.

1. Pseudocode erstellen

2. Materialliste erstellen und benötigte PINS festlegen (EIN/AUSGÄNGE!!!)

3. Skript erstellen und simulieren, ggf. optimieren



- Erstellen Sie den benötigten Pseudocode für die Ampelsteuerung. Der Ablauf ist in Abbildung 2 dargestellt.

Es werden drei LEDs für die Simulation benötigt, rot und grün müssen mit einem

Vorwiderstand von 130Ω versorgt werden. Der allgemeine Ablauf:

Rote LED muss für 2 Sekunden auf HIGH gesetzt werden, anschließend wird

die rote und gelbe LED für 1 Sekunde auf HIGH gesetzt.

Im nächsten Schritt werden rot und gelb abgeschaltet und die grüne LED für

3 Sekunden angeschaltet.

Im Anschluss daran wird die grüne LED abgeschaltet und die gelbe LED für

1 Sekunde eingeschaltet, danach wieder abgeschaltet.

Dieser Vorgang soll zum Testen 4-mal wiederholt werden.

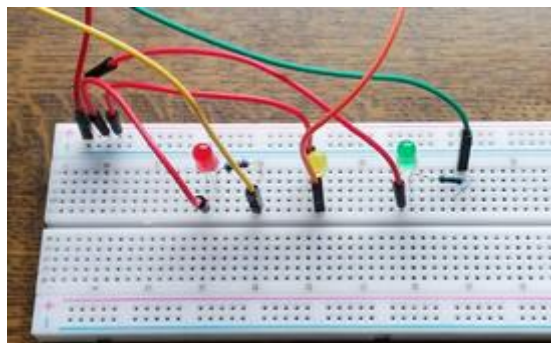
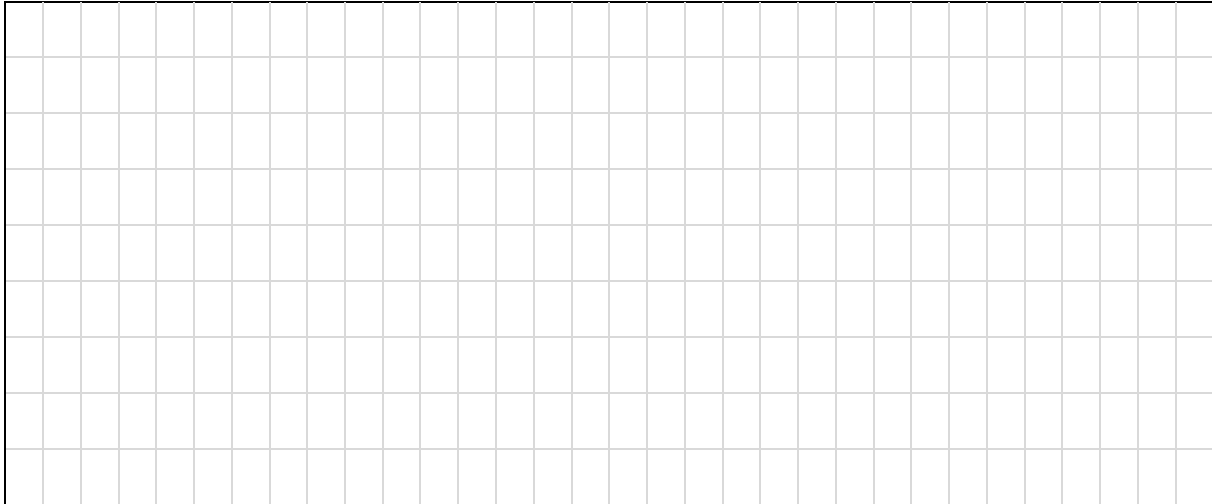


Abbildung 13: Beispielschaltung

- Erstellen Sie einen Schaltplan für die Ampelschaltung. Orientieren Sie sich dabei am Schaltplan aus dem Arbeitsblatt: 1.LED Schaltung.



- Wandeln Sie den von Ihnen erstellten Pseudocode in ein funktionsfähiges Pythonskript um. Bauen Sie die Schaltung auf und testen diese auf Funktionalität.

Hilfestellung: Für die Zeitfunktion müssen Sie im Header die Bibliothek Zeit importieren (*import time*), die Zeitfunktion an sich wird mit der syntax (*time.sleep()*) abgerufen. Um die gesamte Funktion mehrmals ablaufen zu lassen, benötigen wir eine for-Schleife (*for x in range()*).

```

import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(36, GPIO.OUT)
GPIO.setup(38, GPIO.OUT)
GPIO.setup(40, GPIO.OUT)

d=2

for x in range(3):
    GPIO.output(36, GPIO.HIGH)
    time.sleep(d)
    GPIO.output(36, GPIO.HIGH)
    GPIO.output(38, GPIO.HIGH)
    time.sleep(d)
    GPIO.output(36, GPIO.LOW)
    GPIO.output(38, GPIO.LOW)
    GPIO.output(40, GPIO.HIGH)
    time.sleep(d)
    GPIO.output(40, GPIO.LOW)
    GPIO.output(38, GPIO.HIGH)
    time.sleep(d)
    GPIO.output(38, GPIO.LOW)

GPIO.cleanup()

```




Sehr geehrte Schülerinnen und Schüler,

nachdem Sie sich mittlerweile mit diversen Schleifen, der richtigen Syntax und den Anfängen der Struktur des Betriebssystems Linux befasst haben. Ist es für unser Projekt an der Zeit sich neben der Programmierung der Ausgänge des GPIO's auch die Handhabung der GPIO – Pins für die Eingabe anzueignen. Programmieren Sie einen Bewegungsmelder als Eingang des RPi. Sobald der Bewegungsmelder ein Signal ausgibt, soll ein Bildschirm eingeschaltet werden. Dieses Einschalten wird bei Ihnen mit einer Bildschirmausgabe auf der Programmoberfläche Linux mit „Monitor EIN“ angezeigt, ansonsten soll „Monitor AUS“ aufleuchten. Bedenken Sie, dass die Bewegungsmelderabfrage einer Endlosschleife gleichgesetzt wird.

Mit freundlichen Grüßen

Elektro GmbH



Abbildung 14: smart mirrohr (Skript_Ingolstadt)



1. Orientieren Sie sich an der Kundenanfrage und notieren sich die geforderten Informationen stichpunktartig.

2. Geben Sie mögliche Lösungsansätze für das Kundenproblem an.

Die „Mirrohrs“ könnten mit Hilfe des RPi über einen Bewegungsmelder angesteuert werden. Für dem im Badezimmer befindlichen Smart Mirror könnte zusätzlich eine Ausschaltfunktion der Zeit, mittels eines akustischen Signals angewendet werden.

- Erkundigen Sie sich mit Hilfe des Internets über die while-Schleife für die Pythonanwendung. Gehen Sie dabei sowohl auf das Prinzip, sowie auf die Syntax ein.

Prinzip: Eine while-Schleife wiederholt sich so lange, bis entweder eine Forderung erfüllt ist oder eine programmierte Ausgangsmöglichkeit gezogen wird.

Syntax:

```
try:  
    while True:  
        ....  
        ....  
    Zum dauerhaften abfragen, zum Beispiel  
    Fehlerabfrage etc. ...
```

Beispiel: <https://studyflix.de/informatik/while-schleifen-in-c-687>

3. Erstellen Sie als erstes einen Pseudocode für die Ansteuerung des Verbrauchers mit einem Bewegungsmelder.

Der Bewegungsmelder muss als Eingang in den RPi gehen, der Mirror als Ausgang. Der BWM muss zu jeder Zeit abgefragt werden. Gibt er ein Signal, muss für eine Zeit von 20 Sekunden der Mirror auf Ein geschaltet werden. Im Anschluss wieder auf Aus. Das soll sich in einer Endlosschleife wiederholen.

4. Schreiben Sie das zugehörige Pythonskript. Zur Simulation soll sowohl eine Bildschirmausgabe mit „Monitor EIN“ bzw. „Monitor AUS“, sowie eine visuelle Anzeige in Form einer LED ausgegeben werden.
- Für den Anschluss des Bewegungsmelders verwenden Sie die Informationen des folgenden Bildes.

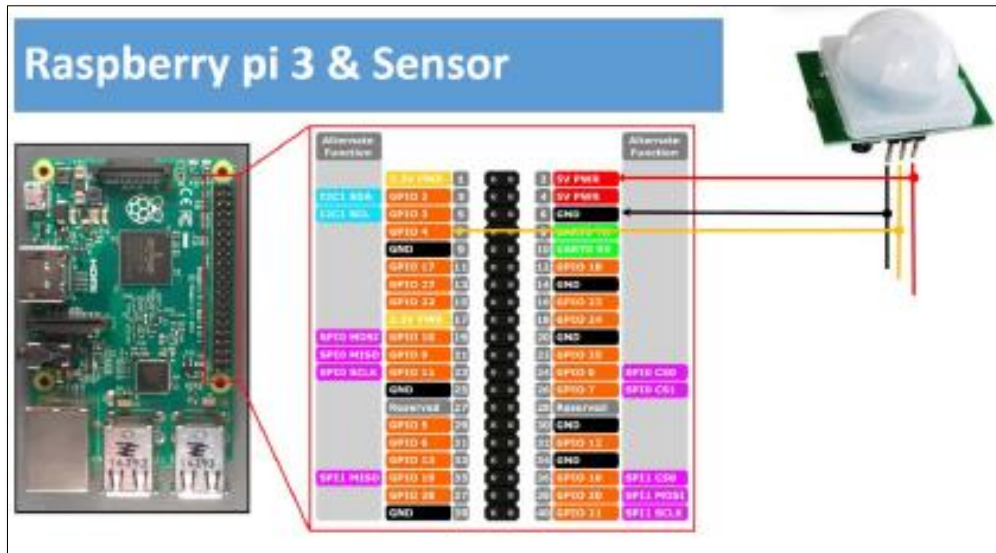


Abbildung 2: smart mirror (Skript_Ingolstadt)

Legen Sie zusätzlich noch den PIN für die Visualisierung fest. Beachten Sie dabei, dass der PIN vom Typ GPIO sein muss! Achten Sie auch darauf, ob Sie mit BCM oder BOARD programmieren, hier sind die Bezeichnungen der PINS unterschiedlich! Mit dem Befehl `gpio readall` können Sie sich die Belegungen jederzeit ansehen.

Weitere Hilfestellung:

GPIO-Einbindung in Python

- Paket importieren: `import RPi.GPIO as GPIO`
- GPIO definieren: z.B. `PIN = 18`
- Setup-Methode festlegen: `GPIO.setmode(GPIO.BCM)` oder `(GPIO.BOARD)`
- Als input definieren: `GPIO.setup(PIN, GPIO.IN)`
- Abfrage des Sensors mit: `GPIO.input(PIN)`
- Nicht vergessen: Aufräumen, wenn das Script beendet ist: `GPIO.cleanup()`

Nützliche Befehle:

<code>mkdir [Parameter] (Neues Verzeichnis)</code>	Erstellt einen neuen Ordner
<code>cp [Parameter] [Ausgangsdatei] [Ziel]</code>	Kopiert die Ausgangsdatei ins angegebene Ziel
nano (Texteditor zum Programmieren)	Aufruf mit <code>>> nano datei</code> Falls die Datei noch nicht existiert, wird sie neu angelegt Speichern mit Strg + o Beenden mit Strg + x
<code>cd Ordner</code>	Öffnet den gewünschten Ordner (wie Doppelklick auf Ordner bei Windows)
<code>ls</code>	Zeigt Dateien, die in dem Ordner enthalten sind (Mit <code>„ls -a“</code> werden alle Dateien angezeigt)

Skript:

```

import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(7, GPIO.IN)
GPIO.setup(16, GPIO.OUT)
import time
z=0
try:
    while True:
        if(GPIO.input(7)):
            GPIO.output(16, GPIO.HIGH)
            print ("Ein")
            time.sleep(2)
        else:
            z = z+1
            if(z==10):
                GPIO.output(16, GPIO.LOW)
                time.sleep(0.5)
                z = 0
                print ("Aus")
except KeyboardInterrupt:
    print ("Exit")

```

5. Nachdem Sie den Bewegungsmelder erfolgreich programmiert haben, schreiben Sie Ihrem Vorgesetzten eine E-Mail und senden Ihm im Anhang das fertige Skript zu.



Ordnen Sie diese Unterlagen unter „Zeitfunktion und while-Schleife Wildkamera_1“ ein.



Sehr geehrte angehende IT - Fachkräfte,

der Auftrag ist jetzt freigegeben worden. Orientieren Sie sich an dem zugrundeliegenden Kundenauftrag (abgeheftet in: „**Orientierung/- Informationen zum Kundenauftrag**“). Verwenden Sie außerdem die von Ihnen erarbeiteten Materialien, um den Auftrag zu bearbeiten. Mit Ihrem erlangten Fachwissen sollte der Auftrag in ca. 120 min. abgeschlossen sein. Erstellen Sie im Anschluss eine kleine Beschreibung für den Kunden, damit keine Bedienfehler entstehen. Diese Übergeben Sie ihm bei der Präsentation der Anlage. Nach der Übergabe erfolgt die Abrechnung. Erstellen Sie für dem Kunden eine Rechnung mit Material und Arbeitskosten. Vergessen Sie nicht die aktuelle Mehrwertsteuer sowie ggf. gewährtes Skonto. Für die Anbindung der Kamera steht Ihnen die beigelegte Bedienungsanleitung zur Verfügung.

Mit freundlichen Grüßen

Elektro GmbH



Abbildung 15: Wildkamera ohne Gehäuse [www.rasppishop.de]

1. Orientieren Sie sich an der Kundenanfrage und notieren sich die geforderten Informationen stichpunktartig.

Der Kunde möchte eine „Wildkamera“ für den Parkplatz seines Firmengeländes installiert

bekommen. Diese Anlage muss von unserem Unternehmen ausgewählt und installiert werden.

Wir benötigen dazu einen Raspberry Pi, einen Bewegungsmelder, eine Wildkamera und ggf.

ein externes Speichermedium. Zudem sollten passende Gehäuse erworben werden.

Die Kamera soll bei Betätigung des Bewegungsmelders ein Foto von dem zu überwachenden

Bereich schießen und dieses im Anschluss speichern. Zudem sollte das Programm leicht

abänderbar sein, dass ggf. die Kamera kurze Videos abspeichert. Wünschenswert wäre, dass

das System per Tastenbetätigung in Betrieb und außer Betrieb gesetzt werden kann.

2. Erstellen Sie den dazugehörigen Pseudocode:

Der Bewegungsmelder muss ständig auf Bewegung abgefragt werden. Sobald eine Bewegung stattfindet, muss die Kamera aktiviert werden. Der Speicherort der Kamera muss festgelegt und vom Anwender abgefragt werden können.

Zudem werden zwei Taster benötigt, um die Anlage ein- bzw. auszuschalten.

3. Geben Sie Ihr funktionsfähiges Skript an.

Skript:

```
import RPi.GPIO as GPIO

from picamera import PiCamera
camera = PiCamera()

GPIO.setmode(GPIO.BOARD)
GPIO.setup(7, GPIO.IN)
GPIO.setup(16, GPIO.OUT)

import time
z=0
b=0
j= '.jpg'

try:
    while True:
        if(GPIO.input(7)):
            GPIO.output(16, GPIO.HIGH)
            print("Ein")
            b = b+1
            t= str(b) + j
            camera.start_preview()
            camera.capture("/home/pi/Bilder/"+t)
            camera.stop_preview()
            time.sleep(2)

        else:
            z = z+1
            if(z==10):
                GPIO.output(16, GPIO.LOW)
                time.sleep(0.5)
                z=0
                print("AUS")
except KeyboardInterrupt:
    print("Exit")

GPIO.cleanup()
```

4. Erstellen Sie bitte eine Kurzanleitung, die Sie dem Kunden bei der Übergabe reichen können.



5. Erstellen Sie nun zum Projektabschluss eine Rechnung. Halten Sie sich dabei an die gegebenen Normen und verwenden realistische Preise. Bitte erstellen Sie diesen Geschäftsbrief mit MS – Word oder einer Freeware – Version wie z.B. Libre – Office oder Open – Office.
Drucken Sie die fertige Rechnung aus und übergeben Sie diese dem Kunden bei der Übergabe.



Sehr geehrte angehende IT - Fachkräfte,

ich darf Ihnen zum erfolgreichen Projektabschluss gratulieren! Der Kunde ist mit Ihrer Arbeit sehr zufrieden und wird unsere Firma für Folgeaufträge als erste Anlaufstation wählen.

Da wir ein sehr zielstrebiges Unternehmen sind, möchten wir Sie bitten, sich in Arbeitsgruppen aufzuteilen. Sie sollen in den Gruppen versuchen Ihr Vorgehen bei der Problemlösung zu optimieren. Zudem sollten Sie diskutieren, ob die aktuelle Anlage noch optimiert werden kann. Falls ja, wie?

Mit freundlichen Grüßen

Elektro GmbH

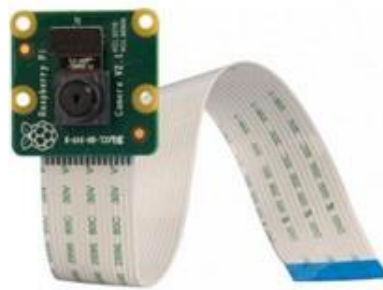


Abbildung 16: Wildkamera ohne Gehäuse [www.rasppishop.de]

1. Halten Sie auf den kommenden Seiten Ihre Gedanken fest.

2. Optimieren Sie Ihr Skript nach Ihren eigenen Beispielen. Sie können auch zwei Taster einbauen, mit denen die Anlage ein- bzw. ausgeschaltet werden kann. *(Verwenden Sie hierfür unter anderem eine if – else Anweisung.)*
3. Skript:

```

import RPi.GPIO as GPIO

from picamera import PiCamera
camera = PiCamera()

GPIO.setmode(GPIO.BOARD)
GPIO.setup(7, GPIO.IN)
GPIO.setup(16, GPIO.OUT)

import time
z=0
b=0
j= '.h264'

try:
    while True:
        if(GPIO.input(7)):
            GPIO.output(16, GPIO.HIGH)
            print("Ein")
            b = b+1
            t= str(b) + j
            camera.start_preview()
            camera.resolution = (854, 480)
            camera.framerate = 30
            camera.start_recording("/home/pi/Bilder/"+t)
            camera.wait_recording(20)
            camera.stop_recording()
            camera.stop_preview()
            time.sleep(2)

        else:
            z = z+1
            if(z==10):
                GPIO.output(16, GPIO.LOW)
                time.sleep(0.5)
                z=0
                print("AUS")
except KeyboardInterrupt:
    print("Exit")

GPIO.cleanup()

```

Projekt Abgeschlossen

