Onderzoek van Onderwijs (10 EC)

EDUCATIE EN COMMUNICATIE IN DE BÈTAWETENSCHAPPEN – NATUURKUNDE

Arduino lesmodule voor het Natuurkunde curriculum

AUTEUR

S.J. Lentelink

S0169064

ELAN, VAKGROEP DOCENTONTWIKKELING

COMMISSIE

Dr. ir. H.J. Pol | Universiteit Twente

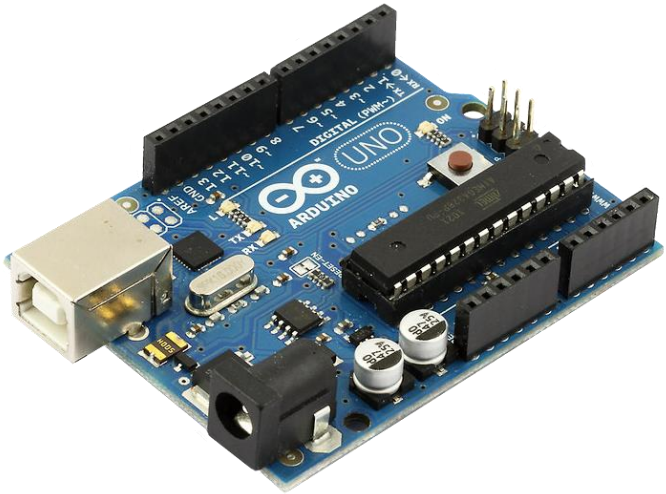
MSc. A.N. van Rossum | Universiteit Twente

29 augustus 2019

UNIVERSITY OF TWENTE.

“Ik vond het leuk dat je echt direct zag wat je had gemaakt
en je kunt veel dingen uitproberen met Arduino.”

(Leerling vwo-leerjaar 3, 2019)



SAMENVATTING

Het is van maatschappelijk belang om leerlingen in het voortgezet onderwijs voor te bereiden op de snel veranderende samenleving waarin digitalisering en technologische ontwikkelingen een belangrijke rol spelen. In het havo/vwo-curriculum van Natuurkunde kunnen deze zogenaamde 21e-eeuwse kennis en vaardigheden samen met diverse leerstofdomeinen worden betrokken middels Arduino; een elektronisch prototype-platform waarmee gebruikers interactieve elektronische objecten kunnen maken die reageren op hun omgeving. In dit onderzoek wordt geprobeerd antwoord te geven op de vraag: hoe ziet een mogelijke Arduino lesmodule eruit voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21e-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering?

INHOUDSOPGAVE

1 INLEIDING 6

2 THEORETISCH KADER 7

2.1 21e -eeuwse vaardigheden 7

2.1.1 Probleemoplossen 8

2.1.2 Computational thinking 8

2.1.3 ICT-basisvaardigheden 9

2.1.4 Samenwerken 9

2.2 Domein Elektriciteit 11

2.3 Domein Technische automatisering 12

2.4 Arduino 13

2.5 Ontwerpeisen lesmodule 14

2.5.1 Leertheorie 14

2.5.2 Projectmatige lessenserie 15

2.5.3 Effectief leren 15

3 VOORONDERZOEK – INTERVIEW DOCENT NATUURKUNDE 16

4 ONDERZOEKSVRAGEN 17

4.1 Onderzoeksvraag 17

4.2 Doelen 17

4.3 Programma van eisen 17

4.3.1 Praktijk 17

4.3.2 Theorie 17

5 METHODE 18

5.1 Context 18

5.2 Procedure 18

5.3 Respondenten 20

5.4 Instrumenten 20

5.5 Analyse 20

6 ONTWERP – PROTOTYPE 21

7 RESULTATEN 24

7.1 Praktijktest 24

7.1.1 (Pre- en post) zelfkennistest 24

7.1.2 Observaties 25

7.1.3 Reflectievragen 28

7.2 Peerfeedback docenten en TOA Natuurkunde 29

7.2.1 TOA Natuurkunde 29

7.2.2 Docent Natuurkunde met Arduino ervaring 29

7.2.3 Docent Natuurkunde 29

8 EINDPRODUCT 30

9 DISCUSSIE & CONCLUSIE 31

9.1 Discussie 31

9.1.1 Methode 31

9.1.2 Zelfkennistest 32

9.1.3 Observaties 32

9.1.4 Reflectievragen 33

9.1.5 Peerfeedback 33

9.2 Conclusie 34

10 REFERENTIES 35

BIJLAGEN 38

10.1 Informatiebrief aan ouders van leerlingen 39

10.2 Vooronderzoek – Interview docent Natuurkunde 40

10.3 Praktijktest – Arduino met bijbehorende componenten 41

10.4 Praktijktest – Zelfkennistest 42

10.5 Praktijktest – Reflectievragen 44

10.6 Arduino lesmodule – Leerling werkboek 46

10.7 Arduino lesmodule – Docent handleiding 80

1 INLEIDING

Het voortgezet onderwijs in Nederland is voortdurend onderhevig aan verandering, toegespitst op de ontwikkelingen in de samenleving. Een belangrijke vraag op dit moment is welke kennis en vaardigheden van belang zijn om leerlingen voor te bereiden op de snel veranderende maatschappij waarin digitalisering en technologische ontwikkelingen een belangrijke rol spelen (Onderwijsraad, 2017; Platform Onderwijs2032, 2016; SLO, 2014). Om deze vraag te beantwoorden heeft SLO (2019b), maar ook Next Generation Science Standards (2017) en Platform Onderwijs2032 (2016), geprobeerd om deze zogenaamde 21e-eeuwse of vakoverstijgende kennis en vaardigheden vorm te geven.

Het maatschappelijk belang om leerlingen voor te bereiden op de toekomst door het verwerven en versterken van vakoverstijgende 21e-eeuwse kennis en vaardigheden (SLO, 2014) wordt versterkt door het belang om leerlingen te enthousiasmeren voor de maakbaarheid van technologie om ons heen, en ze te leren hier zelfstandig mee om te gaan (Duifhuis & van Asseldonk, 2018; Platform Onderwijs2032, 2016). Een manier om dit te bewerkstelligen is door middel van Arduino (Scheltenaar, Van der Poel, & Bekker, 2015); een opensource elektronisch prototype-platform waarmee gebruikers interactieve elektronische objecten kunnen maken die reageren op hun omgeving door middel van digitale en analoge inputsignalen.

In het havo/vwo-curriculum voor Natuurkunde kan Arduino naast een bijdrage aan de 21e-eeuwse kennis en vaardigheden ook een belangrijke ondersteunende rol spelen in diverse domeinen in de onder- en bovenbouw. Hierbij kan worden gedacht aan: Ontwerpen; Modelvorming; Technisch-instrumentele vaardigheden; Elektriciteit; Technische automatisering; en Fysische informatica (College voor Toetsen en Examens, 2017a, 2017b; Duifhuis & van Asseldonk, 2018; Mooldijk & Pols, 2017; SLO, 2015). Aangezien Arduino echter wel kennis vereist van elektronica en programmeren gaat de voorkeur uit naar de domeinen Elektriciteit en Technische automatisering (Mooldijk & Pols, 2017). Voor deze domeinen is de inzet van Arduino met name interessant voor havo/vwo-leerjaar 3. Voor de kennismaking met Arduino wordt namelijk veelal gebruik gemaakt van relatief eenvoudige elektrische schakelingen en componenten welke veel overlap hebben met de leerstof in dit leerjaar. Ook wordt Technische automatisering vaak al gedeeltelijk in dat leerjaar behandeld. Daarnaast is er in de onderbouw meer vrijheid om zoiets als Arduino in te kunnen zetten. Tot slot kan Arduino leerlingen enthousiasmeren om ook in de bovenbouw voor Natuurkunde te kiezen, of ze anderzijds toch nog bepaalde kennis en vaardigheden meegeven.

Arduino biedt mogelijkheden voor integratie in lesmodules, practica, projectweken, Natuur Leven Technologie, en profielwerkstukken. Voor dit onderzoek is, in overleg met de betrokken school, gekozen om een Arduino lesmodule te ontwikkelen voor Natuurkunde havo/vwo-leerjaar 3. De lesmodule is vormgegeven in een basis voor Arduino met hierin de domeinen Elektriciteit en Technische automatisering verwerkt. Dit onderzoek is een 'ontwerp-onderzoek', wat inhoudt dat er doelgericht en systematisch een nieuwe werkwijze wordt ontworpen, ontwikkeld, en geëvalueerd om de praktijk te verbeteren (Van der Donk & Van Lanen, 2016).

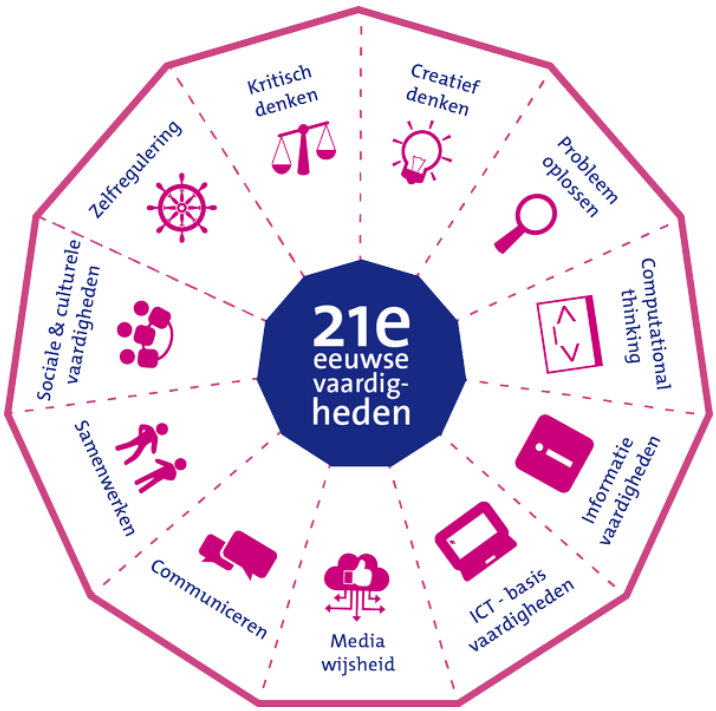
De onderzoeksvraag is:
Hoe ziet een mogelijke Arduino lesmodule eruit voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21e-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering?

2 THEORETISCH KADER

Voor het doelgericht en systematisch ontwerpen, ontwikkelen, en evalueren van een Arduino lesmodule zal in dit theoretisch kader onderzoek worden gedaan naar 21e-eeuwse vaardigheden, het domein Elektriciteit, het domein Technische automatisering, Arduino, en de ontwerpeisen voor een lesmodule.

2.1 21E -EEUWSE VAARDIGHEDEN

Er is momenteel veel discussie over de kennis en vaardigheden die leerlingen zouden moeten verwerven en versterken met het oog op toekomstige ontwikkelingen in de samenleving. In dit hoofdstuk zal daarom worden gekeken naar een concrete invulling van deze zogenaamde 21e-eeuwse of vakoverstijgende vaardigheden. Zo heeft Stichting Leerplanontwikkeling (SLO, 2019b) hiervoor 11 vaardigheden gepresenteerd, zie figuur 1, welke deels ook door Next Generation Science Standards (2017) en Platform Onderwijs2032 (2016) worden benoemd. Deze vaardigheden moeten volgens SLO (2014) in de toekomst een grotere rol gaan spelen in de beoogde en uitgevoerde lesmethodes. De onderdelen ICT-basiskennis, informatievaardigheid, mediawijsheid, en computational thinking worden daarbij door Platform Onderwijs2032 (2016) aangeschreven als kern van toekomstgericht onderwijs. Deze onderdelen, samengevat als digitale geletterdheid, zijn namelijk gericht op het werken en leren in de digitale wereld met nieuwe technologieën (Platform Onderwijs2032, 2016).



Figuur 1 – 21e-eeuwse vaardigheden (SLO, 2019a)

In het kader van dit onderzoek, waarbij Arduino als ondersteunend middel wordt gebruikt, is gekozen voor de volgende vaardigheden (SLO, 2014):

- Probleemoplossen
- Computational thinking
- ICT-basisvaardigheden
- Samenwerken

2.1.1 Probleemoplossen

Probleemoplossen is het vermogen om een probleem te herkennen en planmatig op te lossen (Leraar24, 2019), waarbij de leeractiviteit bestaat uit een authentiek probleem met een voor de leerling nog onbekende uitkomst (SLO, 2019b). Voor een Arduino lesmodule kan hier worden gedacht in de context van het realiseren/oplossen van een praktijksituatie met behulp van Arduino en eventuele kennis van Elektriciteit en Technische automatisering.

Deelvaardigheden in het leerplankader voor probleemoplossen zijn:

Tabel 1 – Deelvaardigheden in het leerplankader voor probleemoplossen (SLO, 2019b)

Probleemoplossen	De leerling:
Probleemoplosprocessen	Kent procedurele en conceptuele kennis over probleemoplosprocessen. Kent verschillende probleemoplosstrategieën en bijbehorende stappen daarin.
Problemen signaleren	Signaleert, onderkent een probleem of vraag.
Problemen definiëren	Verkent, duidt, verheldert en definieert problemen en vraagstukken. Herkent verschillen in mate van gestructureerd zijn, domein specifiekheid/abstractie en complexiteit.
Problemen analyseren	Onderzoekt en ontleedt een probleem grondig.
Oplossingen zoeken, bedenken, selecteren	Zoekt, bedenkt, selecteert, met gebruik van verschillende technieken (deductie, inductie, 'best fit', patroonherkenning).
Een oplossing realiseren	Realiseert een oplossing (volgens plan).
Evalueren van de oplossing	Toetst een oplossing aan het oorspronkelijke probleem. Reflecteert op het doorlopen proces. (Zo nodig stappen herhalen (terug naar analyse, deelproblemen, of naar oplossingen, het plan bijstellen)).

2.1.2 Computational thinking

Computational thinking is het vermogen om problemen zo te formuleren dat ze op te lossen zijn met behulp van ICT (Leraar24, 2019). Leerlingen kunnen dit leren door technologie te gebruiken, zoals door kennismaking met programmeren of werken met robotica (Platform Onderwijs2032, 2016). Daarbij wekt het interesse en biedt het een basis voor eventuele latere specialisatie (Platform Onderwijs2032, 2016).

Deelvaardigheden in het leerplankader voor computational thinking zijn:

Tabel 2 – Deelvaardigheden in het leerplankader voor computational thinking (SLO, 2019b)

Computational thinking	De leerling:
Gegevens verzamelen	Kan relevante informatie verzamelen via verschillende bronnen zoals artikelen, interviews, enquêtes of literatuurstudie.
Gegevens analyseren	Kan logisch ordenen en analyseren van gegevens, begrijpen van gegevens, vinden van patronen, trekken van conclusies, evalueren van grafieken of toepassen van relevante statistische methodes.
Gegevens visualiseren	Kan gegevens weergeven door middel van bijvoorbeeld modellen van de werkelijkheid, of door grafieken, tabellen, woorden en plaatjes en het selecteren van de effectiefste representatie.

Probleem decompositie	Kan een taak opdelen in kleinere, overzichtelijke taken, zoals bijvoorbeeld het opdelen van een lange lijst met opdrachten in subcategorieën van typen opdrachten en het plannen van een project door middel van deelprojecten.
Abstractie	Kan de complexiteit van een probleem reduceren om te komen tot de kern van het probleem.
Algoritmes en procedures	Kan een serie geordende stappen gebruiken om een probleem op te lossen of een bepaald doel te bereiken.
Automatisering	Kan een computer zichzelf herhalende of eentonige taken laten uitvoeren totdat een oplossing is bereikt.
Simulatie en modellering	Kan een model of een proces weergeven, of het uitvoeren van een experiment op basis van dat model of proces.
Parallellisatie	Kan zorgen voor gelijktijdige uitvoering van taken om een gezamenlijk doel te bereiken.

2.1.3 ICT-basisvaardigheden

ICT-basisvaardigheden zijn de kennis en vaardigheden omtrent computer hardware en software (Leraar24, 2019). Hierbij gaat het om de bediening, mogelijkheden, en beperkingen van digitale technologie (Platform Onderwijs2032, 2016; SLO, 2019b). In dit onderzoek zijn met name basisbegrip ICT en infrastructuur technologie van toepassing.

Deelvaardigheden in het leerplankader voor ICT-basisvaardigheden zijn:

Tabel 3 – Deelvaardigheden in het leerplankader voor ICT-basisvaardigheden (SLO, 2019b)

ICT-basisvaardigheden	De leerling:
Basisbegrip ICT	Kan functies van computers en computernetwerken benoemen.
Infrastructuur technologie	Kan hardware benoemen, aansluiten en bedienen. Kan verschillende apparaten en programma's bedienen. Kan informatie opslaan en toegankelijk maken.
Standaardtoepassingen	Kan omgaan met standaard kantoortoepassingen en andere softwareprogramma's voor onder meer internetgebruik, beeldbewerking, samenwerking en betalingsverkeer.
Veiligheid	Is op de hoogte van en kan omgaan met beveiligings- en privacyaspecten in het kader van persoonlijke en financiële gegevens.

2.1.4 Samenwerken

Samenwerken omvat het gezamenlijk behalen van een doel en daarbij anderen kunnen aanvullen en ondersteunen (Leraar24, 2019). De leeractiviteit moet gericht zijn op taken die leerlingen uitnodigen om samen tot de beste resultaten te komen (SLO, 2019b). Hierbij moet er aandacht zijn voor reflectie en zelfregulerend vermogen (SLO, 2019b). Leerlingen moeten actief deelnemen en oog hebben voor zowel hun eigen rol en bijdrage als voor anderen en het groepsproces (SLO, 2019b). Het is belangrijk dat leerlingen het belang inzien van samenwerken en dit ook ervaren (SLO, 2019b). Hierbij moet samenwerken gekoppeld worden aan concrete leerinhouden (Platform Onderwijs2032, 2016).

Deelvaardigheden in het leerplankader voor samenwerken zijn:

Tabel 4 – Deelvaardigheden in het leerplankader voor samenwerken (SLO, 2019b)

Samenwerken	De leerling:
Communiceren	Communiceert effectief (specifiek: luisteren, observeren, duidelijk spreken, reageren, overleggen). Vraagt en geeft hulp en feedback, staat open voor hulp en feedback.
Taakuitvoering en (team)rollen	(H)erkent verschillende rollen bij zichzelf en anderen en benut deze in de samenwerking. Houdt zich aan afspraken. Stelt zich verantwoordelijk op ten aanzien van zowel de eigen bijdrage, het groepsproces als het te bereiken doel.
Organiseren	Kan een proces inrichten om een gezamenlijk doel te bereiken. Onderhandelt en maakt afspraken met anderen in een team.
Werken in een team en houding	Ondersteunt en begeleidt anderen. Functioneert in heterogene groepen. Heeft een positieve en open houding ten aanzien van andere ideeën. Heeft respect voor verschillen. Wil samen met anderen tot een goed resultaat komen. Heeft vertrouwen in de eigen capaciteiten.

Samenwerkend leren is een belangrijke instructiestrategie omdat het leidt tot effectief leren, het de onderwijsleersituatie structureert, en het beantwoordt aan een maatschappelijke vraag (Ebbens & Ettekoven, 2015). De volgende vijf sleutelbegrippen zijn van belang voor succesvol samenwerken (Ebbens & Ettekoven, 2015):

- Positieve wederzijdse afhankelijkheid; het elkaar nodig hebben om een opdracht uit te voeren. Een docent heeft hierop invloed door te zorgen voor bronafhankelijkheid (groepsleden hebben hun eigen kennis), taakafhankelijkheid (groepsleden vervullen verschillende taken), of rolafhankelijkheid (groepsleden vervullen verschillende rollen).
- Individuele aanspreekbaarheid; het aanspreekbaar zijn op zowel de eigen bijdrage als het groepsresultaat. Een docent heeft hierop invloed in de vormgeving van de opdracht door voorafgaand aan te kondigen dat iedereen individueel getoetst wordt op het einde, leerlingen met kleurstiften hun eigen bijdrage zichtbaar moeten maken, leerlingen een zichtbare groepsrol te geven, of door aan te geven dat de klassikale nabespreking met een willekeurig groepslid zal plaatsvinden.
- Directe interactie; het elkaar aanmoedigen, ondersteunen en op directe wijze communiceren. Een docent heeft hierop invloed door leerlingen tegenover elkaar te laten zitten.
- Sociale vaardigheden; het bezitten van de vaardigheden om samen te kunnen leren.
- Reflectie op inhoud en leerproces; het samen doorspreken van de bijdrage van elk groepslid. Een docent heeft hierop invloed door leerlingen regelmatig hun eigen functioneren te laten bespreken en hierover suggesties ter verbetering te laten doen.

2.2 DOMEIN ELEKTRICITEIT

Voor het domein Elektriciteit beschrijft het SLO (2015) voor de havo/vwo onderbouw het domein 'Electriciteit en magnetisme' met de bijbehorende leerdoelen zoals gedefinieerd in tabel 5.

Tabel 5 – SLO Leerdoelenkaart; gedifferentieerde beheersingsniveaus voor de onderbouw van voortgezet onderwijs (SLO, 2015)

Elektriciteit en magnetisme	
Tussendoel havo	Tussendoel vwo
Je legt het opwekken van elektrische energie in een elektriciteitscentrale uit en je beschrijft het transporteren ervan.	Je legt het opwekken van elektrische energie in een elektriciteitscentrale uit en je beschrijft het transporteren ervan.
Je beschrijft hoe zekering en aardlekschakelaar toegepast wordt om een elektrische huisinstallatie te beveiligen, je legt de werking hier van uit, je herkent het risico van elektrische stroom door het lichaam en je beschrijft hoe dat vermeden wordt.	Je legt uit hoe zekering en aardlekschakelaar toegepast wordt om een elektrische huisinstallatie te beveiligen, je legt de werking hier van uit, je beschrijft het risico van elektrische stroom door het lichaam en je beschrijft hoe dat vermeden wordt.
Je legt eigenschappen van elektrische geleiders en isolatoren uit en je legt een aantal van hun toepassingen uit.	Je legt eigenschappen van elektrische geleiders en isolatoren uit en je legt een aantal van hun toepassingen uit.
Je beschrijft en gebruikt spanningsbronnen als energiebron, je herkent lading als drager/transporteur van energie en je beschrijft een aantal toepassingen.	Je beschrijft en gebruikt spanningsbronnen als energiebron, je beschrijft lading als drager/transporteur van energie en je beschrijft een aantal toepassingen.
Je legt het verband tussen stroomsterkte en spanning uit met een model, bijvoorbeeld met het waterkraanmodel.	Je verklaart het verband tussen stroomsterkte en spanning met een model, bijvoorbeeld met het waterkraanmodel.
Je beschrijft het verschil een serie- en parallelschakeling in een stroomkring en het effect dat serie- en parallelschakelingen hebben op de stroomsterkte en spanning.	Je legt uit het verschil een serie- en parallelschakeling in een stroomkring en het effect dat gecombineerde serie- en parallelschakelingen hebben op de stroomsterkte en spanning.
Je legt uit hoe de grootte van de spanning en stroomsterkte in een schakeling gemeten wordt met een V- en een A-meter en je legt met een model uit waarom deze zo geschakeld moeten zijn.	Je legt uit hoe de grootte van de spanning en stroomsterkte in een schakeling gemeten wordt met een V- en een A-meter en je verklaart waarom deze zo geschakeld moeten zijn.
Je beschrijft het energiegebruik uit als product van vermogen en tijd (E=P*t) en het vermogen als product van spanning en stroomsterkte (P=U*I).	Je legt het energiegebruik uit als product van vermogen en tijd (E=P*t) en het vermogen als product van spanning en stroomsterkte (P=U*I).

Mooldijk et al. (2017) beschrijven in hun opbouw voor het leerstofdomein 'Elektriciteit en magnetisme' het voor dit onderzoek toepasbare deelthema 'Elektrische schakelingen'. Dit deelthema delen zij op in: stroomkring; spanning, stroomsterkte en weerstand; en serie- en parallelschakeling. Aanbevolen wordt om te beginnen met de stroomkring, door het maken van eenvoudige en meer complexe schakelingen bestaande uit een spannings-bron, schakelaars, snoeren, en lampjes. Te denken valt aan het laten branden van een lampje, respectievelijk het laten branden van twee lampjes met eigen schakelaars. Belangrijk is om tegelijkertijd de bijbehorende schakelschema's te presenteren zodat leerlingen met verschillende representaties leren werken. Spanning, stroomsterkte, en weerstand worden opgebouwd door leerlingen eerst geleiders en isolatoren te laten onderscheiden via een spanningsbron en lampje zodat het begrip weerstand naar voren komt. Door leerlingen vervolgens spanning en stroomsterkte te laten meten in serie- en parallelschakelingen, waarbij ook de weerstand wordt veranderd, wordt geprobeerd leerlingen gevoel te laten krijgen voor spanning als eigenschap van de spanningsbron en stroomsterkte als gevolg van de spanning over een weerstand in de stroomkring. Tot slot kunnen leerlingen serie- en parallelschakelingen onderzoeken omtrent het effect op spanning en stroom; spanningsverdeling in een serieschakeling, en stroomverdeling in een parallelschakeling.

Op de betrokken school wordt voor havo/vwo-leerjaar 3 gebruik gemaakt van de lesmethode Nova (Malmberg, 2014a, 2014b). Hieruit wordt het hoofdstuk 'Schakelingen | Automatisch regelen' behandeld waarin het domein Elektriciteit naar voren komt. Dit hoofdstuk bevat globaal de volgende onderwerpen:

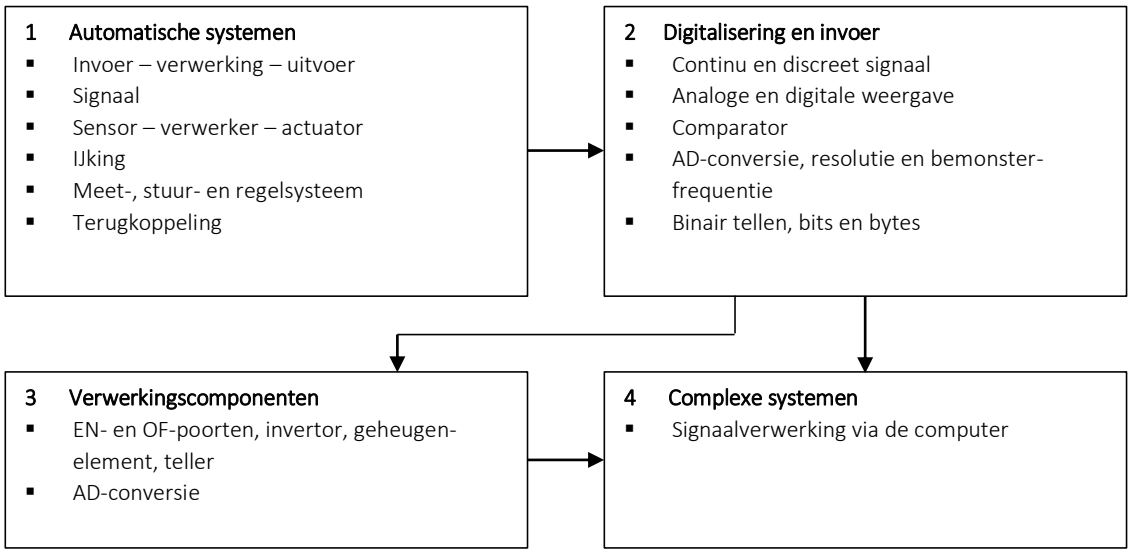
1. Lading en spanning
2. Weerstand
3. Weerstanden schakelen
4. Schakelingen in de automatisering

Voor dit onderzoek is gekozen om al deze onderwerpen, met Arduino als hulpmiddel, zoveel mogelijk terug te laten komen, waarbij de theorie en opgaven uit Nova (Malmberg, 2014a, 2014b) worden gebruikt als leidraad.

2.3 DOMEIN TECHNISCHE AUTOMATISERING

Het domein Technische automatisering wordt, ondanks dat het is ingedeeld in het eindexamenprogramma bij bovenbouw havo, in veel gevallen al (gedeeltelijk) in havo/vwo-leerjaar 3 behandeld. Volgens SLO (2017) moeten leerlingen in dit domein uiteindelijk meet-, stuur- en regelsystemen kunnen construeren en de functie en werking van de componenten kunnen beschrijven. Vakbegrippen die hierbij komen kijken zijn: meet-, stuur- en regelsystemen; sensoren en de werking daarvan; verwerkers en actuatoren; terugkoppeling; analoge en digitale signalen; en binair tellen (SLO, 2017).

Een mogelijke opzet voor mogelijke deelthema's, en de behandelroute, voor zowel onder- als bovenbouw is:



Figuur 2 – Opbouw van het leerstofdomein technische automatisering (Mooldijk & Pols, 2017)

Op de betrokken school wordt, zoals in *§2.2 Domein Elektriciteit* is benoemd, voor havo/vwo-leerjaar 3 gebruik gemaakt van de lesmethode Nova (Malmberg, 2014a, 2014b). Hieruit wordt het hoofdstuk 'Schakelingen | Automatisch regelen' behandeld waarin het domein Technische automatisering naar voren komt.

Voor dit onderzoek is gekozen om al deze onderwerpen, met Arduino als hulpmiddel, zoveel mogelijk terug te laten komen, waarbij de theorie en opgaven uit Nova (Malmberg, 2014a, 2014b) worden gebruikt als leidraad.

2.4 ARDUINO

Arduino is een opensource elektronisch prototype-platform waarmee gebruikers interactieve elektronische objecten kunnen maken die reageren op hun omgeving door middel van digitale en analoge inputsignalen. Arduino kan ondersteunen in het verwerven en versterken van vakoverstijgende 21e-eeuwse kennis en vaardigheden (Scheltenaar et al., 2015; SLO, 2014), en kan leerlingen enthousiasmeren voor de maakbaarheid van technologie om ons heen en ze leren hier zelfstandig mee om te gaan (Duifhuis & van Asseldonk, 2018). Arduino biedt leerlingen ook de mogelijkheid om kennis te maken met programmeren en het gebruiken van microcontrollers, sensoren, en actuatoren. Ook staan in de Natuurkunde huidige practica soms wat ver van de hedendaagse techniek af (zoals het systeembord), of is het erg vereenvoudigd en gesloten (zoals CoachLab). Arduino zou hiervoor een mogelijk alternatief kunnen zijn (Nelk & Pols, 2016).

De kracht van Arduino ligt in het eenvoudig kunnen opzetten en realiseren van projecten (Mooldijk & Pols, 2017), waarmee leerlingen snel tot een volledig prototype kunnen komen (Pols, 2017; in Mooldijk & Pols, 2017). Daarbij zijn Arduino en bijbehorende sensoren relatief goedkoop, bieden veel mogelijkheden (Mooldijk & Pols, 2017), en is er online veel ondersteuning en toepassingen te vinden. Ook speelt Arduino een steeds grotere rol in toekomstige ontwikkelingen zoals het Internet of Things (IoT), STEAM (Arduino, 2019), en initiatieven zoals Makersworld (Wearable Art Awards Holland, 2017).

Volgens Mooldijk & Pols (2017) biedt Arduino veel overlap met onderdelen uit het leerstofdomein elektriciteit, zoals het uitlezen van spanningsdelers, het maken van serieschakelingen, en het gebruik van diverse elektrische componenten. Ook biedt Arduino volgens hen mogelijkheden voor het ontwerpen van practica waarbij de Arduino als regel- of meetinstrument kan dienen. Echter geven zij aan dat Arduino nog wel in de kinderschoenen binnen het voortgezet onderwijs staat. Zij stellen dat het voor Arduino wenselijk is om de domeinen Elektriciteit en Technische automatisering te behandelen omdat Arduino kennis van elektronica en programmeren vereist. Hierbij leent het zich ook erg goed voor differentiatie, aangezien sommige leerlingen dit snel in de vingers hebben.

De opbouw van opdrachten om leerlingen met Arduino te leren werken lijkt eenduidig (Mooldijk & Pols, 2017). Leerlingen beginnen veelal met het laten knipperen van een LED (Herger & Bodarky, 2015; Kampen, 2019; Mooldijk & Pols, 2017; F. Pols, 2017a; Van Dijk, 2019). Hierna vervolgen zij met het aansturen van een knipperlicht, wat hen inzicht geeft in de programmeertaal (Kampen, 2019; Mooldijk & Pols, 2017; F. Pols, 2017a; Van Dijk, 2019). Een verdere uitbreiding kan worden gemaakt met een drukknop en meerdere LEDs, zoals een reactietijdspel of verkeerslicht met voetgangersoversteekplaats (Kampen, 2019; Mooldijk & Pols, 2017; F. Pols, 2017a; Van Dijk, 2019). Door deze drukknop te relateren aan het uitlezen van een sensor is de stap klein naar het uitlezen van sensoren en het aansturen van actuatoren (Mooldijk & Pols, 2017). Ook kan de LED zo geprogrammeerd worden dat deze automatisch dimt en weer feller gaat branden, waarmee een overstap naar motor met instelbare draaisnelheid kan worden gemaakt (F. Pols, 2017a; Van Dijk, 2019). Andere voorbeelden zijn een inbraakalarm dat afgaat bij detectie van beweging en een weerstation gekoppeld aan een thermostaat (Mooldijk & Pols, 2017). Aandacht is echter wel vereist bij grotere projecten, aangezien leerlingen vaak moeite hebben om een project op te delen in kleine stukken (Mooldijk & Pols, 2017). Aanvullend biedt deze manier van opbouwen een stapsgewijze mogelijkheid voor het aanleren van de benodigde programmeervaardigheden.

Ondanks dat Arduino binnen het voortgezet onderwijs nog in de kinderschoenen staat, zijn hiervoor al wel veel initiatieven te vinden. Zo worden er beginnerslessen aangeboden in de vorm van handleidingen (Killian, 2015; F. Pols, 2019; Wetzels, Kooyman, Moonen, & den Teuling, 2018), PowerPoint (Dingemans & Philippi, 2015), en video (Kampen, 2019; Van Dijk, 2019). Dit laatste biedt ook mogelijkheden voor werkvormen zoals Flipping the classroom. Ook worden er workshops aangeboden voor leerlingen via pre-universiteit-programma's (U-Talent, 2018; Universiteit Twente, 2018) en voor docenten (Boswell-Bèta, 2019). Er worden ook hulpprogramma's ontwikkeld om de programmeerdrempel te verlagen, zoals Scratch (MIT Media Lab, 2019), waarmee leerlingen op een interactieve manier met simpele bouwstenen de Arduino kunnen programmeren. Ook wordt Arduino al ingezet in de Natuurkunde lespraktijk, zoals bij: bestudering van een vrije val (Moya, 2018); thermodynamica, optica en golven (Petry, Pacheco, Lohmann, Correa, & Moura, 2016); technisch Ontwerpen (F. Pols, 2017b); en robotica (Duifhuis & van Asseldonk, 2018). Daarnaast wordt er al onderzoek gedaan naar het opdoen van 21e-eeuwse vaardigheden met Arduino in het voortgezet onderwijs. Zo onderzoeken Scheltenaar et al. (2015) het gebruik van Design Based Learning in combinatie met Arduino. Hierbij werken leerlingen op creatieve spelende

wijze via een ontwerpopdracht aan 21e-eeuwse vaardigheden via Arduino. Interessante aandachtspunten die zij vonden zijn het belang van: het relateren van opdrachten en activiteiten aan 'echte' problemen; het koppelen van leerdoelen uit het curriculum aan 21e-eeuwse vaardigheden; het duidelijk geven van instructies in verband met de gelimiteerde beschikbaarheid van de docent in een klas; en het lastig kunnen meten van de mate van ontwikkeling van 21e-eeuwse vaardigheden in de praktijk.

Tot slot zijn er sinds het ontstaan van Arduino vele zeer gedetailleerde boeken (Blum, 2013; Boxall, 2013; Margolis, 2012; Nussey, 2013) te vinden over werken met Arduino die de (leerling en) docent kan ondersteunen.

In dit onderzoek is gekozen om bovenstaande aspecten zoveel mogelijk mee te nemen in het ontwerp van de lesmodule aangezien er niet specifiek voor de domeinen 21e-eeuwse kennis en vaardigheden, Elektriciteit, en Technische automatisering een Arduino initiatief is gevonden, laat staan voor havo/vwo-leerjaar 3. Aanvullend zal in dit ontwerp geen gebruik worden gemaakt van hulpprogramma's zoals Scratch. De reden hiervoor is dat voor de te behalen doeleinden relatief weinig geprogrammeerd zal hoeven worden. Hierdoor kunnen leerlingen toch kennis maken met de 'echte' Arduino, en wordt hier niet alsnog een zogenaamde 'black box' van gemaakt.

2.5 ONTWERPEISEN LESMODULE

Voor de ontwikkeling van de lesmodule wordt eerst gekeken naar leertheorieën, gevolgd door de gemaakte keuze richting een projectmatige lessenserie en aanvullende opzet voor effectief leren.

2.5.1 Leertheorie

De inzet van Arduino om 21e-eeuwse kennis en vaardigheden te verwerven en versterken, en om leerdoelen voor de domeinen Elektriciteit en Technische automatisering te behalen, zou kunnen worden vormgegeven in een alomvattende lesmodule. Echter is de vraag in hoeverre het samenvoegen van al deze doelen opweegt tegen het afzonderlijk behandelen ervan. Uitgaande van deze twee uitersten wordt daarom ondersteuning gezocht in de leertheorieën. Voor de opzet van een Arduino lesmodule zijn met name het cognitivisme en constructivisme erg toepasselijk omdat leerlingen naar verwachting in eerste instantie veel ondersteuning nodig zullen hebben, waarna ze richting een meer zelfstandige werkhouding kunnen gaan.

In het cognitivisme wordt uitgegaan van het voortbouwen op de voorkennis en het bevorderen van denken door het stellen van vragen (Van der Veen & Van der Wal, 2016). Dit is een aantrekkelijke methode voor het versterken van de domeinen Elektriciteit en Technische automatisering, welke immers voorafgaand aan deze lesmodule al grotendeels zijn behandeld in de les. Hierin past ook goed de guided discovery van Bruner – het zelf tot inzichten laten komen door sturende vragen te stellen (Bruner, 1966; in van der Veen & van der Wal, 2016). Anderzijds past guided discovery ook goed bij het overdragen van nieuwe kennis, met name voor het aanleren van de 21e-eeuwse kennis en vaardigheden. Aanvullend stellen Gagné, Briggs, & Wager (1992) als voorwaarde dat er verschillende niveaus van leren zijn die elk een ander soort instructie vereist. Hierbij schetsen ze op hiërarchische wijze een set van educatieve gebeurtenissen (zie *§2.5.3*) met de bijbehorende cognitieve processen. Deze gebeurtenissen lijken geschikt voor zowel de domeinen Elektriciteit en Technische automatisering alsook de 21e-eeuwse vaardigheden.

In het constructivisme wordt uitgegaan van ontdekkend leren via concrete toepassingscontexten (Van der Veen & Van der Wal, 2016). Hierbij wordt van leerlingen verwacht dat ze actief en zelfstandig werken en de aangeboden kennis op actieve wijze verwerken, voortbouwend op hun voorkennis (Van der Veen & Van der Wal, 2016). Tevens wordt leren gezien als een sociaal proces, waarbij kennis ontstaat en wordt gedeeld met anderen (sociaal constructivisme). Passende onderwijsvormen hiervoor zijn projectgestuurd leren of probleemgericht leren, wat goed past bij het ontwikkelen van 21e-eeuwse kennis en vaardigheden, maar eveneens geschikt kan zijn voor de domeinen Elektriciteit en Technische automatisering.

Zowel het cognitivisme als het constructivisme biedt mogelijkheden voor het behalen van de 21e-eeuwse kennis en vaardigheden alsook de domeinen Elektriciteit en Technische automatisering. Gebruikmakend van deze leertheorieën zou ieder onderdeel apart kunnen worden behandeld. Echter biedt de overlap in deze situatie ook goede mogelijkheden om de onderdelen te combineren in een alomvattende lesmodule, gebruikmakend van beide theorieën. Zo kan een projectgestuurde onderwijsvorm worden gebruikt (constructivisme) waarbij leerlingen zelf tot inzicht komen via sturende vragen (cognitivisme; Bruner) die gerealiseerd zijn via stapsgewijze

gebeurtenissen (cognitivisme; Gagné), waarbij Arduino een geschikt middel kan zijn om de onderdelen in één lesmodule te combineren.

Als vervolgstap is gekozen om verdieping te zoeken in het opzetten van een projectgestuurde onderwijsvorm, in dit geval een projectmatige lessenserie, en hoe hierbij de leerstof zo effectief mogelijk kan worden overgebracht.

2.5.2 Projectmatige lessenserie

De ontwikkeling van de lesmodule kan worden gerelateerd aan het opzetten van een projectmatige lessenserie. Hierbij werken leerlingen voornamelijk zelfstandig in kleine groepen terwijl de docent een begeleidende rol heeft (Kortland & Poorthuis, 2017). De nadruk ligt op de onderzoekende werkwijze en het gebruik van kennis en vaardigheden voor nieuwe situaties, met focus op zelfredzaamheid, denkactiviteit, en samenwerking bij leerlingen (Kortland & Poorthuis, 2017). Hierbij zijn de volgende leerprincipes van belang (Geerlings & Van der Veen, 1987; APA, 2015; in Kortland & Poorthuis, 2017):

- Biedt leerlingen oriëntatie op hetgeen wat geleerd moet worden.
- Sluit aan bij, en activeer, de voorkennis van leerlingen.
- Laat leerlingen actief bezig zijn met de leerstof.
- Gebruik reële probleemsituaties voor de kennisverwerving van leerlingen.
- Geef regelmatig feedback op de vorderingen van leerlingen.

Een mogelijke invulling van deze principes is als volgt: voor het bieden van oriëntatie op hetgeen geleerd moet worden kan aan leerlingen vooraf de leerdoelen worden meegedeeld. Voor het aansluiten bij en activeren van de voorkennis kunnen leerstof en opdrachten uit de domeinen Elektriciteit en Technische automatisering worden gebruikt. De Arduino kan daarbij een geschikt middel zijn om leerlingen hier actief mee bezig te laten zijn. Reële probleemsituaties zouden met de Arduino kunnen worden vormgegeven, aansluitend op de belevingswereld van de leerling. Tot slot zou de docent regelmatig feedback kunnen geven op de vorderingen van leerlingen.

2.5.3 Effectief leren

Voor de ontwikkeling van een lesmodule is het belangrijk dat de leerdoelen zo effectief mogelijk op de leerlingen worden overgebracht. Volgens Gagné et al. (1992) moeten hiervoor de volgende gebeurtenissen aanwezig zijn:

- Focus de aandacht.
- Informeer over de doelstellingen.
- Stimuleer de voorkennis.
- Presenteer stimulerend materiaal.
- Zorg voor begeleiding.
- Laat aantonen wat is geleerd.
- Geef feedback.
- Beoordeel de prestaties.
- Verbeter het behoud en overdracht.

Een mogelijke invulling van deze gebeurtenissen is als volgt: voor het focussen van de aandacht kan in de klas een Arduino project worden gedemonstreerd, of hier een video van laten zien. Informeren over de doelstellingen kan door vooraf de leerdoelen mee te delen. De voorkennis stimuleren zou kunnen worden gedaan door leerstof en opdrachten uit de domeinen Elektriciteit en Technische automatisering aan te bieden. Hierbij kan de Arduino als stimulerend materiaal worden gebruikt. Voor de begeleiding kan de docent worden ingezet, of zouden leerlingen een handleiding/werkboek kunnen gebruiken. Een handleiding/werkboek kan aanvullend geschikt zijn om efficiënt en zelfstandig te werken. Door passende opdrachten te gebruiken zou aan leerlingen kunnen worden getoond wat ze hebben geleerd, maar hierin zou ook de docent een belangrijke rol kunnen spelen. De docent kan tevens worden ingezet voor het geven van feedback en het beoordelen van de prestaties.
Om de docent te ondersteunen kan ervoor worden gekozen om de gebeurtenissen vooraf kenbaar te maken aan de docent. Anderzijds kunnen deze gebeurtenissen ook worden verwerkt in de opzet van de lesmodule.

3 VOORONDERZOEK – INTERVIEW DOCENT NATUURKUNDE

Als aanvullend vooronderzoek voor de ontwikkeling van de Arduino lesmodule is een semigestructureerd interview, zie bijlage 10.2, gehouden met een docent Natuurkunde met enkele jaren ervaring omtrent het inzetten van Arduino bij havo/vwo-leerjaar 3 in het domein Elektriciteit. De vragen in dit interview hebben betrekking op de onderwerpen 21e-eeuwse vaardigheden, Elektriciteit, Technische automatisering, en op hoe de betrokken docent zijn eigen Arduino lesmodule heeft opgezet.

De belangrijkste uitkomsten uit het interview zijn:

- Belangrijke 21e-eeuwse vaardigheden, zowel om te behandelen als passend voor Arduino, zijn programmeren en inzicht in schakelingen. De leerlingen hebben van schakelingen al beperkte kennis.
- Behandelde onderwerpen in het domein Elektriciteit die tevens passen bij Arduino zijn schakelingen.
- Behandelde onderwerpen in het domein Technische automatisering zijn er dit jaar niet.
- De huidige opzet van de Arduino lesmodule ziet er als volgt uit:
    - Er worden 4 lessen besteed aan Arduino.
    - Leerlingen worden ingedeeld in groepjes van 2 a 3 leerlingen.
    - Het te behandelen onderwerp is vooral programmeren.
    - Het leerdoel is om leerlingen te stimuleren eigen projecten te ontwikkelen, in algoritmes te denken, en out of the box te denken.
    - Er wordt een PowerPoint gebruikt met daarin alle stappen.
    - De opbouw van de lessenserie begint met een voorbeeldprogramma dat leerlingen moeten modificeren. Dit is het laten knipperen van de on-board LED op de Arduino. Het einddoel is een verkeerssituatie, met een schakelaar in de weg om verkeerslichten in beweging te zetten.
    - Leerlingen worden beoordeeld op hun proces.
- Er is voor Arduino gekozen omdat met Arduino eigen projecten kunnen worden ontwikkeld.
- Problemen uit het verleden waren veelal technische problemen, zoals het niet herkennen van de Arduino na aansluiten op het besturingssysteem Linux dat gebruikt werd op de beschikbare laptops op school.
- Een algemene tip zou zijn om leerlingen een ruimte te bieden waar ze hun Arduino projecten kunnen laten staan zodat zij er op eigen gelegenheid aan verder kunnen werken.

4 ONDERZOEKSVRAGEN

In dit hoofdstuk wordt de onderzoeksvraag, de doelen, en het programma van eisen besproken.

4.1 ONDERZOEKSVRAAG

De onderzoeksvraag is:
Hoe ziet een mogelijke Arduino lesmodule eruit voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21e-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering?

4.2 DOELEN

De doelen voor deze lesmodule zijn:

- Leerlingen voorbereiden op (toekomstige) ontwikkelingen in de samenleving door het verwerven en versterken van 21e-eeuwse kennis en vaardigheden:
  - Probleemoplossen: het vermogen om een probleem te herkennen en planmatig op te lossen.
  - Computational thinking: het vermogen om problemen zo te formuleren dat ze op te lossen zijn met behulp van ICT.
  - ICT-basisvaardigheden: kennis en vaardigheden omtrent computer hardware en software.
  - Samenwerken: het gezamenlijk behalen van een doel en daarbij anderen kunnen ondersteunen.
- Leerlingen verdieping en verbreding bieden op het onderwijs in de Natuurkunde domeinen Elektriciteit en Technische automatisering.

4.3 PROGRAMMA VAN EISEN

Voor een overzicht van de aspecten die in de lesmodule betrokken moeten worden is een verzameling van eisen gedefinieerd gebaseerd op het theoretisch kader, het vooronderzoek, en overleg met de betrokken school. Hierbij is vanuit praktisch oogpunt gekozen voor een lesmodule van 4 lesuren, passend in de beschikbare tijd voor havo/vwo-leerjaar 3 van de betrokken school (met het oog op de toekomst), en voor een werkboek en handleiding voor respectievelijk leerling en docent om op efficiënte en zelfstandige wijze de lesmodule te kunnen uitvoeren.

4.3.1 Praktijk

- Uitvoerbaar in 4 lesuren.
- Werkboek voor de leerling.
- Handleiding voor de docent.

4.3.2 Theorie

- De volgende hardware en software moeten inhoudelijk aan bod komen:
  - Arduino.
  - Componenten: breadboard; jumper; LED; en weerstand.
  - Arduino IDE (met focus op: sketch; void setup(); void loop(); verifieer; en upload).
- De volgende 21e-eeuwse vaardigheden moeten inhoudelijk aan bod komen:
  - Probleemoplossen.
  - Computational thinking.
  - ICT-basisvaardigheden.
  - Samenwerken.
- Het domein Elektriciteit moet inhoudelijk als volgt aan bod komen:
  - Rekenen met spanning, stroomsterkte, en weerstand met behulp van: wet van Ohm; soortelijke weerstand; serieschakelingen; en parallelschakelingen.
  - Tekenen van elektrische schakelingen, inclusief serie- en parallelschakelingen.
- Het domein Technische automatisering moet inhoudelijk als volgt aan bod komen:
  - Basisdelen van een geautomatiseerd systeem, inclusief componenten om die taken uit te voeren
  - Meet-, stuur-, en regelsysteem.

5 METHODE

In de methode wordt de context, procedure, respondenten, instrumenten, en analyse in dit onderzoek besproken.

5.1 CONTEXT

In dit onderzoek is een Arduino lesmodule ontwikkeld voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 met focus op 21e-eeuwse kennis en vaardigheden en de domeinen Elektriciteit en Technische automatisering. Dit onderzoek is gebaseerd op literatuuronderzoek, ondersteuning van ervaringsdeskundigen, en een evaluatie met leerlingen in de praktijk. De evaluatie is uitgevoerd in het Natuurkunde curriculum van vwo-leerjaar 3 op een middelbare school in Nederland. Hierin is het prototype van de ontwikkelde Arduino lesmodule uit dit onderzoek geëvalueerd, maar ook vergeleken met de al bestaande persoonlijke Arduino lesmodule van de betrokken docent met Arduino ervaring (in dit onderzoek *controle lesmodule* genoemd).

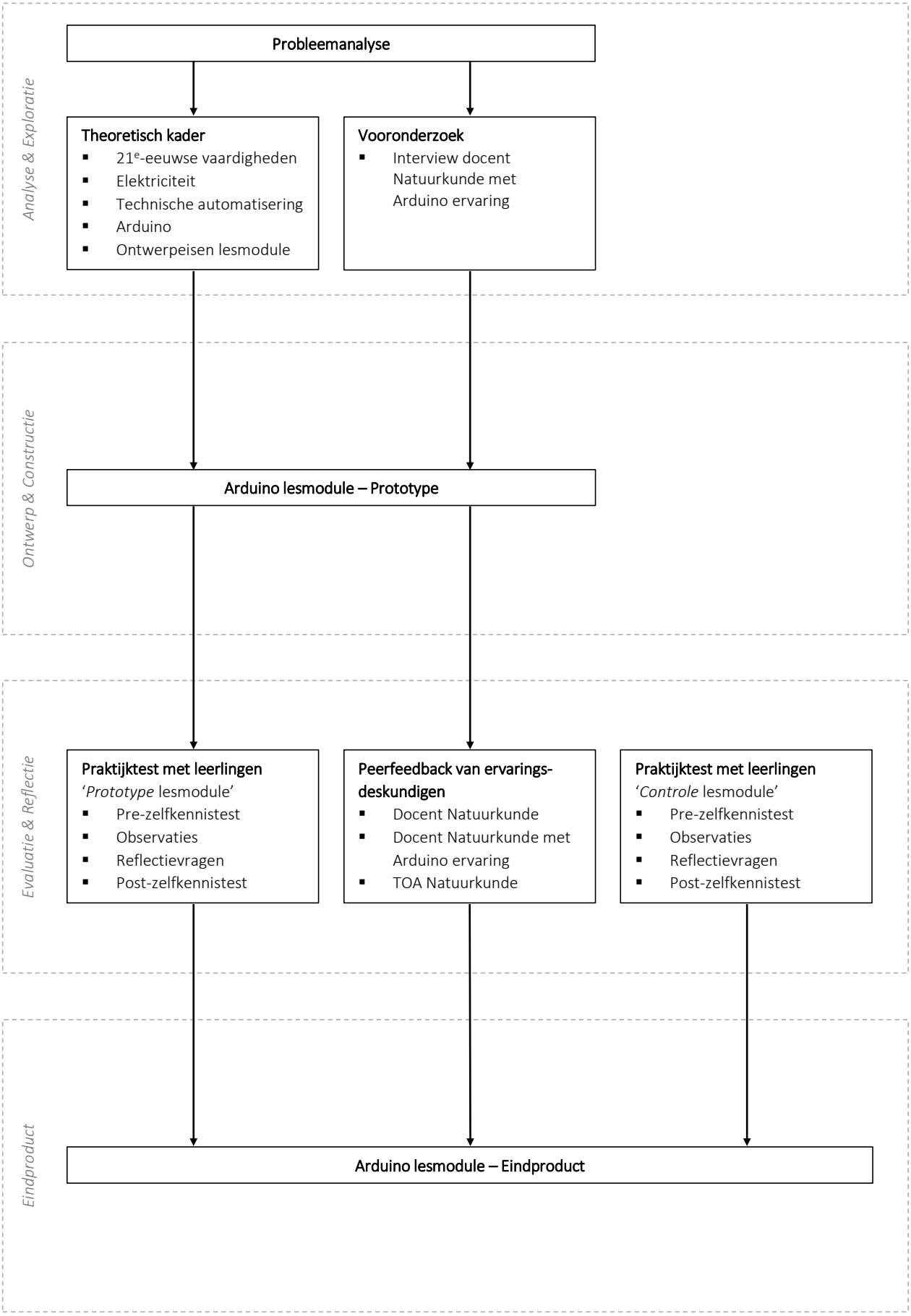
Let op: in dit onderzoek wordt over twee verschillende lesmodules gesproken:

- *Prototype lesmodule:* *het in dit onderzoek ontwikkelde Arduino lesmodule prototype, zie §6 Ontwerp (niet te verwarren met het eindproduct, zie §8 Eindproduct).*
- *Controle lesmodule:* *de al bestaande persoonlijke Arduino lesmodule van de betrokken docent Natuurkunde met enkele jaren ervaring omtrent het inzetten van Arduino in het domein Elektriciteit bij havo/vwo-leerjaar 3.*

5.2 PROCEDURE

De opzet van dit onderzoek is in grote lijnen weergegeven in figuur 3. Het onderzoek begon met een probleem-analyse, gevolgd door een literatuuronderzoek naar '21e-eeuwse vaardigheden', het 'domein Elektriciteit', het 'domein Technische automatisering', 'Arduino', en de 'ontwerpeisen voor een lesmodule'. Daarnaast is een semigestructureerd interview gehouden met een docent Natuurkunde met enkele jaren ervaring omtrent het inzetten van Arduino in het domein Elektriciteit bij havo/vwo-leerjaar 3. Dit interview is tezamen met het literatuuronderzoek gebruikt voor het *prototype*, ofwel een eerste volledige versie, van de Arduino lesmodule. Deze *prototype* lesmodule is vervolgens geëvalueerd in een praktijktest met 22 leerlingen uit een vwo-leerjaar 3 klas gedurende 2 Natuurkunde lessen van 50 minuten aan het einde van de behandelde module 'Schakelingen | Automatisch regelen' van de lesmethode Nova (Malmberg, 2014b). Hierbij is voorafgaand aan deze lesmodule bij de leerlingen een pre-zelfkennistest afgenomen, zijn de leerlingen tijdens de lesmodule geobserveerd, en hebben de leerlingen achteraf reflectievragen en een post-zelfkennistest ingevuld. De praktijktest is uitgevoerd door de onderzoeker onder begeleiding van de betrokken docent met Arduino ervaring. Ook is deze *prototype* lesmodule geëvalueerd met 2 Natuurkunde docenten (de docent met Arduino ervaring, en een bovenbouw docent die in het verleden les heeft gegeven aan havo/vwo-leerjaar 3), en met de TOA van de vaksectie Natuurkunde.
Daarnaast is de *prototype* lesmodule uit dit onderzoek vergeleken met de *controle* lesmodule van de betrokken docent Natuurkunde met Arduino ervaring. Deze *controle* lesmodule is door deze docent in een praktijktest uitgevoerd bij een parallel vwo-leerjaar 3 klas met 20 leerlingen gedurende 2 Natuurkunde lessen van 50 minuten. Hierbij zijn op dezelfde wijze als bij de *prototype* lesmodule dezelfde pre-zelfkennistest, reflectievragen, en post-zelfkennistest afgenomen. Gedurende beide lessen is er door de onderzoeker geobserveerd.
Het *eindproduct* in dit onderzoek is een verbetering van het *prototype* van de Arduino lesmodule op basis van de uitkomsten uit beide praktijktesten en de ontvangen peerfeedback van de docenten en TOA Natuurkunde.

Voor dit onderzoek is gekozen voor volledige openheid over het doel en de aard van het onderzoek; gegevens zijn anoniem verwerkt en opgeslagen, en de verwachting was dat respondenten weinig tot geen ongemak van het onderzoek ervaarden. Het onderzoek vond in goed overleg met de betrokken docent plaats, waarbij voor het anoniem verzamelen van leerlingdata voor toestemming gebruik is gemaakt van een zogenaamd 'passief informed consent'. Hiervoor hebben de ouders van de leerlingen een brief ontvangen, zie bijlage 10.1. Voor dit onderzoek is een ethiekaanvraag ingediend en goedgekeurd door de ethische commissie van de Universiteit Twente onder verzoeknummer 190860.



Figuur 3 – Onderzoeksopzet

5.3 RESPONDENTEN

In dit onderzoek lag de focus op het Natuurkunde curriculum, domein Elektriciteit en Technische automatisering, van havo/vwo-leerjaar 3. De betrokken respondenten in deze situatie waren:

- Docenten; die lesgeven/gaven aan havo/vwo-leerjaar 3 in het vak Natuurkunde.
- TOA; die werkzaam is in de vaksectie Natuurkunde.
- Leerlingen; uit vwo-leerjaar 3 die deze lesmodule als leerstof krijgen.

In dit onderzoek hebben in totaal 2 docenten, 1 TOA, en 42 leerlingen deelgenomen.

Er is in dit onderzoek door anoniem verzamelen van gegevens geen onderscheid gemaakt tussen respondenten.

5.4 INSTRUMENTEN

Voor het uitvoeren van de lesmodule ontvingen leerlingen een werkboek, vergelijkbaar met het eindproduct in bijlage 10.6, en hebben zij gewerkt met Arduino en bijbehorende componenten, zie bijlage 10.3.

In dit onderzoek is data verzameld via een interview, zelfkennistests, observaties, reflectievragen, en peerfeedback zoals expliciet beschreven in *§5.2 Procedure*. Het interview, zie bijlage 10.2, is gebaseerd op de betrokken onderwerpen 21^{e}-eeuwse vaardigheden, Elektriciteit, Technische automatisering, en op hoe de betrokken docent zijn eigen Arduino lesmodule heeft opgezet. De pre- en post-zelfkennistest, zie bijlage 10.4, zijn hetzelfde, en gebaseerd op de 21^{e}-eeuwse vaardigheden en leerstof die in de prototype Arduino lesmodule terugkomen. De observaties zijn gebaseerd op het met een kritische blik observeren van de leerlingen tijdens het werken met de lesmodule. De reflectievragen, zie bijlage 10.5, zijn opgesteld met het oog op Arduino, de leerstof Elektriciteit en Technische automatisering, en de 21^{e}-eeuwse vaardigheden, aangevuld met wat leerlingen goed en minder goed vonden aan de lesmodule. Tot slot is de peerfeedback vormgegeven door de betrokken Natuurkunde docent met Arduino ervaring, een bovenbouw Natuurkunde docent met havo/vwo-leerjaar 3 ervaring, en een TOA van de vaksectie Natuurkunde te vragen de lesmodule kritisch te beoordelen.

5.5 ANALYSE

In de analyse van de verzamelde data zijn het interview uit het vooronderzoek, de observaties, en de peerfeedback uitgeschreven in termen van 'belangrijkste punten', zoals door de onderzoeker geïnterpreteerd. De zelfkennistests en reflectievragen zijn onderworpen aan een statistische analyse, waarvoor door de beperkte steekproefgrootte en niet-normale verdeling voor vergelijkingen binnen de lesmodules de Wilcoxon Signed Ranks Toets is gebruikt, en voor vergelijkingen tussen de lesmodules de Mann-Whitney U toets. Hierbij werd een p-waarde <0,05 als significant beschouwd. De zogenaamde 'Tops' en 'Tips' uit de reflectievragen zijn gesorteerd en samengevat.

6 ONTWERP – PROTOTYPE

In dit hoofdstuk wordt het ontwerp van het prototype, ofwel de eerste volledige versie, van de Arduino lesmodule besproken. Dit prototype is gebaseerd op het theoretisch kader, vooronderzoek, en programma van eisen.

De lesmodule bestaat uit een werkboek voor de leerling en een handleiding voor de docent. Het werkboek is opgezet voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 voor het verwerven en versterken van 21e-eeuwse kennis en vaardigheden en het behalen van leerstof uit de domeinen Elektriciteit en Technische automatisering. De lesmodule is modulair vormgegeven en kan in 4 lessen van 50 minuten worden gegeven. Leerlingen werken hierbij zelfstandig doorlopend aan de lesmodule. De docent kan bij minder tijd, of gewenste doelen, onderdelen door leerlingen laten overslaan. Ter voorbereiding is een computer met Arduino IDE nodig, een Arduino set (bestaande uit Arduino, breadboard, jumpers, LEDs, weerstanden, lichtgevoelige weerstand, en drukknop), en dient het werkboek en de docenthandleiding te worden uitgeprint.

In het ontwerp van de lesmodule zijn de volgende 21e-eeuwse kennis en vaardigheden meegenomen: Probleemoplossen; Computational thinking; ICT-basisvaardig-heden; en Samenwerken. Probleemoplossen is geprobeerd te betrekken door het implementeren van miniprojecten aan het einde van de module. Hierin worden verkeerssituaties geschetst die leerlingen met Arduino moeten zien te realiseren met behulp van de opgedane kennis. Computational thinking komt voornamelijk terug in de lesmodule doordat leerlingen kennismaken met programmeren en door het gebruiken van algoritmes en procedures om opdrachten uit te voeren met de Arduino. De Arduino ondersteunt bij ICT-basisvaardigheden door leerlingen kennis op te laten doen en te werken met de hardware en software, en in te laten zien welke functies hiermee mogelijk zijn. Tot slot wordt Samenwerken gestimuleerd door leerlingen in groepsverband aan de lesmodule te laten werken. Hierdoor kunnen zij elkaar ondersteunen in zowel het eigen maken van de theorie alsook het maken van de opdrachten. Leerlingen worden gemotiveerd een bijdrage te leveren aan het groepsresultaat doordat de klassikale nabespreking via willekeurige groepsleden gebeurd. Ook moeten leerlingen tegenover elkaar plaatsnemen, in plaats van naast elkaar, en dienen leerlingen per les af te wisselen tussen het werken met de hardware en software. Dit houdt in dat ze alles wel samen moeten doen, maar per les afwisselen tussen het schrijven van de programma's op de laptop en het werken met de Arduino. Ook loopt de docent rond om kritisch te kijken naar de samenwerking en hierop te reflecteren.

Het werkboek voor de leerling is als volgt vormgegeven: het werkboek begint met een korte inleiding over de relevantie van de lesmodule, gevolgd door een opsomming van de leerdoelen. Daarna volgt algemene informatie over Arduino, gevolgd door het domein Technische automatisering dat hier goed op aansluit. De informatie en opdrachten over Technische automatisering bevatten leerstof uit de lesmethode Nova, maar ook opgestelde begrippen door SLO, zoals het benoemen van de basisonderdelen van een geautomatiseerd systeem, of het geven van voorbeelden van een meet- stuur- en regelsysteem. Hierbij wordt Arduino ingezet als toepassingscontext. Overigens is het niet per se noodzakelijk dat leerlingen de leerstof voorafgaand al behandeld hebben gekregen. Het domein Elektriciteit komt vervolgens naar voren door enerzijds het aanbieden van verdiepende leerstof, zoals de LED, maar anderzijds vooral door het aanbieden van de vele praktische opdrachten. Hiervoor is het wel van belang dat leerlingen de leerstof voorafgaand hebben gehad, aangezien de opdrachten met name versterkend werken. Hierbij is de leerstof uit de lesmethode Nova, maar ook de tussendoelen van SLO en opzet van Mooldijk et al., zoveel mogelijk verwerkt. Leerlingen starten qua leerstof met een eenvoudige stroomkring, waarbij ze via een opdracht over soortelijke weerstand ondervinden dat de LED in serie met een weerstand moet worden geschakeld. Hierbij moeten ze het schakelschema tekenen en met de wet van Ohm de benodigde weerstands-waarde berekenen om de LED op de juiste spanning te kunnen laten branden. Het uitproberen op de Arduino is hierbij de motiverende verbinding tussen theorie en praktijk. Ter afwisseling van de standaard lesmethode wordt vervolgd met de integratie van een sensor in de schakeling, de drukknop. Hierbij ligt de focus meer richting het programmeren, en is deze ook nodig voor de miniprojecten aan het einde van de lesmodule. Opvolgend gaan leerlingen bezig met weerstanden in serie- en paralellelschakeling, waarbij ze eerst het schakelschema moeten tekenen alvorens via uitproberen op de Arduino de vragen gaan beantwoorden. Daarna wordt kort de lichtgevoelige weerstand (LDR) behandeld, om inzicht te krijgen in dynamische weerstanden en de toepasbaarheid hiervan. Vervolgens gaan leerlingen kijken naar LEDs in serie- en parallelschakeling, waarbij ze eerst het schakelschema moeten tekenen alvorens via uitproberen op de Arduino de vragen gaan beantwoorden.

Hierna wordt weer meer aandacht besteed aan het programmeren en maken leerlingen een stoplicht door verschillende LEDs te combineren met de Arduino. Tot slot worden leerlingen uitgedaagd aan de hand van miniprojecten waarin zij een situatieschets krijgen van een voetgangersoversteekplaats die zij moeten realiseren. Hierbij kunnen zij gebruik maken van de opgedane kennis uit het werkboek.
Alle onderdelen in de lesmodule zijn zo opgezet dat kennis en vaardigheden op ordelijke en stapsgewijze manier worden verworven of versterkt, voortbouwend op de voorkennis. Daarbij zijn de opdrachten zo gemaakt dat zij leerlingen via sturende vragen zoveel mogelijk zelf tot inzichten laten komen.

De lesmodule voor de docent is voornamelijk ondersteunend en bevat informatie over de doelgroep, leerdoelen, tijdsduur, voorbereiding, opzet, en de uitwerkingen. Hierbij is voor de opzet gekozen om de gebeurtenissen uit *§2.5.3 Effectief leren* te gebruiken.

In tabel 6 op de volgende pagina staat een globaal overzicht van de inhoud van het werkboek voor de leerling. In figuur 4 worden enkele voorbeeldpagina's van het werkboek getoond.

Figuur 4 – Werkboek voor de leerling

Tabel 6 – Overzicht van de inhoud van het werkboek voor de leerling in het prototype van de Arduino lesmodule

- Inleiding
- Leerdoelen

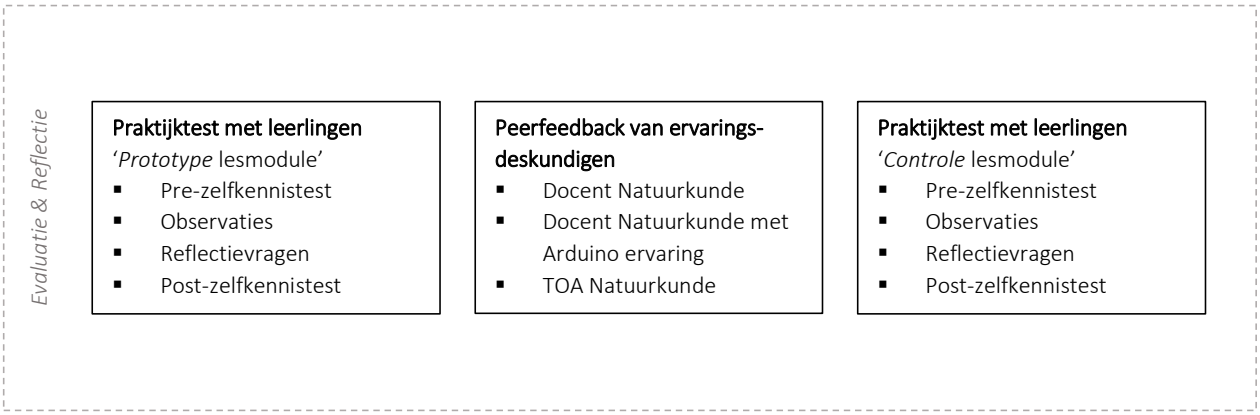
  Arduino
  - Je kunt de onderstaande Arduino hardware en software beschrijven en hiermee werken:
    - Arduino, inclusief voorbeelden van Arduino projecten
    - Microcontroller, inclusief enkele voordelen hiervan, en voorbeelden uit het dagelijks leven waarvoor de microcontroller wordt gebruikt
    - Breadboard
    - Componenten: jumper, LED, weerstand, LDR, en drukknop
    - Arduino IDE, inclusief Arduino sketch, void setup(), void loop(), verifieer, en upload

  Schakelingen | Automatisch regelen
  - Je kunt de drie basisdelen van een geautomatiseerd systeem benoemen, inclusief de hiervoor gebruikte componenten om die taken uit te voeren *(voorkennis)*
  - Je kunt voorbeelden van een meet-, stuur-, en regelsysteem noemen met Arduino projecten *(voorkennis)*
  - Je kunt rekenen met spanning, stroomsterkte en weerstand met behulp van:
    - Wet van Ohm *(voorkennis)*
    - Soortelijke weerstand *(voorkennis)*
    - Serieschakelingen *(voorkennis)*
    - Parallelschakelingen *(voorkennis)*
  - Je kunt elektrische schakelingen tekenen, inclusief serie- en parallelschakelingen *(voorkennis)*

  21e-eeuwse vaardigheden
  - Je kunt een probleem herkennen en planmatig oplossen
  - Je kunt problemen zo formuleren dat ze op te lossen zijn met behulp van ICT
  - Je ontwikkeld kennis en vaardigheden omtrent computer hardware en software
  - Je kunt gezamenlijk een doel behalen en anderen daarbij aanvullen en ondersteunen
- Kennismaken met Arduino
  - Arduino
  - Breadboard
  - Componenten
    - Jumper
    - LED
    - Weerstand
    - LDR
    - Drukknop
  - Software
    - Arduino sketch
    - Void setup() & void loop()
    - Verifieer & Upload
- Werken met Arduino
  - Knipperende LED op de Arduino zelf
  - Een LED aansluiten op de Arduino
  - Een drukknop op de Arduino
  - Weerstanden op de Arduino – Serieschakeling
  - Weerstanden op de Arduino – Parallelschakeling
  - Een LDR op de Arduino
  - LEDs op de Arduino – Serieschakeling
  - LEDs op de Arduino – Parallelschakeling
  - LEDs afwisselen op de Arduino
  - Projecten met Arduino
    - Voetgangersoversteekplaats – Beginner
    - Voetgangersoversteekplaats – Expert

7 RESULTATEN

In dit hoofdstuk worden de uitkomsten van de evaluaties met leerlingen in de praktijk en de peerfeedback van de docenten en TOA Natuurkunde besproken. Zie voor een overzicht hiervan figuur 5.



Figuur 5 – Onderzoeksopzet - Evaluatie & Reflectie

7.1 PRAKTIJKTEST

In de praktijktest zijn voor de *prototype* lesmodule en de *controle* lesmodule een pre- en post-zelfkennistest, observaties, en reflectievragen gebruikt. De uitkomsten van al deze onderdelen worden hieronder gepresenteerd.

7.1.1 (Pre- en post) zelfkennistest

De zelfkennistest, zie bijlage 10.4, bestaat uit een enkele vragenlijst die in zowel de *prototype* lesmodule als *controle* lesmodule vooraf (pre) en achteraf (post) is afgenomen. De uitkomsten van deze vragen zijn voor de *prototype* lesmodule uitgezet in tabel 7, en voor de *controle* lesmodule in tabel 8. In de zelfkennistest beantwoordden leerlingen subjectieve vragen op een schaal van 1 t/m 4 over hun verwachte kennisniveau van Arduino en de leerstof uit het domein Elektriciteit, en over hoe belangrijk zij de betrokken 21^{e}-eeuwse vaardigheden vinden. Leerlingen beantwoordden ook objectieve vragen over het domein Technische Automatisering zoals het noemen van de basisonderdelen van een geautomatiseerd systeem, en het geven van voorbeelden van een meet- stuur- en regelsysteem. Deze vragen zijn door de onderzoeker gescoord van 0 t/m 3 op basis van correctheid van het antwoord.

Tabel 7 – Resultaten van de zelfkennistest in de prototype lesmodule

	Pre-test gemiddelde	Post-test gemiddelde	Verschil	Significantie
Subjectieve vragen; score 1 - 4				
Kennisniveau van hardware en software zoals Arduino	1.6	2.2	0.6	0.012
Kennisniveau Wet van Ohm	2.1	2.5	0.4	0.089
Kennisniveau soortelijke weerstand	2.3	2.5	0.2	0.166
Kennisniveau serieschakelingen	2.8	3.0	0.2	0.396
Kennisniveau parallelschakelingen	2.7	2.8	0.1	0.637
Kennisniveau tekenen van elektrische schakelingen	2.3	2.6	0.3	0.154
Belang van Probleemoplossen	3.7	3.7	0,0	0.739
Belang van ICT-basisvaardigheden	3.0	2.8	- 0.2	0.672
Belang van Computational thinking	2.7	2.9	0.2	0.526
Belang van Samenwerken	3.7	3.7	0,0	1.000
Objectieve vragen; score 0 - 3				
Basisonderdelen geautomatiseerd systeem	0.45	1.1	0.7	0.013
Voorbeelden van meet-, stuur-, en regelsysteem	0.59	1.0	0.4	0.222

Tabel 8 – Resultaten van de zelfkennistest in de controle lesmodule

	Pre-test gemiddelde	Post-test gemiddelde	Verschil	Significantie
Subjectieve vragen; score 1 - 4				
Kennisniveau van hardware en software zoals Arduino	1,8	2.3	0.5	0.039
Kennisniveau Wet van Ohm	1.9	2.0	0.1	0.593
Kennisniveau soortelijke weerstand	2.3	2.5	0.2	0.509
Kennisniveau serieschakelingen	2.8	3.0	0.2	0.317
Kennisniveau parallelschakelingen	2.7	3.0	0.3	0.265
Kennisniveau tekenen van elektrische schakelingen	2.6	2.7	0.1	0.618
Belang van Probleemoplossen	3.7	3.2	- 0.5	0.059
Belang van ICT-basisvaardigheden	2.7	2.4	- 0.3	0.681
Belang van Computational thinking	2.5	2.6	0.1	0.163
Belang van Samenwerken	3.7	3.6	- 0.1	0.705
Objectieve vragen; score 0 - 3				
Basisonderdelen geautomatiseerd systeem	0.38	0.18	- 0.2	0.046
Voorbeelden van meet-, stuur-, en regelsysteem	0.25	0.35	0.1	0.414

Aanvullend is bij beide lesmodules aan leerlingen gevraagd om een toelichting te geven bij hun score omtrent het kennisniveau van hardware en software zoals Arduino. In de toelichting van alle leerlingen samengevat gaf meer dan de helft aan hier vooraf nooit mee te hebben gewerkt. Daarbij had slechts een enkele leerling Arduino of Lego Mindstorms al eens geprobeerd, kende enkele onderdelen, of wist iets van programmeren.

Ook is bij beide lesmodules aan leerlingen gevraagd of Probleemoplossen, ICT-basisvaardigheden, Computational thinking, en Samenwerken aandacht krijgt op school. Hieruit kwam van alle leerlingen samengevat naar voren dat Samenwerken het meeste voorkomt (94% zei ja), waarbij werd aangegeven dat dit bij veel vakken gebeurt, vaak via groepsopdrachten. Daarna volgt Probleemoplossen (78% zei ja), voorkomend bij onder andere projecten, (exacte) vakken, en toetsen. ICT-basisvaardigheden komt minder voor (19% zei ja), en wordt veelal gewijd aan het moeten volgen van het vak Informatica in de bovenbouw. En Computational thinking komt ook minder voor (14% zei ja) en werd eveneens vaak verbonden aan het moeten volgen van Informatica.

7.1.2 Observaties

Observaties zijn door de onderzoeker uitgevoerd tijdens de 2 lessen met de *prototype* lesmodule en tijdens de 2 lessen met de *controle* lesmodule. Hierbij is de *prototype* lesmodule door de onderzoeker zelf gegeven, waarbij leerlingen zelfstandig met het werkboek aan de slag gingen en de onderzoeker rondliep ter ondersteuning. De *controle* lesmodule is gegeven door de betrokken docent met Arduino ervaring, waarbij een klassikale benadering in de vorm van directe instructie is gebruikt met een PowerPoint presentatie als leidraad.

Prototype lesmodule

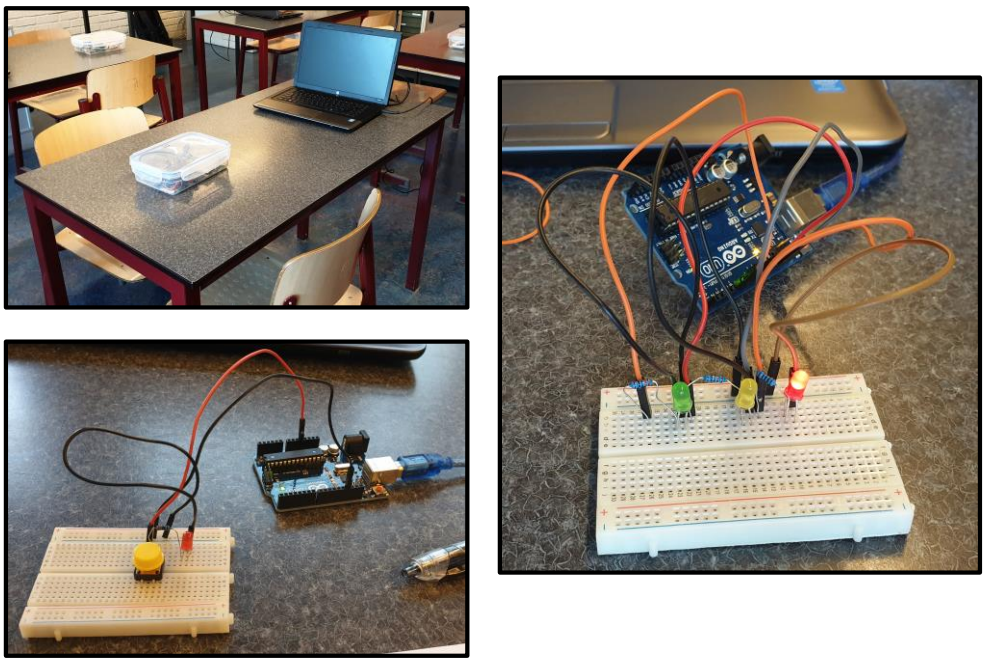
In les 1 van de *prototype* lesmodule viel het volgende op:

- Leerlingen werkten serieus en zelfstandig met het werkboek. Er waren gedurende de les 5-10 vragen.
- De Arduino werd door ongeveer de helft tijdens het lezen bekeken zoals het werkboek als tip gaf.
- Voor sommige leerlingen was het niet duidelijk waar zij de LED konden vinden op de Arduino bij de bijbehorende opdracht. Enkelen hiervan hadden de Arduino nog niet aangesloten op de computer.
- Er waren geen problemen bij het uploaden van de software op de Arduino. Velen hadden de on-board LED al snel aan het knipperen. Bij het aanpassen van de code draaide een enkele leerling de volgorde van aan/uit om, of had aanvullende hulp nodig bij het begrijpen van de code.
- Veel leerlingen gebruikten de knop Verifieer niet bij het uitvoeren van de programma's.
- Een aantal leerlingen wilde het theoretische huiswerk liever thuis doen zodat zij nu nog even verder konden experimenteren met de Arduino.

In les 2 van de *prototype* lesmodule viel het volgende op:

- Leerlingen gaven bij klassikale rondvraag aan dat de theoretische achtergrond na doorlezen duidelijk was. Een opmerking hierbij was dat niet beschreven was hoe het breadboard aangesloten moest worden.
- Veel leerlingen waren lang bezig met het vinden van de juiste weerstand, en dan met name vanaf welke kant je deze moest aflezen. Een afzonderlijke vraag van een tweetal was hoe je de weerstand om aan moest sluiten (zoals dat bij de LED wel van belang is).
- Leerlingen waren snel met het bouwen van de eerste extern knipperende LED. Eén tweetal begon na het oplossen van een probleem enthousiast te klappen dat hij het nu deed.
- Kleine problemen die bij een enkeling voorkwamen:
    - Niet weten hoe het breadboard werkt.
    - Niet weten dat alle jumpers hetzelfde zijn, ongeacht de kleur.
    - Het vergeten terug te zetten van de puntkomma in de code.
    - Het niet voldoende stevig vastprikken van de drukknop op het breadboard.
- Bij een aantal leerlingen werkte de schakeling niet doordat zij de LED verkeerd om aangesloten hadden. Door het stellen van kritische vragen door de docent wist ongeveer de helft hier zelf achter te komen.
- Een aantal groepjes was het gelukt om de drukknop nog toe te passen binnen de les, en twee groepjes waren het nog gelukt om het stoplicht te laten branden. Zie figuur 6.
- Bij klassikale rondvraag gaf ± 1/3 aan wel verder te willen met Arduino, ± 1/3 misschien, en ± 1/3 niet.
- Een drietal leerlingen wilde in eigen tijd nog verder met de Arduino.
- Het modulair inzetten van de lesmodule i.v.m. een verminderd aantal lessen leek te werken.

Tijdens de observaties zijn ook de door leerlingen ingevulde werkboekjes bekeken. Als huiswerk waren de opgaven omtrent Technische automatisering, maar ook het berekenen van de soortelijke weerstand, stroomsterkte, en voorschakelweerstand, en het teken van het schakelschema voor een LED en voorschakel-weerstand in serie opgegeven. Gekeken naar de gemaakte opgaven kwamen leerlingen hier over het algemeen goed uit. De opgave waar het minder goed ging was de opgave waar leerlingen per systeemdeel de component moesten noemen die werd gebruikt om de taken uit te voeren (± 1/3 had het niet goed). Een andere opgave waar het minder goed ging was het berekenen van de waarde van de voorschakelweerstand die nodig is om de LED op de juiste spanning te laten branden (± 4/5 had het niet goed). De leerlingen konden hierbij wel met de wet van Ohm rekenen, maar gebruikten hiervoor de verkeerde spanning. Zij namen de waarde uit de opdracht-omschrijving, terwijl deze verrekend had moeten worden met de bronspanning.



Figuur 6 – Foto's van de werkplek (linksboven), gemaakte drukknop-schakeling door leerlingen (linksonder), en gemaakte stoplicht-opstelling door leerlingen (rechts) tijdens de praktijktest

Controle lesmodule

In les 1 van de *controle* lesmodule van de betrokken docent met Arduino ervaring viel het volgende op:

- De docent liet de Arduino aan de klas zien en gaf een korte introductie over wat je er mee kunt. Hij sloot de Arduino aan en liet de knipperende on-board LED zien. Vervolgens opende de docent een PowerPoint presentatie over Arduino die als leidraad tijdens de lessen zou worden gebruikt.
- De PowerPoint bevatte veel tekst en werd relatief snel doorlopen. De docent gaf aan dat leerlingen zelf de on-board LED gingen laten knipperen. Hierbij besprak hij kort pin 13 en de benodigde functies zoals void setup, void loop, en delay. Hij benoemde ook kort de onderdelen waar leerlingen mee gingen werken, zoals het breadboard en de LED. Hij gaf hierop geen toelichting.
- Leerlingen openden de PowerPoint ook zelf op hun computer en begonnen op verzoek van de docent met het voorbeeldprogramma Blink. Een aantal leerlingen stelde vragen over wat ze ermee moesten, waar ze het programma konden vinden, of waarom de Arduino het niet deed. Een aantal leerlingen had het wel direct door en was al aan het experimenteren. Bij navraag konden deze leerlingen echter niet aangeven wat ze nou precies aan het doen waren.
- Aanvullende vragen van leerlingen waren: wat doen de RX en UX lampjes? Hoe moet je de sketch opslaan? Kun je de LED sneller of langzamer laten knipperen?
- Een aantal leerlingen had niet door dat de tijd bij delay in milliseconden was. Hierdoor hadden ze hem ingesteld op 1 milliseconde en zagen ze hem met het blote oog dus niet knipperen.
- De docent vervolgde klassikaal met de uitleg om een externe LED aan te sluiten. Hij behandelde eerst kort via de PowerPoint het breadboard en hoe deze intern verbonden is. Hij vervolgde met de Arduino schakeling die leerlingen moesten namaken. Hierbij benadrukte de docent dat de lange zijde van de LED het pluspunt is. Hij gaf geen toelichting waarom dat zo was. Aanvullend gaf hij leerlingen de kleurcode die ze voor de weerstand in de schakeling moesten zoeken in hun Arduino set.
- Leerlingen gebruikten de PowerPoint en bouwden de schakeling na. Er ontstonden veel vragen, zoals hoe ze alles moesten aansluiten, welke weerstand ze nodig hadden, maar ook over wat een LED nou was of wat een weerstand deed. Ook hadden sommigen een te grote weerstand gepakt waardoor de LED niet brandde. Bij rondvragen werd duidelijk dat leerlingen ook niet wisten waarom ze een weerstand in serie met de LED moesten schakelen. De indruk werd gewekt dat leerlingen niet goed wisten wat ze deden.
- De docent vervolgde klassikaal hoe leerlingen de pin konden veranderen. Hierbij werd de code getoond en kort per onderdeel aangegeven waar ze wat moesten veranderen.
- Eén leerling was op eigen initiatief LEDs in serie aan het zetten en liep vast op hoe hij de schakeling zo moest maken dat het werkte.

In les 2 van de *controle* lesmodule van de betrokken docent met Arduino ervaring viel het volgende op:

- De docent begon klassikaal dat leerlingen de pin-ingang gingen veranderen en een drukknop gingen gebruiken om hiermee een stoplicht te kunnen maken die reageert op de drukknop. De docent zette de code op het scherm en de leerlingen konden de Arduino opstelling voorin de klas komen bekijken. Op het bord zette hij de te nemen stappen om het programma te vinden en ter illustratie de verkeerssituatie. Een aanvullende tip die werd gegeven was om online informatie te zoeken (enkele leerlingen hadden dit succesvol gedaan). Een aantal leerlingen had vragen, en een aantal kwam er zelfstandig uit.
- Een tweetal had de lampjes in parallel gezet i.p.v. serie. Via doorvragen van de docent kwamen ze er zelf achter dat het niet klopte, want het ene lapje zou niet meer werken als de andere werd verwijderd.
- Een aantal leerlingen vroeg hoe de drukknop werkte, en of de kleur uitmaakte.
- De meeste leerlingen was het uiteindelijk gelukt om de schakeling na te bouwen. De indruk werd echter wel gewekt dat leerlingen niet wisten wat ze precies aan het doen waren.
- De leerlingen ontvingen een cijfer voor de lesmodule op basis van het proces dat zij hebben doorgemaakt.

7.1.3 Reflectievragen

De reflectievragen, zie bijlage 10.5, bestaan uit een vragenlijst die in zowel de *prototype* lesmodule als *controle* lesmodule achteraf is afgenomen. Leerlingen beantwoordden subjectieve vragen op een schaal van 1-4 over hoe zij de lesmodule hebben ervaren. De uitkomsten van deze vragen zijn voor beide lesmodules uitgezet in tabel 9. Aanvullend is aan leerlingen gevraagd wat zij goed (Tops) en minder goed (Tips) vonden aan de lesmodule.

Tabel 9 – Resultaten van de reflectievragen in de prototype en controle lesmodule

	Prototype lesmodule - gemiddelde	Controle lesmodule - gemiddelde	Verschil	Significantie
Subjectieve vragen; score 1 - 4				
Arduino is leerzaam om mee te (leren) werken	3.4	2.9	0.5	0.033
Arduino is een goede toevoeging aan het hoofdstuk Schakelingen | Automatisch regelen	3.0	2.6	0.4	0.186
Arduino is een goede manier om te werken aan kennis en vaardigheden voor de toekomst	3.0	2.5	0.5	0.059

Prototype lesmodule

De meest voorkomende 'Tops', ofwel wat leerlingen goed/leuk vonden aan de lesmodule, zijn samengevat:

- Dat je zelf iets kunt maken en uitproberen. *(10 leerlingen)*
- Dat de uitleg en opdrachten goed en duidelijk zijn. *(6 leerlingen)*
- Dat het goed aansluit op de leerstof. *(3 leerlingen)*
- Dat je iets praktisch doet met elektriciteit. *(2 leerlingen)*
- Dat je nieuwe dingen probeert. *(2 leerlingen)*
- Dat je zelf kunt programmeren/ziet hoe het werkt (en het helemaal niet zo moeilijk is). *(2 leerlingen)*

De meest voorkomende 'Tips', ofwel wat leerlingen minder goed/leuk vonden aan de lesmodule, zijn samengevat:

- Dat er te weinig tijd was. *(8 leerlingen)*
- Dat er te veel huiswerk was opgegeven. *(7 leerlingen)*
- Dat het huiswerk niet in volgorde was met de leerstof uit het werkboek. *(5 leerlingen)*
- Dat het soms onduidelijk was wat je moest doen/waar de informatie stond. *(3 leerlingen)*

Controle lesmodule

De meest voorkomende 'Tops', ofwel wat leerlingen goed/leuk vonden aan de lesmodule, zijn samengevat:

- Dat het wat anders is dan de normale les (leuker, leerzamer, interessanter). *(10 leerlingen)*
- Dat je veel zelf kan doen/bezig kan zijn. *(4 leerlingen)*
- Dat het interessant en leerzaam is. *(3 leerlingen)*
- Dat je het zelf moet uitzoeken. *(2 leerlingen)*

De meest voorkomende 'Tips', ofwel wat leerlingen minder goed/leuk vonden aan de lesmodule, zijn samengevat:

- Dat er meer uitleg moet worden gegeven (zodat je weet wat je moet doen/aan het doen bent). (*13 leerlingen)*
- Dat je voorafgaand al informatie krijgt (op papier zodat je het nog een keer rustig kunt doorlezen). *(2 leerlingen)*
- Dat de PowerPoint onduidelijk is. *(2 leerlingen)*

7.2 PEERFEEDBACK DOCENTEN EN TOA NATUURKUNDE

Ter verbetering van het *prototype* van de Arduino lesmodule zijn de betrokken docent Natuurkunde met Arduino ervaring, een bovenbouw docent Natuurkunde die in het verleden ook les heeft gegeven aan havo/vwo-leerjaar 3, en de TOA van de vaksectie Natuurkunde afzonderlijk gevraagd om hier feedback op te geven.

7.2.1 TOA Natuurkunde

De TOA Natuurkunde gaf het volgende aan:

- Het is een complete lesmodule met goede opbouw, opmaak, leerdoelen, en theoretische achtergrond.
- Een handleiding is handig voor het kunnen terugzoeken van informatie (papier voorkeur t.o.v. digitaal).
- Voor deze lesmodule zijn zeker 4 lessen nodig.
- De lesmodule past beter in havo/vwo-leerjaar 4 omdat het programmeren waarschijnlijk uitdagend is.
- Er is veel te lezen theorie voordat leerlingen aan het werk kunnen, ondanks dat de theorie wel nodig is.
- De na te bouwen schakelschema's zijn duidelijk en overzichtelijk.
- Arduino vergt een extra denkstap en meer inzet ten opzichte van de logischer en overzichtelijker reguliere methode met lampjes en kabeltjes. Daarbij: hoe zinvol is leren programmeren binnen Natuurkunde?
- Leerlingen vinden met name parallelschakelingen lastig, wat ook te maken heeft met het overzicht van alle benodigde kabels. Met Arduino heb je hierop nog minder overzicht vanwege de kleine onderdelen.

7.2.2 Docent Natuurkunde met Arduino ervaring

De Natuurkunde docent met Arduino en havo/vwo-leerjaar 3 ervaring gaf het volgende aan:

- De lesmodule is na enkele kleine aanvullingen goed qua doelen, opbouw, theorie, en opdrachten.
- Aan de 21^{e}-eeuwse doelen zou nog 'het realiseren van eigen projecten' toegevoegd kunnen worden.
- Het niveau en de hoeveelheid van de lesmodule is zeker geschikt voor havo/vwo-leerjaar 3.
- Technische automatisering, ondanks niet gegeven als lesstof dit jaar, past goed in de lesmodule vanwege de grote toepasbaarheid op de Arduino (en dan met name het meet-, stuur-, en regelsysteem).
- Voor deze lesmodule zijn minimaal 4 lessen nodig. Hierbij kunnen leerlingen zelfstandig aan het werk.
- De lesmodule zou niet worden gegeven als onderdeel in het curriculum omdat programmeren niet in het curriculum zit. De lesmodule zou wel heel geschikt zijn als extra-curriculair onderwijs voor leerlingen die tijdig klaar zijn met de leerstof, of bij talentonderwijs.
- Bij meer focus op programmeren zou in de opzet de drukknop- en LDR-opdrachten later behandeld kunnen worden met aanvullende programmeervaardigheden zoals variabelen en de if- en for-loop.
- In de theoretische achtergrond zou de Ohmse weerstand nog kunnen worden toegevoegd.
- Aan de theorie over Software zouden nog kunnen worden toegevoegd: variabelen; dat de software feedback geeft bij fouten; en dat belangrijke functies gekleurd worden weergegeven.
- Het is goed dat leerlingen eerst zelf schakelingen moeten tekenen, waarna ze deze via de Arduino zien.
- De LDR is een belangrijke component voor het leren omdenken omdat dit een dynamisch systeem is.
- Het is belangrijk dat leerlingen bij Arduino succeservaringen ondervinden.

7.2.3 Docent Natuurkunde

De Natuurkunde docent met havo/vwo-leerjaar 3 ervaring gaf het volgende aan:

- De lesmodule is qua theoretische achtergrond en opgaven duidelijk.
- De lesmodule zal voor havo/vwo-leerjaar 3 geen probleem vormen. Bij havo kan het dan praktisch worden gehouden terwijl vwo dieper ingaat op hoe de systemen inhoudelijk werken.
- De lesmodule zou ook geschikt kunnen zijn voor havo/vwo-leerjaar 4, en ook voor in een talentweek.
- Leerlingen hebben over het algemeen meer moeite met het begrip spanning. Hierdoor zou de docent de reguliere methode met lampjes en weerstandjes als eerste stap gebruiken en Arduino als tweede stap.
- Voor deze lesmodule zijn 3 lessen nodig, waarbij de miniprojecten uit de lesmodule dan zelfstandig in de practicumruimte gemaakt kunnen worden.
- Arduino is geen onderdeel van het examenprogramma, maar zou eigenlijk een overkoepelend onderdeel moeten zijn binnen de school. Dit kan ook binnen het vak Natuurkunde op een manier zoals deze.

8 EINDPRODUCT

Het *eindproduct* in dit onderzoek is een verbetering van het *prototype* van de Arduino lesmodule, zie *§6 Ontwerp*, op basis van de uitkomsten van beide lesmodules uit de praktijktest en de ontvangen peerfeedback van de docenten en TOA Natuurkunde. Het *eindproduct* is te vinden in bijlage 10.6 en 10.7.

In grote lijnen zijn de volgende verbeterpunten doorgevoerd:

Werkboek leerling

- Wijzigingen in de theoretische achtergrond omtrent woordkeuze.
- Toevoegingen in de theoretische achtergrond, zoals:
  - De Arduino IDE gebruikt voor tijd milliseconden, geeft feedback bij fouten, kleurt de belangrijkste functies, en vereist niet dat het programma moet worden opgeslagen voor je deze kunt uitvoeren.
  - Het breadboard hoeft niet apart aangesloten te worden.
  - De kleur van een jumper heeft geen invloed op zijn werking.
  - Benadrukken van bepaalde informatie omdat er overheen wordt gelezen, zoals het aansluiten van de LED, of vanaf welke kant de kleurcode van een weerstand moet worden afgelezen.
  - Aanvullende informatie over het aflezen van de kleurcode van een weerstand.
  - Aanvullende context, zoals de Arduino bronspanning vergelijken met de spanning van een batterij, of de accolades in de Arduino IDE vergelijken met de soms benodigde haakjes in een rekenmachine.
  - Verwijzen naar online ondersteuning voor Arduino.
  - Aangeven dat de Arduino hardware en software nog programma's van medeleerlingen kan bevatten.
- Toevoegingen in de opdrachten, zoals:
  - De drukknop moet stevig worden vastgedrukt op het breadboard om te werken.
  - Vraagstelling wijzigen bij het zoeken naar de zelf aan te sturen LED op de Arduino (leerlingen hadden vaak de Arduino nog niet aangesloten op de computer).
  - Vraagstelling wijzigen bij het noemen van de componenten die de taken uitvoeren in een geautomatiseerd systeem (leerlingen dachten veelal in termen van Arduino en microcontrollers).
  - Vraagstelling wijzigen bij het berekenen van de waarde van de voorschakelweerstand die nodig is om de LED op de juiste spanning te laten branden (leerlingen vergaten een stap in de berekening).

Handleiding docent

- Benoemen in de voorbereiding dat enige voorkennis van Arduino wel vereist is.

9 DISCUSSIE & CONCLUSIE

In dit hoofdstuk worden de discussie en conclusie van dit onderzoek besproken.

9.1 DISCUSSIE

In de discussie worden de gebruikte methode en resultaten uit dit onderzoek besproken. De resultaten bestaan uit de zelfkennistest, observaties, reflectievragen, en peerfeedback.

9.1.1 Methode

Met de gebruikte methode in dit onderzoek kon de onderzoeksvraag naar tevredenheid worden beantwoord, en konden de gestelde doelen en het programma van eisen worden behaald. Het theoretisch kader bood, met het vooronderzoek als inspiratie, voldoende ondersteuning om een *prototype* lesmodule te kunnen ontwikkelen. Idealiter zou eerst een conceptversie zijn geëvalueerd via peerfeedback van de docenten en TOA Natuurkunde, alvorens tot een *prototype* over te gaan. Echter is deze stap door omstandigheden niet gerealiseerd met als gevolg dat er voor de stap van *prototype* naar *eindproduct* meer aanpassingen moesten worden verricht, die vervolgens ook niet meer met leerlingen konden worden geëvalueerd. De uitgevoerde praktijktest gaf bruikbare informatie over zowel de *prototype* lesmodule alsook de *controle* lesmodule. Hierbij konden beide lesmodules goed met elkaar worden vergeleken. De leerdoelen van de betrokken docent waren echter meer gericht op programmeren, en minder overeenkomstig met de leerdoelen uit dit onderzoek. Aangezien programmeren vaak wel nodig is bij het werken met Arduino zou bij een vervolgstap nog kunnen worden gekeken naar de vereisten vanuit het examenprogramma van Informatica, of naar het betrekken van docenten uit dit vakgebied.
In de praktijktest waren oorspronkelijk 4 lessen per lesmodule ingepland. Echter is dit aantal door omstandigheden teruggebracht naar 2 lessen per lesmodule. Hierdoor ontbrak een groot deel van de evaluatie en kon het werkboek slechts worden uitgevoerd tot aan *Een drukknop op de Arduino (zie Tabel 6)*. Hierdoor is er geen mogelijkheid geweest om de weerstanden in serie- en parallelschakeling, de lichtgevoelige weerstand (LDR), LEDs in serie- en parallelschakeling, LEDs in een stoplicht, en de miniprojecten met leerlingen te evalueren. Ook is hierdoor de planning gewijzigd, waardoor het werkboek in een andere volgorde is uitgevoerd en de leerlingen ook huiswerk opgedragen kregen. De reden hiervoor is dat leerlingen op deze manier in de les nog zoveel mogelijk konden werken met de Arduino doordat zij de theoretische leerstof en opdrachten thuis hebben uitgevoerd. Dit laat echter wel zien dat het mogelijk is om met aanvullend huiswerk in twee lessen de kennismaking met Arduino te kunnen uitvoeren, aangevuld met de opdrachten van het domein Technische automatisering, en enkele opdrachten van het domein Elektriciteit. Een punt van aandacht hier is dat de betrokken leerlingen niet het domein Technische automatisering behandeld hadden gekregen. Dit was echter geen belemmering voor het maken van de opdrachten omdat deze informatie in het lesboek te vinden was. Wat betreft de hoeveelheid respondenten waren dit er voldoende. Idealiter zou er na een verbeterslag nog met een aanvullende klas worden getest, maar dit was gezien de beschikbare tijd niet mogelijk. Een aandachtspunt hierbij is wel dat er in dit onderzoek enkel met een vwo-klas is getest. De verwachting is dat met een havo-klas soortgelijke resultaten zouden zijn behaald, maar dit zou als vervolgstap getest moeten worden alvorens hier uitspraken over te kunnen doen.
De gebruikte instrumenten in de praktijktest waren voldoende en leidden met de gebruikte analyse tot interessante en bruikbare informatie. Echter is het wel de vraag hoe groot de invloed is van het maar kunnen evalueren van 2 lessen, waarbij ook nog eens de vragenlijsten op zeer korte termijn achter elkaar zijn ingevuld. Daarbij is de zelfkennistest in grote mate subjectief, waardoor over deze uitkomsten niet eenzijdig conclusies kunnen worden getrokken. Het was binnen de beschikbare tijd echter niet mogelijk om leerlingen objectief te testen op hun kennisniveau, wat in een vervolgstap wel aanbevolen zou worden.
De ontvangen peerfeedback van de docenten en TOA Natuurkunde gaven bruikbare informatie over de *prototype* lesmodule. Hierbij was het zinvol om een TOA, een docent met havo/ vwo-leerjaar 3 ervaring, en een docent met enkele jaren Arduino ervaring in havo/vwo-leerjaar 3 te betrekken.

Een aandachtspunt is dat de ontwikkelde lesmodule verondersteld dat de docent enige basiskennis heeft van Arduino, zoals het zelfstandig kunnen aanschaffen van de benodigde materialen, installeren van de Arduino IDE op de computers, en de kennis benodigd voor het kunnen begeleiden van de leerlingen en hun werkboek. De verwachting is dat de meeste docenten deze kennis via enige zelfstudie of een (online) cursus kunnen verkrijgen.

9.1.2 Zelfkennistest

De resultaten van de zelfkennistest laten zien dat leerlingen in zowel de *prototype* lesmodule als *controle* lesmodule een statistisch significante toename in kennisniveau hebben *ervaren* van hardware en software zoals Arduino. Dit is te verklaren doordat leerlingen in beide lessen veel met Arduino hebben gewerkt. Er is echter relatief weinig verschil tussen beide lesmodules. Dit is opvallend omdat in de *prototype* lesmodule veel meer achtergrondinformatie is aangeboden. Het lijkt erop dat dit slechts een kleine invloed heeft gehad op de perceptie van leerlingen. Aanvullend is het interessant om te zien dat meer dan de helft van alle betrokken leerlingen nog nooit met hardware en software zoals Arduino hebben gewerkt, waarvoor deze lesmodule dus goed uitkwam.

De resultaten van het verwachte kennisniveau van de wet van Ohm, soortelijke weerstand, serieschakelingen, parallelschakelingen, en het tekenen van elektrische schakelingen lijken voor beide lesmodules een lichte positieve tendens te laten zien. Er was echter een statistisch significante toename verwacht in de *prototype* lesmodule bij het verwachte kennisniveau van de wet van Ohm, soortelijke weerstand, en het tekenen van elektrische schakelingen, omdat leerlingen hierover opdrachten hebben moeten maken. Dat dit echter niet zo is kan komen doordat leerlingen in dit onderzoek hierover slechts enkele opdrachten hebben kunnen maken.

De resultaten van de mate waarin leerlingen de betrokken 21^{e}-eeuwse vaardigheden belangrijk vinden laat in de *prototype* en *controle* lesmodule zien dat leerlingen Probleemoplossen en Samenwerken erg belangrijk vinden, en dat dit op school ook veel aandacht krijgt. ICT-basisvaardigheden en Computational thinking worden door leerlingen in beide lesmodules respectievelijk als bovengemiddeld en middelmatig belangrijk gevonden, en krijgen op school weinig aandacht. Het is niet duidelijk of tussen het belang en de aandacht een verband zit. Echter laat deze uitkomst wel zien dat de lesmodule een rol kan spelen om deze vaardigheden onder de aandacht te brengen. Tot slot, wanneer gekeken wordt naar de mening van leerlingen voor- en achteraf, valt alleen een onverklaarbare negatieve tendens op bij het belang van Probleemoplossen in de *controle* lesmodule.

De resultaten van de objectieve vragen over de basisonderdelen van een geautomatiseerd systeem, en het noemen van voorbeelden van meet-, stuur-, en regelsystemen, waren in de pre-test erg laag bij zowel de *prototype* als *controle* lesmodule. Een logische verklaring hiervoor is dat beide onderwerpen niet behandeld zijn in de les. In vergelijking met de post-test toonde de *prototype* lesmodule bij de vraag over basisonderdelen een statistisch significante verbetering, en de *controle* lesmodule een statistisch significante verslechtering. Bij voorbeelden van meet-, stuur- en regelsystemen waren de verschillen tussen de pre- en post-test statistisch niet significant, al lijkt er een positieve tendens zichtbaar in de *prototype* lesmodule. Al deze resultaten kunnen wellicht worden verklaard doordat leerlingen in de *prototype* lesmodule opdrachten hebben moeten maken over Technische automatisering.

Een aandachtspunt is dat de meeste vragen in deze test subjectief waren. Hierdoor kunnen over deze vragen slechts conclusies worden getrokken over de eigen interpretatie van leerlingen. Voor een vervolgstap zou de aanbeveling daarom zijn om deze onderwerpen op objectieve wijze te testen.

9.1.3 Observaties

De observaties laten zien dat de *prototype* lesmodule voldoende duidelijke informatie en opdrachten lijkt te bevatten om leerlingen zelfstandig te laten werken. Verbeteringen die kunnen worden gemaakt zijn het eerst laten aansluiten van de Arduino alvorens leerlingen op zoek te laten gaan naar de zelf aan te sturen LED. Ook kan de informatie over het vinden van de juiste weerstand aan de hand van kleurcodes worden verbeterd aangezien leerlingen daar relatief lang mee bezig waren. Aanvullend kan bepaalde informatie nog extra worden benadrukt omdat leerlingen hier overheen lijken te lezen, zoals het aansluiten van de LED. Ook kan de opgave over het noemen van de componenten die de taken uitvoeren in een geautomatiseerd systeem (ofwel: sensor, verwerker, actuator) qua vraagstelling worden verbeterd. Leerlingen dachten namelijk direct in termen van Arduino en microcontrollers. Een andere te verbeteren opgave is het berekenen van de waarde van de voorschakelweerstand die nodig is om de LED op de juiste spanning te laten branden. Leerlingen gebruikten in hun berekening de waarde uit de opdrachtomschrijving, terwijl deze eigenlijk nog verrekend had moeten worden met de bronspanning.

De *controle* lesmodule gebruikte een klassikale benadering met een PowerPoint als leidraad en een geprepareerde Arduino als voorbeeld. Dit had als gevolg dat het voor sommige leerlingen te weinig ondersteuning gaf of te snel ging. Tevens werden de onderdelen van de Arduino wel benoemd, maar vaak niet verder toegelicht. Hierdoor is de kans groot dat leerlingen niet weten wat ze om welke reden aan het doen zijn, wat ook te zien was aan de

gestelde vragen door leerlingen. Leerlingen waren hierdoor ook veelal aan het kopiëren wat de docent deed, en werden niet gestimuleerd om zelfstandig na te denken over de uitgevoerde handelingen. Hierbij was het doel van de docent meer het aanleren van programmeren, denken in algoritmes, en ontwikkelen van projecten, dan het voortbouwen op het behandelde hoofdstuk in de les. Leerlingen kregen in deze lesmodule echter wel veel vrijheid, waardoor ze zelf creatief aan de gang konden gaan. De leerlingen werden uiteindelijk beoordeeld op hun proces. Wat opvalt in de *controle* lesmodule is dat veel van de gestelde vragen en ontbrekende kennis van leerlingen wel in het werkboek van de ontwikkelde *prototype* lesmodule aan de orde komt. Ook is de manier van beoordeling in de *controle* lesmodule erg subjectief. De *prototype* lesmodule laat de docent hierin vrij, maar suggereert wel de gemaakte opdrachten uit het werkboek te gebruiken voor een objectieve beoordeling. De aanbeveling is hier dan ook om een werkboek te gebruiken voor de leerlingen en de opdrachten te gebruiken voor de beoordeling.

Een aandachtspunt is dat de observaties uitgevoerd zijn door de onderzoeker zelf, waardoor er mogelijk minder objectief is geobserveerd. In een vervolgstap zou hiervoor een onafhankelijke observator kunnen worden ingezet.

9.1.4 Reflectievragen

De resultaten van de reflectievragen laten zien dat leerlingen het bij de *prototype* lesmodule zeer leerzaam vonden om met de Arduino te (leren) werken, tegenover bovengemiddeld leerzaam bij de *controle* lesmodule. Dit verschil tussen beide modules is statistisch significant en kan wellicht worden verklaard doordat leerlingen in de *prototype* lesmodule meer leerstof ontvingen en opdrachten maakten waardoor zij meer de indruk kregen iets te leren. Leerlingen gaven ook aan dat Arduino in de *prototype* lesmodule een bovengemiddeld goede toevoeging is aan het hoofdstuk Schakelingen | Automatisch regelen die zij in de les gebruiken, tegenover middelmatig bij de *controle* lesmodule. Het verschil tussen beide modules is echter niet statistisch significant. Dit was onverwacht, aangezien de *prototype* lesmodule veel leerstof uit het hoofdstuk betrekt terwijl de *controle* lesmodule meer focust op het programmeren. Een mogelijke verklaring hiervoor is dat leerlingen niet de gehele lesmodule hebben kunnen doorlopen. Tot slot gaven leerlingen aan dat Arduino in de *prototype* lesmodule een bovengemiddeld goede manier is om te werken aan kennis en vaardigheden voor de toekomst, tegenover middelmatig bij de *controle* lesmodule. Het verschil tussen beide modules is echter niet statistisch significant.

De benoemde positieve punten voor de *prototype* lesmodule laten voornamelijk zien dat leerlingen het erg leuk vinden om zelf dingen te maken en uitproberen met Arduino. Ook werd benadrukt dat de uitleg en opdrachten van de lesmodule goed en duidelijk zijn, en het goed aansluit op de leerstof. Verbeterpunten waren het tekort aan tijd, teveel aan huiswerk, dat het huiswerk niet in volgorde was met de leerstof uit het werkboek, en dat het soms onduidelijk was wat je moest doen/waar de informatie stond. De eerste drie verbeterpunten kunnen waarschijnlijk worden gewijd aan het moeten inkorten van het aantal lessen. Hierdoor is de volgorde in het werkboek gewijzigd, is er huiswerk opgegeven, en was er dus te weinig tijd om de hele lesmodule te kunnen doorlopen. Daarbij was het huiswerk al voor de volgende dag opgegeven. Het laatste verbeterpunt kan wellicht ook aan deze argumenten te wijten zijn, maar was niet dusdanig gedetailleerd beschreven dat dit kan worden aangenomen.

De benoemde positieve punten voor de *controle* lesmodule laten voornamelijk zien dat leerlingen het positief vinden dat de lesmodule iets anders is dan de normale les (leuker, leerzamer, interessanter), je veel zelf kan doen/bezig kan zijn, het interessant en leerzaam is, en dat je het zelf moet uitzoeken. Verbeterpunten waren dat er meer uitleg moet worden gegeven, je voorafgaand al informatie moet krijgen, en dat de PowerPoint onduidelijk is. De verbeterpunten kunnen te wijten zijn aan het onvoldoende of te snel behandelen van de leerstof via de PowerPoint of klassikale uitleg. Wellicht dat het gebruiken van een werkboek met theoretische achtergrond en opdrachten zoals in de *prototype* lesmodule hier een passende oplossing voor is.

9.1.5 Peerfeedback

De peerfeedback op de *prototype* lesmodule van zowel de docenten als TOA Natuurkunde is positief in termen van doelen, opbouw, theoretische achtergrond, en opgaven. Hierbij schatten ze voor de lesmodule 3 à 4 benodigde lessen, wat overeenkomt met de vooraf in dit onderzoek gestelde eis van 4 lessen. De docenten verwachten dat de lesmodule qua hoeveelheid en niveau goed aansluit op het beoogde havo/vwo-leerjaar 3, terwijl de TOA verwacht dat het beter past bij leerjaar 4 doordat het programmeren een uitdaging is. Ook twijfelt de TOA over de zin van het leren programmeren binnen Natuurkunde, en geeft hij voor de behandeling van het domein Elektriciteit de voorkeur aan de reguliere methode met lampjes en kabeltjes. Eén van de docenten is het

met dit laatste eens omdat leerlingen spanning een lastig begrip vinden, maar ziet de Arduino lesmodule juist als goede vervolgstap. Deze docent ziet programmeren ook als overkoepelend schoolonderdeel dat geïntegreerd binnen Natuurkunde terug mag komen, maar eveneens geschikt is voor in een talentweek. De docent met Arduino ervaring ziet deze lesmodule vanwege het programmeren eerder als extra-curriculair of talentonderwijs, met nadruk op het programmeren. Hieruit kan worden meegenomen dat de ontwikkelde Arduino lesmodule afhankelijk van de beschikbare tijd en docent zowel een rol kan spelen in het curriculum als erbuiten, maar dat de nadruk wel op de natuurkundige aspecten moet blijven liggen.

9.2 CONCLUSIE

In dit onderzoek is een Arduino lesmodule ontworpen, ontwikkeld, en geëvalueerd voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21^{e}-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering. Hiermee lijkt antwoord te zijn gegeven op de in dit onderzoek gestelde onderzoeksvraag en doelen. Echter kan door de beperkte evaluatie niet met zekerheid worden gezegd in welke mate leerlingen deze doelen daadwerkelijk hebben behaald. Wel kan op basis van de wetenschappelijke onderbouwing, de praktijktest met leerlingen, en de peerfeedback van ervaringsdeskundigen worden aangenomen dat de in dit onderzoek ontwikkelde lesmodule hier grotendeels aan lijkt te voldoen. De 21^{e}-eeuwse vaardigheid Probleemoplossen komt met name terug via de miniprojecten aan het einde van de lesmodule, waarbij leerlingen een probleem analyseren en hier vervolgens oplossingen voor bedenken en realiseren met behulp van Arduino, Deze miniprojecten konden door omstandigheden echter niet worden geëvalueerd. Computational thinking lijkt enigszins te zijn behaald doordat leerlingen op zichtbaar positieve wijze kennis hebben gemaakt met programmeren door het gebruiken van algoritmes en procedures om opdrachten uit te voeren met de Arduino. De ICT-basisvaardigheden lijken ook enigszins te zijn behaald, doordat leerlingen kennis op hebben kunnen doen en gewerkt hebben met zowel hardware als software, en hebben kunnen inzien welke functies hiermee mogelijk zijn. In de zelfreflectie van leerlingen kwam ook naar voren dat deze vaardigheid statistisch significant is toegenomen. Tot slot lijkt Samenwerken ook enigszins behaald door het op zichtbaar positieve wijze samenwerken van leerlingen in groepsverband. Hierbij communiceerden leerlingen met elkaar terwijl zij verschillende taken uitvoerden. Het domein Elektriciteit in de lesmodule biedt leerlingen door de wetenschappelijke onderbouwing aanvulling op, en versterking van, de voorkennis. Echter is dit domein niet in zijn geheel met leerlingen geëvalueerd. De wetenschappelijke onderbouwing, met aanvullend de observaties en peerfeedback van ervaringsdeskundigen, geven aanleiding om te stellen dat dit doel wel grotendeels is behaald. Het domein Technische automatisering biedt leerlingen door de wetenschappelijke onderbouwing ook aanvulling op, en versterking van, de voorkennis. Dit domein is wel in zijn geheel met leerlingen geëvalueerd. Hierbij bleek het door de leerlingen kunnen noemen van de basisonderdelen van een geautomatiseerd systeem statistisch significant toegenomen. De wetenschappelijke onderbouwing, met aanvullend de observaties en peerfeedback van ervaringsdeskundigen, geven aanleiding om te stellen dat ook dit doel grotendeels is behaald.
Aanvullend toont dit onderzoek dat het mogelijk lijkt om met aanvullend huiswerk in 2 lessen de kennismaking met Arduino te kunnen uitvoeren, aangevuld met de opdrachten van het domein Technische automatisering, en enkele opdrachten van het domein Elektriciteit.

De ontwikkelde Arduino lesmodule lijkt in positieve zin geschikt voor het Natuurkunde curriculum van havo/vwo-leerjaar 3, maar wellicht ook voor extra-curriculair onderwijs of een projectweek. Hierbij wordt van de docent wel enige kennis van Arduino vereist, welke via zelfstudie of cursus kan worden opgedaan. Aandachtspunten uit dit onderzoek zijn de gebruikte subjectieve zelfkennisvragen en de uitgevoerde observaties door de onderzoeker zelf. Daarnaast is de *prototype* lesmodule niet in zijn geheel in de praktijk getest. De aanbeveling is daarom om het *eindproduct* te evalueren in de praktijk met leerlingen uit havo- en vwo-leerjaar 3, waarbij de beoogde 4 lesuren worden gebruikt en een objectievere methode ter beoordeling van de resultaten.

Concluderend biedt dit onderzoek een eerste opzet van een Arduino lesmodule voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21^{e}-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering. Deze lesmodule kan door docenten naar eigen inzicht worden ingezet, en bereidt leerlingen daarmee hopelijk enigszins voor op de snel veranderende samenleving waarin digitalisering en technologische ontwikkelingen een grote rol spelen.

10 REFERENTIES

Arduino. (2019). Teaching, Inspiring and Empowering! Geraadpleegd op 12 juli, 2019, geraadpleegd van https://www.arduino.cc/en/main/education

Blum, J. (2013). *Exploring Arduino*. John Wiley & Sons, Inc.

Boswell-Bèta. (2019). James Boswell Cursus | Workshop Arduino voor Docenten (Docenten). Geraadpleegd op 30 mei, 2019, geraadpleegd van http://www.boswell-beta.nl/workshop/arduino

Boxall, J. (2013). *Arduino Workshop*. William Pollock.

College voor Toetsen en Examens. (2017a). *Natuurkunde havo: Syllabus centraal examen 2019*. Utrecht. Geraadpleegd van https://www.examenblad.nl/examenstof/syllabus-2019-natuurkunde-havo/2019/f=/natuurkunde_havo_2_versie_2019.pdf

College voor Toetsen en Examens. (2017b). *Natuurkunde vwo: Syllabus centraal examen 2019*. Utrecht. Geraadpleegd van https://www.examenblad.nl/examenstof/syllabus-2019-natuurkunde-vwo/2019/f=/natuurkunde_2_versie_vwo_2019.pdf

Dingemans, P., & Philippi, H. (2015). Lessen serie embedded systemen (arduino). Geraadpleegd op 30 mei, 2019, geraadpleegd van https://maken.wikiwijs.nl/60119/lessen_serie_embedded_systemen__arduino_

Duifhuis, P., & van Asseldonk, K. (2018). Robotica in de klas. *NVOX*, 174–175. Geraadpleegd van https://newsroom.nvon.nl/files/default/174-175_LR_NVOX18_TDS_NR-4.pdf

Ebbens, S., & Ettekoven, S. (2015). *Effectief leren* (4th ed.). Groningen: Noordhoff Uitgevers.

Gagné, R. M., Briggs, L. J., & Wager, W. W. (1992). *Principles of instructional design* (4th ed.). New York: Harcourt Brace Jovanovich College Pub.

Herger, L., & Bodarky, M. (2015). Engaging Students with Open Source Technologies and Arduino. In *IEEE Integrated STEM conference* (Vol. 5, pp. 27–32). https://doi.org/10.1109/ISECon.2015.7119938

Kampen, A. (2019). Arduino lessenserie. Geraadpleegd op 30 mei, 2019, geraadpleegd van http://www.degeflipteklas.nl/arduino-lessenserie/

Killian, F. (2015). *MicroControllers programmeren met ARDUINO*. Geraadpleegd van https://deleidseaanpak.nl/uploads/editor/Microcontrollers_programmeren.pdf

Kortland, K., & Poorthuis, H. (2017). Lessenseries. In *Handboek Natuurkundedidactiek* (pp. 20–24). Epsilon Uitgaven.

Leraar24. (2019). 21e eeuwse vaardigheden in het onderwijs: leerlingen voorbereiden op de toekomst. Geraadpleegd op 22 april, 2019, geraadpleegd van https://www.leraar24.nl/21e-eeuwse-vaardigheden-in-het-onderwijs-leerlingen-voorbereiden-op-de-toekomst/

Malmberg. (2014a). *Nova; 3 havo* (4th ed.). 's-Hertogenbosch: Malmberg.

Malmberg. (2014b). *Nova; 3 vwo|gymnasium* (4th ed.). 's-Hertogenbosch: Malmberg.

Margolis, M. (2012). *Arduino Cookbook* (2nd ed.). O'Reilly Media, Inc.

MIT Media Lab. (2019). Scratch. Geraadpleegd op 30 mei, 2019, geraadpleegd van https://scratch.mit.edu/

Mooldijk, A., & Pols, F. (2017). Technische automatisering. In *Handboek Natuurkundedidactiek* (pp. 206–211). Epsilon Uitgaven.

Mooldijk, A., Van der Veen, J., & Snelder, M. (2017). Elektriciteit en magnetisme. In *Handboek Natuurkundedidactiek* (pp. 198–206). Epsilon Uitgaven.

Moya, A. (2018). An Arduino experiment to study free fall at schools. *Physics Education*, *53*(5), 4.

Nelk, J., & Pols, F. (2016). Een leerlingpracticum quantummechanica voor minder dan €30,-. *NVOX*, *41*(3), 132–133.

Next Generation Science Standards. (2017). *Topic Arrangements of the Next Generation Science Standards*. Geraadpleegd van https://www.nextgenscience.org/sites/default/files/AllTopic.pdf

Nussey, J. (2013). *Arduino for dummies*. John Wiley & Sons, Ltd.

Onderwijsraad. (2017). *Doordacht digitaal: Onderwijs in het digitale tijdperk*. Den Haag. Geraadpleegd van https://www.onderwijsraad.nl/upload/documents/publicaties/volledig/Doordacht-digitaal-a.pdf

Petry, C., Pacheco, F., Lohmann, D., Correa, G., & Moura, P. (2016). Project Teaching beyond Physics: Integrating Arduino to the Laboratory. In *Technologies Applied to Electronics Teaching (TAEE)* (pp. 1–6). https://doi.org/10.1109/TAEE.2016.7528376

Platform Onderwijs2032. (2016). *Ons Onderwijs2032; Eindadvies*. Geraadpleegd van https://www.rijksoverheid.nl/binaries/rijksoverheid/documenten/rapporten/2016/01/23/eindadvies-platform-onderwijs2032-ons-onderwijs2032/eindadvies-platform-onderwijs2032-ons-onderwijs2032.pdf

Pols, C. (2017). Technisch ontwerpen in 4V. *NVOX*, *42*(1), 44–45.

Pols, F. (2017a). Arduino op MTech scholen; een eerste kennismaking. *NVOX*, *42*(1), 40–41. Geraadpleegd van https://newsroom.nvon.nl/files/default/pols-F-2017-arduino-op-mtech-scholen_nvox_42-1_40-41.pdf

Pols, F. (2017b). Technisch ontwerpen in 4V. *NVOX2*, *1*(1), 44–45. Geraadpleegd van https://newsroom.nvon.nl/files/default/44-45_NVOX17_XXX_TDS_NR_1.pdf

Pols, F. (2019). *Arduino*. Geraadpleegd van https://deleidseaanpak.nl/uploads/editor/Arduino_module_basis_cursus%2C_Freek_Pols.pdf

Scheltenaar, K., Van der Poel, J., & Bekker, M. (2015). Design-Based Learning in Classrooms Using Playful Digital Toolkits. *In: Chorianopoulos K., Divitini M., Baalsrud Hauge J., Jaccheri L., Malaka R. (Eds) Entertainment Computing - ICEC 2015. ICEC 2015. Lecture Notes in Computer Science*, *9353*, 126–139. https://doi.org/10.1007/978-3-319-24589-8_10

SLO. (2014). *21e eeuwse vaardigheden in het curriculum van het funderend onderwijs*. Geraadpleegd van http://downloads.slo.nl/Repository/21e-eeuwse-vaardigheden-in-het-curriculum-van-het-funderend-onderwijs.pdf

SLO. (2015). *SLO Leerdoelenkaart natuurkunde: gedifferentieerde beheersingsniveaus voor de onderbouw van het voortgezet onderwijs*. Geraadpleegd van https://slo.nl/publish/pages/5075/leerdoelenkaart-natuurkunde-sept--2015.pdf

SLO. (2017). Subdomein G2 Technische automatisering. Geraadpleegd op 10 mei, 2019, geraadpleegd van http://handreikingschoolexamen.slo.nl/natuurkunde/toetsen-in-het-schoolexamen/vakinhoudelijke-(sub)domeinen-havo/subdomein-g2-technische-automatisering

SLO. (2019a). 21e eeuwse vaardigheden. Geraadpleegd op 25 februari, 2019, geraadpleegd van http://curriculumvandetoekomst.slo.nl/21e-eeuwse-vaardigheden

SLO. (2019b). *Leerlijnen voor 21e eeuwse vaardigheden*. Geraadpleegd van http://downloads.slo.nl/Documenten/SLO_%282019%29_21e_eeuwse_vaardigheden_kijkwijzers_en_leerlijnen.pdf

U-Talent. (2018). Lesgeven met Arduino's. Geraadpleegd op 30 mei, 2019, geraadpleegd van https://u-talent.nl/lesgeven-met-arduinos/

Universiteit Twente. (2018). Pro-u cursus Arduino voor docenten. Geraadpleegd op 30 mei, 2019, geraadpleegd van https://www.utwente.nl/evenementen/!/2018/11/266784/pro-u-cursus-arduino

Van der Donk, C., & Van Lanen, B. (2016). *Praktijkonderzoek in de school* (3e ed.). Bussum: Uitgeverij Coutinho.

Van der Veen, T., & Van der Wal, J. (2016). *Van leertheorie naar onderwijspraktijk*.

Van Dijk, B. (2019). Arduino Beginners Lessen. Geraadpleegd op 30 mei, 2019, geraadpleegd van https://arduino-lessen.nl/

Wearable Art Awards Holland. (2017). Makersworld - WAAH. Geraadpleegd op 12 juli, 2019, geraadpleegd van https://waah.nl/makersworld

Wetzels, M., Kooyman, M., Moonen, K., & den Teuling, K. (2018). *Modulehandleiding Arduino en digitale techniek*. Geraadpleegd van http://dev.mennowetzels.nl/ArduinoSintermeerten/files/Modulehandleiding_Arduino.pdf

BIJLAGEN

In deze bijlage staan de volgende documenten:

10.1 Informatiebrief aan ouders van leerlingen
10.2 Vooronderzoek – Interview docent Natuurkunde
10.3 Praktijktest – Arduino met bijbehorende componenten
10.4 Praktijktest – Zelfkennistest
10.5 Praktijktest – Reflectievragen
10.6 Arduino lesmodule – Leerling werkboek
10.7 Arduino lesmodule – Docent handleiding

10.1 INFORMATIEBRIEF AAN OUDERS VAN LEERLINGEN

Informatiebrief

<naam school> - Onderzoek

27-05-2019

Geachte ouder/verzorger,

Als natuurkundedocent in opleiding bij *<naam school>* doe ik vanuit de docentopleiding aan de Universiteit Twente onderzoek naar het verbeteren van onderwijs. Uw zoon of dochter zal in de komende periode als onderdeel van het standaard schoolprogramma bij het vak natuurkunde gaan werken met een zogenaamde 'Arduino' lesmodule. Een Arduino is een kleine microcontroller waarmee leerlingen elektronische toepassingen kunnen maken, zoals het namaken van een verkeerskruispunt met stoplichten. Deze Arduino module is onderdeel van het natuurkunde hoofdstuk Schakelingen; dat voornamelijk gaat over elektriciteit en toepassingen van elektronica.

Voor dit onderzoek heb ik een Arduino lesmodule ontwikkeld op basis van onder andere literatuuronderzoek. Hierbij ligt de nadruk op het verbeteren van de te behalen leerdoelen voor de leerlingen. Ik zal even kort toelichten wat het onderzoek inhoudt: in dit onderzoek wil ik mijn eigen Arduino lesmodule vergelijken met de huidige Arduino lesmodule. Hiervoor wil ik twee klassen vergelijken; waarbij de ene klas mijn module zal uitvoeren, en de andere klas de huidige module. Beide modules zijn in grote lijnen vergelijkbaar om het onderwijsniveau gelijk te houden. Ik kan u echter nu nog niet mededelen welke klas welke module zal gaan uitvoeren. Voor beide klassen zijn naar alle waarschijnlijkheid 4 lessen nodig om de module door te werken. Om achteraf beide modules te kunnen vergelijken wil ik voorafgaand aan beide modules een korte zelfkennistest van enkele vragen afnemen. Tijdens de lessen wil ik bij beide modules de leerlingen observeren. Afsluitend wil ik opnieuw bij beide klassen een korte zelfkennistest afnemen, tezamen met een vragenlijst over hoe de leerlingen hun module hebben ervaren. In al deze situaties zal de data anoniem worden verzameld. Leerlingen zullen voor deze onderdelen dan ook geen cijfer ontvangen. Tevens zal dit onderzoek plaatsvinden onder begeleiding van de natuurkunde docent van uw zoon of dochter.

Ik stuur u deze informatiebrief omdat uw zoon of dochter minderjarig is, zodat u indien gewenst bezwaar kunt aantekenen tegen dit onderzoek. Dit kunt u eenvoudig kenbaar maken door deze brief ondertekend aan uw zoon of dochter mee terug te geven. Mocht u geen bezwaar hebben, dan hoeft u niks te doen. In beide gevallen zal ik na een week niets ontvangen te hebben er rechtswege vanuit gaan dat u geen bezwaar hebt.

Mocht u na het lezen van deze brief nog vragen hebben dan kunt u altijd contact met mij opnemen.

Alvast heel hartelijk bedankt voor uw medewerking,

Stefan Lentelink
<naam school> - docent in opleiding
E-mail: s.lentelink@<naam school>.nl

Indien u bezwaar hebt kunt u onderstaand hokje aanvinken en de bijbehorende gegevens invullen.

□ Ik teken bezwaar aan voor:

Naam leerling: _____

Handtekening ouder/verzorger: _____

10.2 VOORONDERZOEK – INTERVIEW DOCENT NATUURKUNDE

Vooronderzoek

- 21e-eeuwse vaardigheden
    - Welke vaardigheden vind je belangrijk om te behandelen?
    - Welke vaardigheden passen bij een Arduino lesmodule?
    - Welke vaardigheden heeft de klas al?

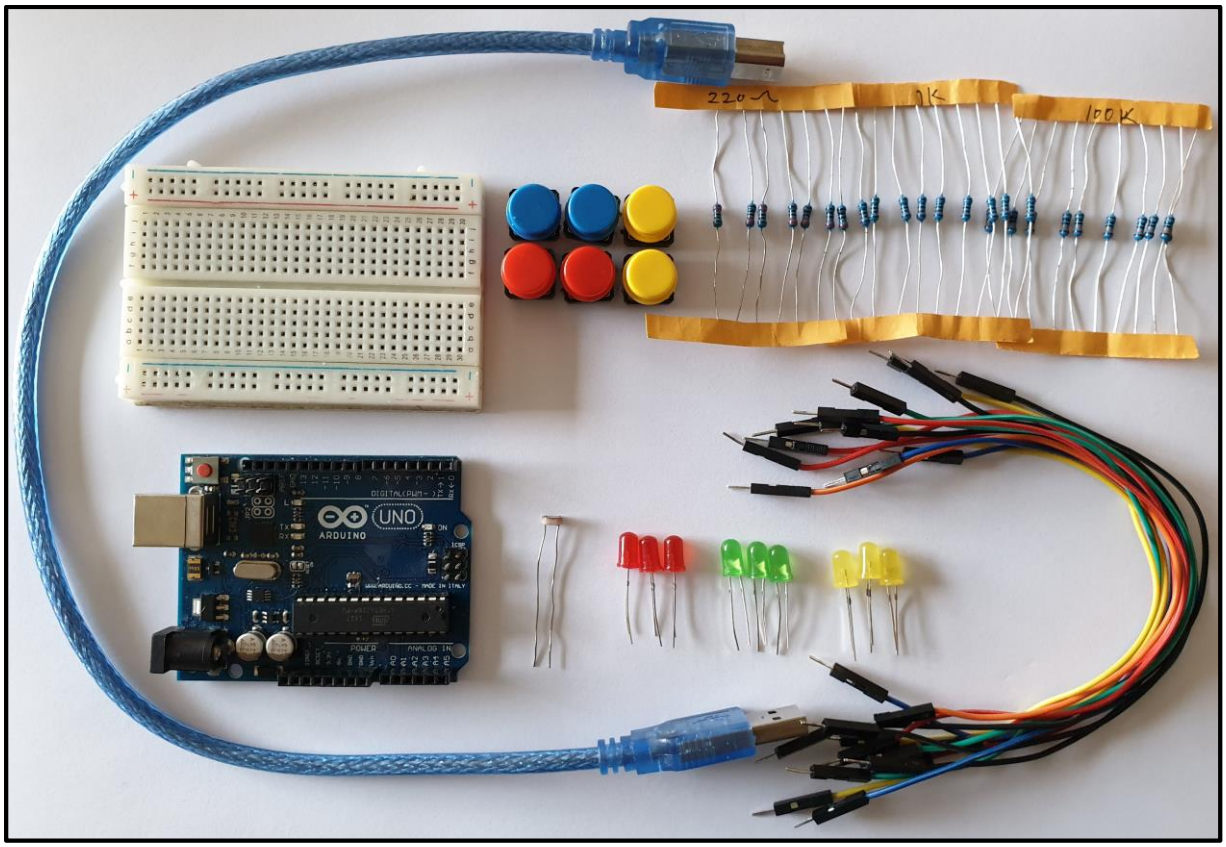
- Elektriciteit
    - Welke methode wordt gebruikt?
    - Welke onderwerpen worden behandeld?
    - Welke onderwerpen passen bij een Arduino lesmodule?

- Technische automatisering
    - Welke methode wordt gebruikt?
    - Welke onderwerpen worden behandeld?
    - Welke onderwerpen passen bij een Arduino lesmodule?

- Hoe ziet je huidige opzet van de Arduino lesmodule eruit?
    - Hoeveel lessen besteed je hieraan?
    - Deel je leerlingen in groepjes in?
    - Welke onderwerpen behandel je?
    - Wat zijn de leerdoelen?
    - Hoe ziet de opbouw eruit?
    - Waarom kies je voor Arduino?
    - Welke problemen ben je in het verleden tegengekomen met de Arduino lesmodule, en hoe heb je deze opgelost?

- Heb je nog algemene tips voor de opzet van een Arduino lesmodule?

10.3 PRAKTIJKTEST – ARDUINO MET BIJBEHORENDE COMPONENTEN



10.4 PRAKTIJKTEST – ZELFKENNISTEST

Alvast bedankt voor het invullen!

Deze vragenlijst is anoniem.

1. Een Arduino is een heel kleine computer waarmee je met behulp van elektronische componenten zoals kabeltjes, LEDs, weerstandjes, LDR's en drukknopjes objecten kunt maken die reageren op hun omgeving. Een voorbeeld is een buitenlamp die aangaat als het buiten donker wordt. Hiervoor gebruik je ook computer-software om de Arduino mee te programmeren.

   Geef aan wat je kennisniveau is bij het werken met dit soort hardware en software.

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

   Leg uit waarom: _____

   _____

   _____

2. Noem de drie basisonderdelen waaruit een geautomatiseerd systeem bestaat. Noem daarbij per basisonderdeel ook de component die hiervoor wordt gebruikt om deze taak uit te voeren.

   1. _____

   2. _____

   3. _____

3. Noem een voorbeeld van een meetsysteem, stuursysteem, en regelsysteem.

   Meetsysteem: _____

   Stuursysteem: _____

   Regelsysteem: _____

4. Geef aan wat je verwachte kennisniveau is van de volgende onderwerpen:

   Wet van Ohm

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

   Soortelijke weerstand

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

   Serieschakelingen

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

   Parallelschakelingen

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

   Tekenen van elektrische schakelingen

   zeer laag ☐ ☐ ☐ ☐ *zeer hoog*

5. In onze maatschappij worden digitalisering en technologische ontwikkelingen steeds belangrijker. Het is daarom belangrijk om voorbereid te worden op deze toekomst door het aanleren van bepaalde kennis en vaardigheden. Een aantal voorbeelden worden hieronder gegeven.

   Geef aan hoe belangrijk je onderstaande kennis en vaardigheden vindt:

   Probleemoplossen: een probleem kunnen herkennen en planmatig oplossen.

   zeer onbelangrijk ☐ ☐ ☐ ☐ *zeer belangrijk*

   ICT-basisvaardigheden: kennis en vaardigheden ontwikkelen over computer hardware/software

   zeer onbelangrijk ☐ ☐ ☐ ☐ *zeer belangrijk*

   Computational thinking: problemen zo kunnen formuleren dat ze op te lossen zijn m.b.v. ICT.

   zeer onbelangrijk ☐ ☐ ☐ ☐ *zeer belangrijk*

   Samenwerken: gezamenlijk een doel kunnen behalen en anderen daarbij aanvullen/ondersteunen.

   zeer onbelangrijk ☐ ☐ ☐ ☐ *zeer belangrijk*

   Herken je dat bovenstaande kennis en vaardigheden op school aandacht krijgen?
   Leg uit waarom wel/niet en geef voorbeelden.

   Probleemoplossen: ☐ Ja ☐ Nee ☐ Weet ik niet

   _____
   _____
   _____
   _____

   ICT-basisvaardigheden: ☐ Ja ☐ Nee ☐ Weet ik niet

   _____
   _____
   _____
   _____

   Computational thinking: ☐ Ja ☐ Nee ☐ Weet ik niet

   _____
   _____
   _____
   _____

   Samenwerken: ☐ Ja ☐ Nee ☐ Weet ik niet

   _____
   _____
   _____
   _____

10.5 PRAKTIJKTEST – REFLECTIEVRAGEN

Alvast bedankt voor het invullen!

Deze vragenlijst is anoniem.

Ik vond het leerzaam om met Arduino te (leren) werken.

helemaal oneens ☐ ☐ ☐ ☐ *helemaal eens*

Ik vond Arduino een goede toevoeging aan het hoofdstuk Schakelingen | Automatisch regelen.

helemaal oneens ☐ ☐ ☐ ☐ *helemaal eens*

Ik vond Arduino een goede manier om te werken aan mijn kennis en vaardigheden voor de toekomst.

helemaal oneens ☐ ☐ ☐ ☐ *helemaal eens*

Tops

Wat vond je goed/ leuk aan deze lesmodule?

_____
_____
_____
_____
_____
_____
_____
_____
_____

Tips

Wat vond je minder goed/leuk of kan er verbeterd worden aan deze lesmodule?

_____
_____
_____
_____
_____
_____
_____
_____
_____

10.6 ARDUINO LESMODULE – LEERLING WERKBOEK

Arduino lesmodule

INHOUDSOPGAVE

1 Kennismaken met Arduino 4

1.1 Arduino 4

1.2 Breadboard 6

1.3 Componenten 6

1.3.1 Jumper 6

1.3.2 LED 7

1.3.3 Weerstand 7

1.3.4 LDR 7

1.3.5 Drukknop 7

1.4 Software 8

1.4.1 Arduino sketch 8

1.4.2 Void setup() & void loop() 9

1.4.3 Verifieer & Upload 9

2 Werken met Arduino 10

2.1 Knipperende LED op de Arduino zelf 10

2.2 Een LED aansluiten op de Arduino 12

2.3 Een drukknop op de Arduino 16

2.4 Weerstanden op de Arduino – Serieschakeling 18

2.5 Weerstanden op de Arduino – Parallelschakeling 20

2.6 Een LDR op de Arduino 22

2.7 LEDs op de Arduino – Serieschakeling 24

2.8 LEDs op de Arduino – Parallelschakeling 26

2.9 LEDs afwisselen op de Arduino 28

3 Projecten met Arduino 30

3.1 Voetgangersoversteekplaats – Beginner 30

3.2 Voetgangersoversteekplaats – Expert 30

INLEIDING

We leven in een snel veranderende maatschappij waar digitalisering en technologische ontwikkelingen een belangrijke rol spelen. Als je alleen al om je heen kijkt zie je hoe afhankelijk we zijn van technologie. Van de bewegingssensor in het toilet voor het licht, tot aan het betalen met de pinpas bij de supermarkt voor een zak Apekoppen. Het is daarom belangrijk om voorbereid te worden op deze toekomst door het aanleren van vaardigheden die ook wel bekend staan als 21e-eeuwse vaardigheden. De Arduino is hiervoor heel geschikt, en je zal gaan zien waarom. Het werken met Arduino vereist wel een klein beetje kennis van elektronica en programmeren, maar hierbij zullen we je helpen. Arduino sluit ook heel goed aan bij de hoofdstukken *Elektriciteit* en *Technische Automatisering* uit je lesboek. Deze hoofdstukken zullen we dan ook veel gaan gebruiken voor de opdrachten in deze lesmodule.

➢ De leerdoelen die we willen bereiken in deze lesmodule zijn als volgt:

Arduino

- Je kunt onderstaande Arduino hardware en software beschrijven en hiermee werken:
  - Arduino, inclusief voorbeelden van Arduino projecten
  - Microcontroller, inclusief enkele voordelen en voorbeelden in het dagelijks leven
  - Breadboard
  - Componenten: jumper, LED, weerstand, LDR, en drukknop
  - Arduino IDE, inclusief Arduino sketch, void setup(), void loop(), verifieer, en upload

Elektriciteit | Technische automatisering

- Je kunt de drie basisdelen van een geautomatiseerd systeem benoemen, inclusief de hiervoor gebruikte componenten om die taken uit te voeren
- Je kunt voorbeelden van een meet-, stuur-, en regelsysteem noemen met Arduino projecten
- Je kunt rekenen met spanning, stroomsterkte, en weerstand met behulp van:
  - Wet van Ohm
  - Soortelijke weerstand
  - Serieschakelingen
  - Parallelschakelingen
- Je kunt elektrische schakelingen tekenen, inclusief serie- en parallelschakelingen

21e-eeuwse vaardigheden

- Je kunt een probleem herkennen en planmatig oplossen
- Je kunt problemen zo formuleren dat ze op te lossen zijn met behulp van ICT
- Je ontwikkelt kennis en vaardigheden omtrent computer hardware en software
- Je kunt gezamenlijk een doel behalen en anderen daarbij aanvullen en ondersteunen

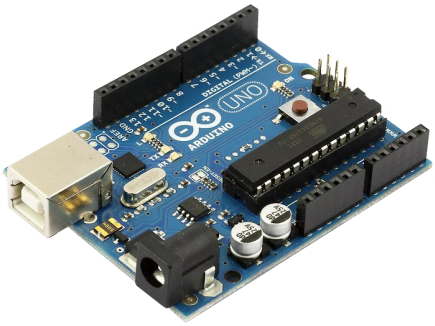
1 KENNISMAKEN MET ARDUINO

Je gaat nu kennismaken met Arduino. Je gaat bekijken wat een Arduino precies is, en hoe je hier mee kunt werken. Na dit onderdeel kun je aan de slag met het maken van je eigen Arduino programma's!

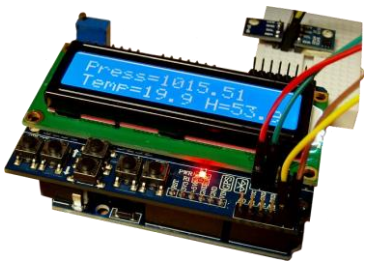
Pak alvast een Arduino-setje zodat je tijdens het lezen alles ook meteen kunt bekijken.

1.1 ARDUINO

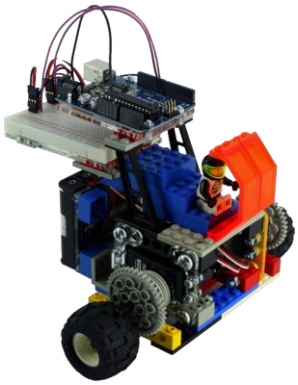
Een *Arduino*, figuur 1, kun je zien als een soort van kleine computer voor iedereen met interesse in het ontwerpen en maken van slimme en creatieve objecten die reageren op hun omgeving. Zo kun je bijvoorbeeld schakelaars, lichtsensoren, bewegingsmeters, afstandsmeters, en temperatuursensoren gebruiken om motoren, lampjes, pompjes en beeldschermen aan te sturen. Arduino projecten gaan van eenvoudige temperatuurmeters, figuur 2, tot aan compleet zelfrijdende robots, figuur 3.



Figuur 1 – Arduino

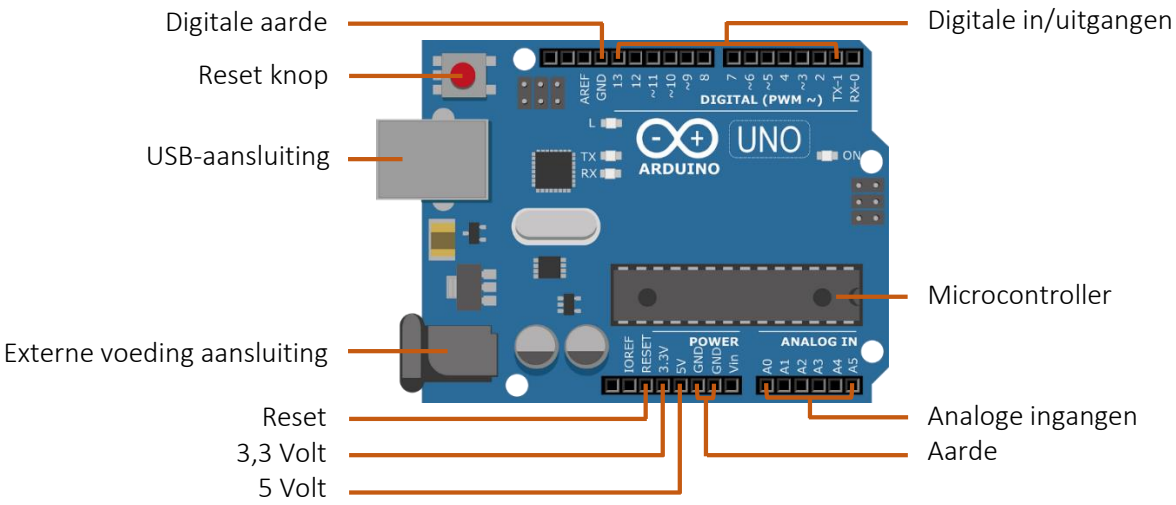


Figuur 2 – Arduino temperatuurmeter



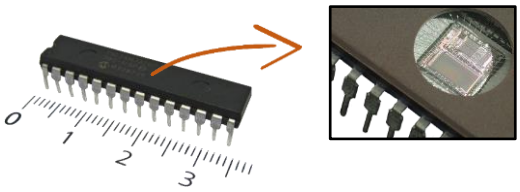
Figuur 3 – Arduino robot

De Arduino zelf is opgebouwd uit een microcontroller, ondersteunende componenten, en een aantal aansluitingen, figuur 4. De *microcontroller* is de computer. Deze is erg klein, figuur 5, en wordt met Arduino voor gebruikers eenvoudig toegankelijk gemaakt. De metalen pootjes van de microcontroller zijn op de Arduino namelijk verbonden met de langwerpige zwarte blokjes met gaatjes boven- en onderaan de Arduino. Op deze manier kun je snel en eenvoudig iets aansluiten en uitproberen.



Figuur 4 – Arduino schematisch weergegeven

Microcontrollers vind je in bijna alle moderne apparaten. Denk aan een televisie, robotstofzuiger, elektrische tandenborstel, of zelfs een zingende wenskaart. Ze worden gebruikt om al deze dingen automatisch aan te sturen, en zijn heel goed in het uitvoeren van één taak.



Figuur 5 – Microcontroller

In het geval van de televisie werkt dit als volgt: stel je selecteert een zender. De microcontroller in de televisie ontvangt het signaal van de afstandsbediening, verwerkt deze, en laat vervolgens de gekozen zender zien op het scherm. Deze drie stappen zijn simpelweg de basis van een geautomatiseerd systeem. Het stroomdiagram per systeemdeel is als volgt:

Tussen deze blokken worden signalen doorgegeven via kleine elektrische spanningen. Indien er geen signaal is, is de spanning 0V. Dit noemen we laag of 0. Bij een maximale spanning, 5 V in het geval van Arduino (ter vergelijking: een AA-batterij geeft 1.5 V), noemen we het signaal hoog of 1.

Opgave 1. Noem nog 3 apparaten binnenshuis waarin zich microcontrollers bevinden.

1. _____
2. _____
3. _____

Opgave 2. Noem 3 voordelen van het gebruik van microcontrollers.

1. _____
2. _____
3. _____

Opgave 3. Noem per systeemdeel uit het stroomdiagram in algemene zin de component die wordt gebruikt om deze taken uit te voeren (gebruik hiervoor je lesboek).

- Invoer: _____
- Verwerking: _____
- Uitvoer: _____

Opgave 4. Beschrijf aan de hand van invoer, verwerking, en uitvoer de werking van de afstands-bediening bij het selecteren van een andere televisiezender.

_____

_____

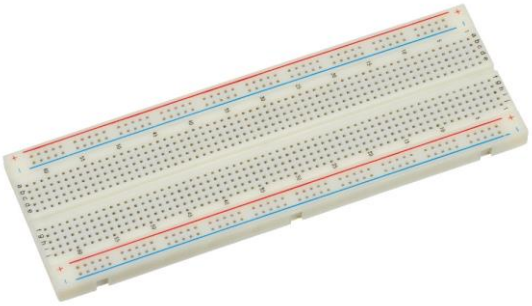
Opgave 5. Met Arduino kun je ontelbare objecten creëren. Noem voor elk van de onderstaande systemen (zie je lesboek) een object die je hiervoor met Arduino zou kunnen maken.

- Meetsysteem: _____
- Regelsysteem: _____
- Stuursysteem: _____

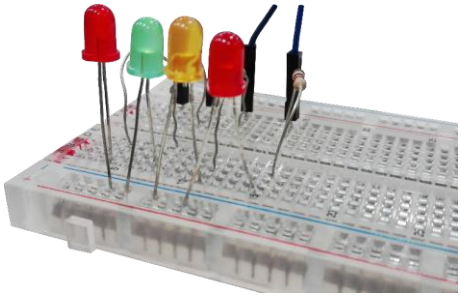
1.2 BREADBOARD

Om met de Arduino elektrische schakelingen te kunnen bouwen wordt het zogenaamde *breadboard* (letterlijk: broodplank) gebruikt, figuur 6. Het breadboard is niets meer dan een aantal rijen met gaatjes, genaamd *pinnen*, waarin je verschillende elektrische componenten kunt vastprikken. Denk bijvoorbeeld aan kabeltjes, LEDs, en weerstandjes, figuur 7. Het breadboard sluit je met kabeltjes aan op de Arduino via de pinnen (gaatjes) in de langwerpige zwarte blokjes boven- en onderaan de Arduino, figuur 4.

Binnenin het breadboard zitten metalen stripjes die de rijen met pinnen met elkaar verbinden. Dit ziet er schematisch uit zoals is weergegeven in figuur 8.

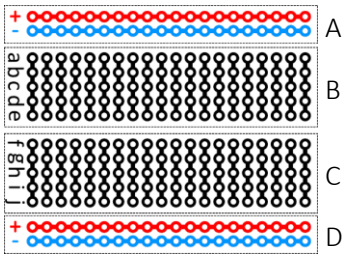


Figuur 6 – Breadboard



Figuur 7 – Breadboard met LEDs

De rijen A en D op het breadboard zijn voor de stroomvoorziening. Hierbij is de (+) rood en de (-) zwart of blauw. De (+) sluit je meestal aan op de 5 V uitgang van de Arduino. De (-) sluit je aan op de GND van de Arduino, wat staat voor ground (Engels voor aarde), ofwel 0 V. Zoals figuur 8 laat zien zijn de pinnen bij A en D per rij verbonden. Sluit je op één pin in deze rij iets aan dan heeft dat invloed op de gehele rij.



Figuur 8 – Breadboard schematisch weergegeven

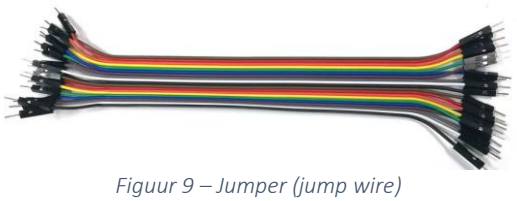
De kolommen B en C worden gebruikt voor de componenten, zoals LEDs. Zoals figuur 8 laat zien zijn de pinnen bij B en C per kolom verbonden. Sluit je op één pin in deze kolom iets aan dan heeft dat invloed op de gehele kolom. De kolommen van B en C zijn niet met elkaar verbonden.

1.3 COMPONENTEN

Er zijn veel componenten te gebruiken voor de Arduino. In deze lesmodule zullen wij de *jumper*, *LED*, *weerstand*, *LDR*, en *drukknop* beschrijven en gebruiken.

1.3.1 Jumper

Een *jumper* (officieel *jump wire*), figuur 9, is simpelweg een elektrische draad (kabeltje) die aan beide uiteinden een pinnetje heeft. Deze gebruik je om de Arduino te verbinden met het breadboard en hierop componenten met elkaar te verbinden, zoals LEDs en weerstanden.



Figuur 9 – Jumper (jump wire)

Er is geen verschil tussen de kleuren van een jumper. Je gebruikt deze enkel voor jezelf om onderscheid te maken in je opstelling, zoals rood voor de (+) aansluiting en zwart voor de (-) aansluiting.

1.3.2 LED

Een *LED (light-emitting diode)*, figuur 10, zendt licht uit als er een elektrisch stroom door de LED loopt. De LED laat stroom maar door in één richting, vandaar de term diode. De lange poot van de LED is de (+) aansluiting en de korte poot de (-) aansluiting.
Let op: sluit je de LED verkeerd om aan, dan doet de LED het niet!

Figuur 10 – LED

LEDs zijn heel bijzonder en niet te vergelijken met bijvoorbeeld een gloeilampje. Een gloeilampje is in principe een stabiele weerstand die door de lopende stroom verhit wordt en begint te gloeien. Een LED heeft deze eigenschap niet. Een LED werkt met een zogenaamde doorlaatspanning. Pas als deze spanning wordt bereikt gaat er een stroom lopen en zal de LED gaan branden, eerder niet. Opvallend is dat voorbij deze doorlaat-spanning de LED (bijna) geen weerstand meer heeft. Je kan de LED dan vergelijken met een zeer goed geleidende stroomdraad. Via de wet van Ohm zie je dan ook dat vanaf deze grens de stroomsterkte in heel korte tijd ontzettend groot wordt.

Figuur 11 – LED doorlaatspanning

De stroomsterkte door een LED is idealiter 20 mA. Bij een hogere stroomsterkte kan de LED al snel kapot gaan. De LED zal daarom altijd in serie geschakeld moeten worden met een weerstand die de stroomsterkte laag houdt. Welke weerstand je hiervoor moet kiezen bereken je via de wet van Ohm. Hierbij geldt voor de spanning over de weerstand: $U_R = U_{totaal} - U_{LED}$. De doorlaatspanning verschilt per LED: rood = 1.9 V; geel = 2.0 V; groen = 2.1 V.

1.3.3 Weerstand

Een *weerstand*, figuur 12, wordt gebruikt om de doorgang van een elektrische stroom te belemmeren. Dit is handig omdat voor sommige componenten de stroomsterkte anders te veel zou zijn, zoals voor de LED. De waarde van een weerstand wordt op de weerstand zelf aangegeven met kleurcodes. Elke kleur staat voor een getal.

Figuur 12 – Weerstand 220 Ω

Zwart	0
Bruin	1
Rood	2
Oranje	3
Geel	4
Groen	5
Blauw	6
Violet	7
Grijs	8
Wit	9

- Ring 1, 2 en 3 geven het 1e, 2e en 3e getal van de weerstand.
- Ring 4 geeft aan met welke macht van 10 het getal van ring 1, 2 en 3 vermenigvuldigd moeten worden.
- Ring 5 geeft het foutpercentage aan, ofwel hoeveel procent de werkelijke waarde af kan wijken van de opgegeven waarde.

Je begint met aflezen vaak bij de ring het dichtst bij de rand, of wat logisch is.
Voorbeeld: rood, rood, zwart, zwart, bruin = 220 · 10 [1] Ω met 1% foutmarge.

1.3.4 LDR

Een *LDR (light-dependent resistor)*, figuur 13, is een weerstand die beïnvloed wordt door de hoeveelheid licht die erop valt. Hoe groter de lichtsterkte, hoe lager de weerstandswaarde. De LDR kan hierdoor goed als sensor dienen in schakelingen.

Figuur 13 – LDR

1.3.5 Drukknop

Een *drukknop*, figuur 14, wordt gebruikt om componenten zoals een LED aan te sturen. Wanneer er op de knop wordt gedrukt, worden de naast elkaar gelegen pootjes met elkaar verbonden (de zichtbare pootjes in figuur 14). De pootjes tegenover elkaar zijn altijd met elkaar verbonden, ongeacht of de knop is ingedrukt.

Figuur 14 – Drukknop

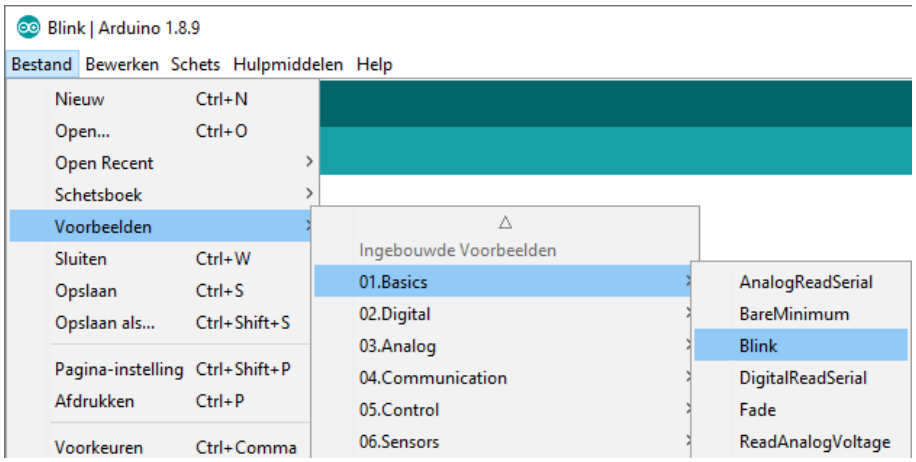
1.4 SOFTWARE

Om de Arduino een opdracht te kunnen laten uitvoeren moet je hem wel eerst vertellen wat hij precies moet doen. Dit doe je door een programma te schrijven waarin alle instructies voor de Arduino staan. Om je hierbij te helpen heeft Arduino speciale software waarmee je gemakkelijk je eigen programma kan schrijven. Ook kan deze software jouw programma meteen op de Arduino zetten. Deze software, figuur 15, heet de *Arduino IDE (Integrated Development Environment)*.

Figuur 15 – Arduino IDE met hierin geopend een lege sketch

1.4.1 Arduino sketch

Een programma die je maakt in de Arduino IDE heet een *sketch*. In figuur 15 staat bijvoorbeeld een lege sketch geopend. In de Arduino IDE zijn veel voorbeeld sketches al voor je gemaakt, zodat je deze niet zelf hoeft te schrijven. Deze sketches, zoals 'Blink' waarmee je een LED kunt laten knipperen, kun je vinden onder Voorbeelden, figuur 16.



Figuur 16 – Arduino IDE sketch voorbeelden

1.4.2 Void setup() & void loop()

De Arduino sketch bestaat altijd uit minstens twee elementen: *void setup()* en *void loop()*, figuur 15. In void setup() staat alles wat van belang is voordat de Arduino een taak gaat uitvoeren, zoals welke aansluiting (pin/gaatje) de Arduino moet gebruiken en wat hij hiermee moet doen. In void loop() staat de opdracht die de Arduino daadwerkelijk moet gaan uitvoeren, zoals het laten knipperen van een LED.

De Arduino zal bij het uitvoeren van een sketch void setup() slechts eenmalig uitvoeren. Void loop() wordt echter, zoals de naam al aangeeft, keer op keer herhaald. Hierdoor kan de Arduino heel goed één specifieke taak uitvoeren. Hoe je void setup() en void loop() moet invullen leer je in de opdrachten.

Standaard termen, zoals void setup() en void loop(), krijgen in de Arduino IDE altijd een specifieke kleur. *Voor meer informatie over deze termen kun je kijken op: https://www.arduino.cc/reference/en/.*

1.4.3 Verifieer & Upload

Als de Arduino sketch klaar is kan deze in principe naar de Arduino worden verzonden. Om ervoor te zorgen dat er geen fouten in de sketch staan laat je deze altijd eerst door de Arduino IDE controleren. Dit gaat via de knop *Verifieer* linksboven in het scherm, figuur 15. Als er geen fouten in zitten geeft de Arduino IDE dit linksonder in het scherm weer met de opmerking *Compileren voltooid*. Als er wel fouten zijn krijg je hierover specifieke informatie te zien die je helpt de fout op te lossen.

Vervolgens kun je de sketch naar de Arduino sturen via de knop *Upload* linksboven in het scherm, figuur 15. Je hoeft de sketch niet eerst nog op te slaan, dat gaat automatisch. Als alles met het uploaden goed is gegaan geeft de Arduino IDE dit linksonder in het scherm weer met de opmerking *Uploaden voltooid*. De Arduino zal nu de sketch gaan uitvoeren.

Let op: *zowel de Arduino IDE, als de Arduino zelf, kunnen nog een sketch van de vorige keer bevatten. De Arduino IDE en Arduino onthouden namelijk de sketch die voor het laatst opgeslagen is. Dit kan dus ook de sketch zijn die niet door jou is gemaakt, maar door een klasgenoot. Open daarom altijd eerst opnieuw jouw eigen sketch en upload deze voor de zekerheid naar de Arduino.*

2 WERKEN MET ARDUINO

Je gaat nu werken met Arduino. Je gaat beginnen met een eenvoudige knipperende LED op de Arduino zelf, waarna je steeds een stapje verder gaat. Na dit onderdeel kun je aan de slag met je eerste project.

2.1 KNIPPERENDE LED OP DE ARDUINO ZELF

Op de Arduino zelf zitten ook LEDs. Deze worden gebruikt om te laten zien dat de Arduino aan staat (ON), data ontvangt (RX), of data verstuurt (TX). Er is echter nog een vierde LED welke je zelf kunt aansturen. In dit onderdeel ga je deze LED laten knipperen door de Arduino te programmeren.

Opgave 6. Zoek de zelf aan te sturen LED op de Arduino en schrijf op welke letter hierbij staat.

_____

➢ Je gaat nu de LED op de Arduino laten knipperen met behulp van een kant-en-klare sketch.

Opgave 7. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.
b. Open de Arduino IDE via de snelkoppeling op het bureaublad.
c. Open de voorbeeld-sketch genaamd *Blink.*
   Bestand → *Voorbeelden* → *01.Basics* → *Blink* (zie figuur 16)
   Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.
d. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.
e. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.
f. Klaar! De LED op de Arduino zelf zal nu gaan knipperen.

Je hebt zojuist je eerste programma op de Arduino gezet en de Arduino een opdracht laten uitvoeren!

Voordat we verder gaan willen we je eerst een aantal *algemene regels* vertellen waar je bij het schrijven van programma's rekening mee moet houden:

ALGEMENE REGELS

- Computers lezen een programma stapsgewijs van boven naar beneden, regel voor regel.
- De programmeertaal van Arduino is hoofdlettergevoelig, ofwel: *Arduino* is anders dan *arduino*.
- Functies, zoals *void setup()* en *void loop()*, gebruiken accolades { } om aan te geven wat bij de functie hoort. Ieder open haakje moet altijd worden gesloten (net zoals in je rekenmachine).
- Elke regel die compleet is moet afgesloten worden met een puntkomma ; In de praktijk is dit meestal iedere regel. Uitzonderingen zijn het commentaar // en afsluiten van een functie }.

➢ Je gaat nu stapsgewijs de opzet van de sketch 'Blink' bekijken, figuur 17.

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Figuur 17 – Arduino IDE sketch 'Blink'

//	Dit is de commentaarregel. Alles wat hierachter staat negeert de computer.
void setup()	Deze functie wordt door de Arduino slechts eenmalig uitgevoerd. Hierin staat alles wat van belang is voordat de Arduino een taak gaat uitvoeren, zoals *pinMode*.
pinMode	Hiermee vertel je aan de Arduino welke pin (aansluiting) hij moet gebruiken en wat hij hiermee moet doen. In dit geval wordt Arduino vertelt dat hij de ingebouwde LED met de naam *LED_BUILTIN* moet gebruiken als een uitgang (*OUTPUT*). Bij *OUTPUT* levert de Arduino spanning. Bij *INPUT* ontvangt de Arduino spanning, zoals van een sensor. *Let op: de ingebouwde LED, hier LED_BUILTIN genoemd, is altijd gekoppeld aan pin 13.*
void loop()	Deze functie wordt door de Arduino herhaaldelijk uitgevoerd. Hierin staat de opdracht die de Arduino daadwerkelijk moet uitvoeren, zoals *digitalWrite*.
digitalWrite	Hiermee kun je op een digitale pin, zoals pin 13, een hoge of lage spanning zetten. Hiervoor moet je wel aangeven welke pin je kiest, en of deze een hoge spanning (5 V) of lage spanning (0 V) moet krijgen. De eerste regel hier geeft dus aan dat *LED_BUILTIN* (ofwel: pin 13) een hoge spanning (*HIGH*) moet krijgen. Hierdoor zal de LED branden. *Let op: de genoemde pin moet wel eerst benoemd zijn in void setup() om te werken.*
delay	Hiermee kun je de computer even laten wachten voordat hij de volgende regel uitvoert. Op deze manier kun je dus een knipperende LED maken: LED aan, 1 s wachten, LED uit, 1 s wachten. Deze cyclus, ofwel het deel tussen { }, herhaalt zich steeds weer opnieuw. *Let op: de ingevulde waarden in de functie delay zijn altijd in milliseconde.*

Opgave 8. Verander de sketch zodanig dat de LED op de Arduino steeds 0,10 seconde aan is en dan weer 0,25 seconde uit blijft. Vul hieronder in wat je in de sketch hebt ingevuld.

```
void loop()  {
   digitalWrite(LED_BUILTIN, _____________);
   delay(_______);
   digitalWrite(LED_BUILTIN, _____________);
   delay(_______);
}
```

2.2 EEN LED AANSLUITEN OP DE ARDUINO

Het is natuurlijk veel leuker om zelf een LED aan te sluiten op de Arduino. Hiervoor kun je de Arduino gebruiken als spanningsbron, en een stroomkring maken waarin je de LED opneemt. Echter heb je al gelezen in §1.3.2 dat hiervoor ook een weerstand nodig is. Deze noem je een voorschakelweerstand, omdat deze weerstand in de stroomkring altijd in serie vóór de LED moet worden geschakeld.

➢ Je gaat nu ter *bewustwording* eerst kijken naar een situatie *zonder* voorschakelweerstand.

Opgave 9. De weerstand van een LED zou je *voorbij* de doorlaatspanning kunnen *benaderen* alsof het een stroomdraad is, §1.3.2. Hiervoor kun je de soortelijke weerstand gebruiken. Bereken de weerstand van de LED uitgaande dat deze als stroomdraad 2 jumpers lang is (20 cm per jumper), een diameter heeft van 0,5 mm, en gemaakt is van ijzer.

Opgave 10. De Arduino levert als spanningsbron een spanning van 5.0 V. De weerstandswaarde van de LED heb je bij opgave 9 berekend. Bereken nu de stroomsterkte door de LED.

Bedenk je nu dat een LED al kapot kan gaan als er meer dan 20 mA (= 0,020 A) doorgaat! Om deze reden moet er dus altijd een voorschakelweerstand in serie worden gebruikt.

Opmerking: in de praktijk zal de Arduino zo'n hoge stroomsterkte echter nooit behalen.

➢ Je gaat nu de *voorbereidingen* treffen om de LED aan te kunnen sluiten op de Arduino.

Opgave 11. Teken het schakelschema voor een stroomkring met de LED en voorschakelweerstand in serie aangesloten op een spanningsbron.

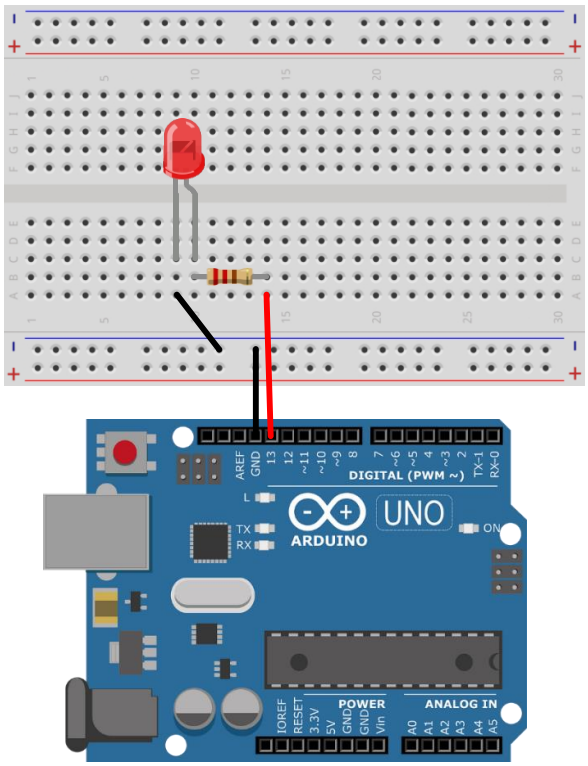
In de stroomkring moet de voorschakelweerstand voldoende groot zijn zodat de stroomsterkte door de LED kleiner is dan 20 mA, §1.3.2. Hiervoor gebruik je de wet van Ohm en regels voor serieschakeling.

Opgave 12. De Arduino levert als spanningsbron een spanning van 5.0 V. Op een rode LED staat 1,9 V / 20 mA (zie voor een toelichting §1.3.2). Bereken de waarde van de voorschakelweerstand die nodig is om de rode LED op de juiste spanning te laten branden.
Let op: vergeet niet de doorlaatpsanning mee te nemen in je berekening, zie §1.3.2.

➢ Je gaat nu de LED aansluiten op de Arduino.

Opgave 13. Zoek in je Arduino setje een weerstand, §1.3.3, die zo dicht mogelijk bij de berekende waarde uit opgave 12 ligt. Let op: de weerstand mag in ieder geval niet kleiner zijn.

Opgave 14. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 15. Voer de volgende stappen uit.
Let op: dit hoeft niet als de sketch 'Blink' uit opgave 7 nog op de Arduino staat.

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.
b. Open de Arduino IDE via de snelkoppeling op het bureaublad.
c. Open de voorbeeld-sketch genaamd *Blink*.
   Bestand → Voorbeelden → 01.Basics → Blink
   Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.
d. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.
e. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.
f. Klaar! De LED zal nu gaan knipperen.

Opgave 16. Verander de sketch zodanig dat de LED niet meer knippert maar continu brandt. Controleer of het werkt en vul hieronder in wat je in de sketch hebt ingevuld.

```
void loop()  {









}
```

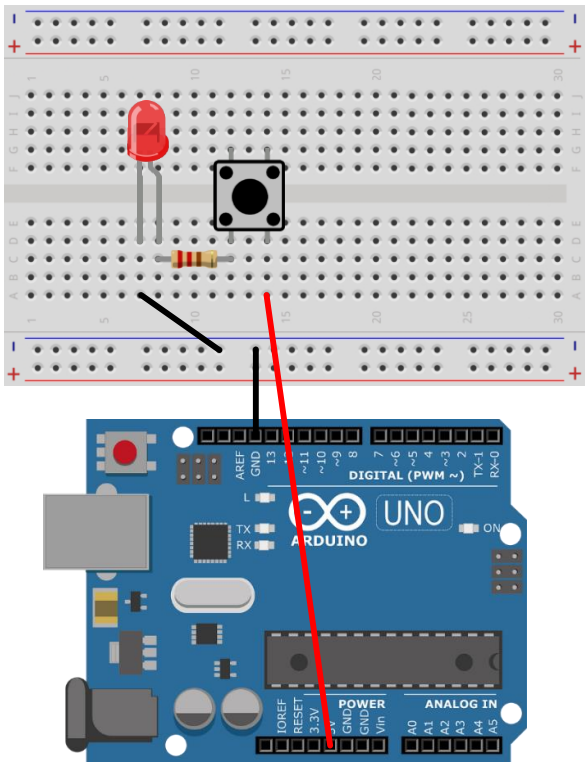
2.3 EEN DRUKKNOP OP DE ARDUINO

Een *drukknop*, §1.3.5, wordt gebruikt om componenten zoals een LED aan te sturen. Wanneer er op de knop wordt gedrukt worden de naast elkaar gelegen pootjes met elkaar verbonden. De pootjes tegenover elkaar zijn altijd met elkaar verbonden, ongeacht of de knop is ingedrukt.

➢ Je gaat nu met een drukknop een LED laten branden.

Opgave 17. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.
Opmerking: zorg ervoor dat de drukknop stevig op het breadboard zit vastgedrukt.

Voor het leren werken met Arduino gebruik je nu in plaats van pin 13 de standaard 5 V pin op de Arduino. Deze aansluiting levert continu 5 V. Dit is ook niet te veranderen, in tegenstelling tot bijvoorbeeld pin 13 (denk aan de knipperende LED). De aansluiting met 5 V is echter zeer handig om een component continu te voorzien van spanning, of tijdelijk in het geval van de drukknop. Je hoeft hiervoor niets te programmeren.



Opgave 18. Wat neem je waar? Geef een toelichting.

_____

_____

_____

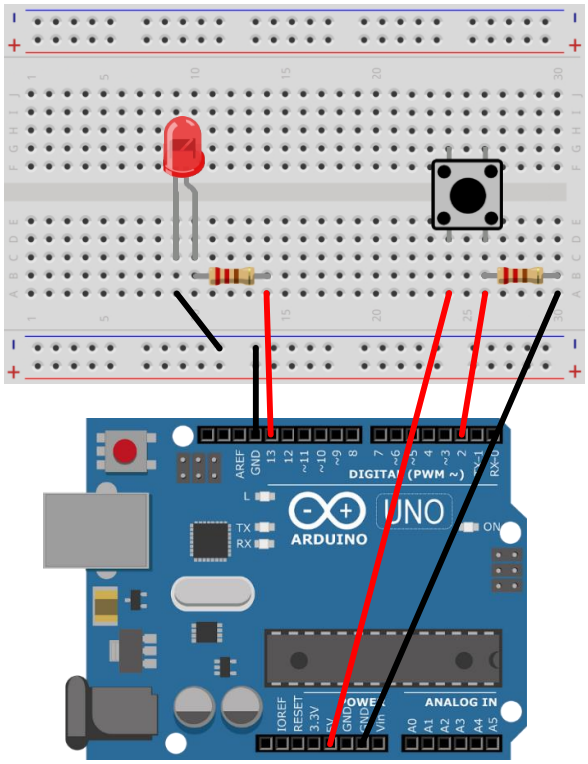
_____

_____

Het is ook mogelijk om de Arduino te laten registreren dat de drukknop is ingedrukt. Dit is handig als je de LED continu wil laten branden zonder de drukknop ingedrukt te moeten houden.

➢ Je gaat nu met een drukknop een LED continu laten branden nadat je er eenmalig op hebt gedrukt.

Opgave 19. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 20. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.

b. Open de Arduino IDE via de snelkoppeling op het bureaublad.

c. Open de voorbeeld-sketch genaamd *Button.*
Bestand → Voorbeelden → 02.Digital → Button
Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.

d. Verwijder in de sketch het onderstaande geel gearceerde deel:

```
} else {
  // turn LED off:
  digitalWrite(ledPin, LOW);
}
```

e. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.

f. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.

g. Klaar! De LED op de Arduino zal nu bij eenmalig drukken continu blijven branden.
Opmerking: over de inhoud van de Arduino sketch hoef je je nog niet druk te maken.

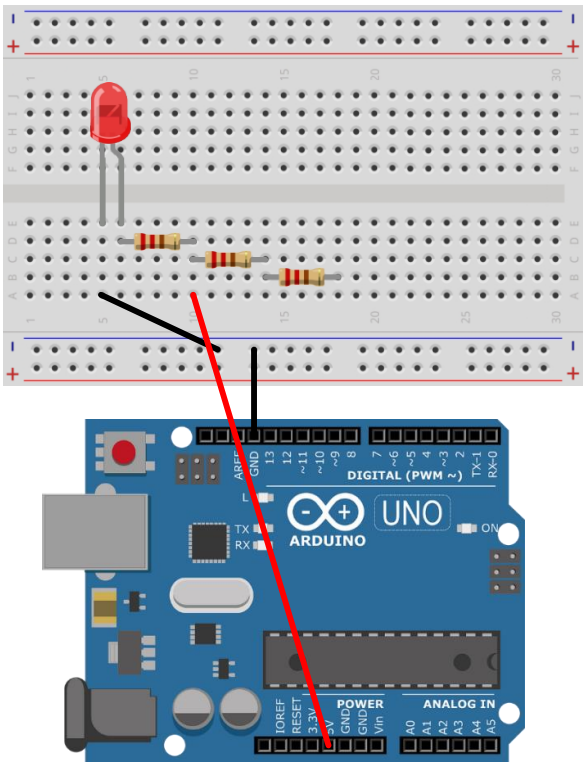
2.4 Weerstanden op de Arduino – Serieschakeling

Wat gebeurt er met de LED als je meerdere weerstanden aansluit? En maakt het dan nog uit of je de weerstanden aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *serieschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 weerstanden in *serie* aansluiten op de Arduino.

Opgave 21. Teken het schakelschema voor een stroomkring met een LED in serie met 3 voorschakelweerstanden op een spanningsbron.

Opgave 22. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 23. In de schakeling van opgave 22 staat nu slechts 1 weerstand in serie in de stroomkring. De overige 2 weerstanden doen nu niks. Verander de aansluiting van de rode jumper op het breadboard zodanig dat je 2 weerstanden in serie hebt staan in de stroomkring. Doe daarna hetzelfde voor 3 weerstanden. Wat neem je waar? Geef een toelichting.

_____

_____

_____

_____

_____

Opgave 24. Bereken de stroomsterkte door de LED bij 3 weerstanden in serie in de stroomkring.

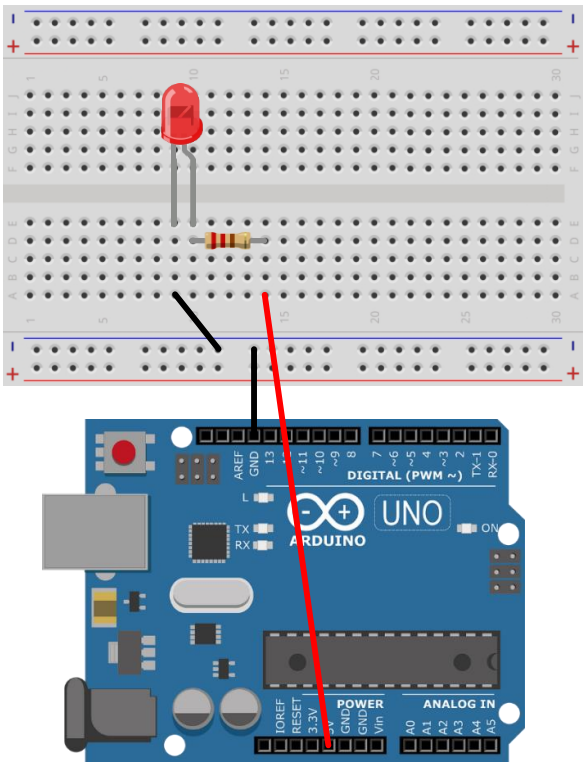
2.5 WEERSTANDEN OP DE ARDUINO – PARALLELSCHAKELING

Wat gebeurt er met de LED als je meerdere weerstanden aansluit? En maakt het dan nog uit of je de weerstanden aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *parallelschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 weerstanden in *parallel* aansluiten op de Arduino.

Opgave 25. Teken het schakelschema voor een stroomkring met een LED in serie met 3 parallel geschakelde voorschakelweerstanden op een spanningsbron.

Opgave 26. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 27. In de schakeling van opgave 26 staat nu 1 weerstand in serie met de LED in de stroomkring. Voeg nu eenzelfde weerstand toe op het breadboard zodat beide weerstanden ten opzichte van elkaar in parallel staan, maar samen nog wel in serie staan met de LED. Doe daarna hetzelfde voor 3 weerstanden.

Let op: *laat de LED telkens maar 1 seconde branden zodat hij niet kapot gaat!*

Wat neem je waar? Geef een toelichting.

_____

_____

_____

_____

_____

Opgave 28. Bereken de stroomsterkte door de LED bij 3 weerstanden parallel in de stroomkring.

Indien je zowel §2.4 als §2.5 hebt gemaakt:

Opgave 29. Brandt de LED het felst bij de weerstanden in serie- of parallelschakeling?
Geef een toelichting.

_____

_____

_____

_____

_____

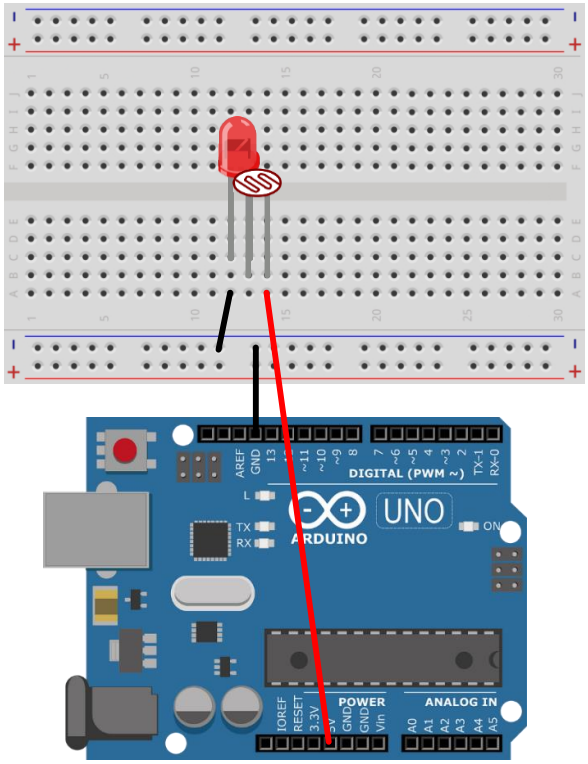
2.6 Een LDR op de Arduino

Een *LDR*, §1.3.4, is een weerstand die beïnvloed wordt door de hoeveelheid licht die erop valt. Hoe groter de lichtsterkte, hoe lager de weerstandswaarde. De LDR kan hierdoor goed als sensor dienen.

➢ Je gaat nu een LDR als sensor gebruiken om een LED feller of minder fel te laten branden.

Opgave 30. Teken het schakelschema voor een stroomkring met een LED in *serie* met een LDR aangesloten op een spanningsbron.

Opgave 31. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



➢ Laat nu afwisselend meer en minder licht op de LDR vallen.

Opgave 32. Wat neem je waar? Geef een toelichting.

_____

_____

_____

_____

_____

Opgave 33. Noem een toepassing waarvoor een LDR zou kunnen worden gebruikt.

_____

_____

_____

_____

_____

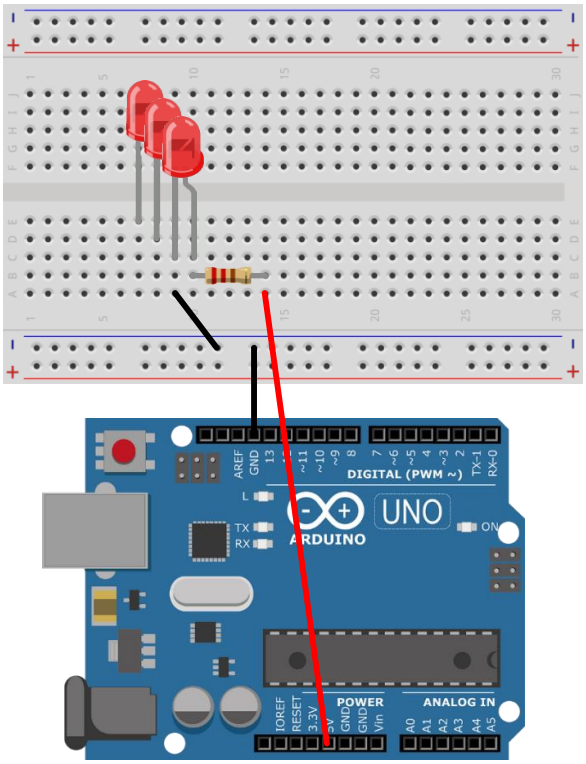
2.7 LEDs op de Arduino – Serieschakeling

Wat gebeurt er als je meerdere LEDs aansluit op de Arduino? En maakt het dan nog uit of je de LEDs aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *serieschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 LEDs in *serie* aansluiten op de Arduino.

Opgave 34. Teken het schakelschema voor een stroomkring met een voorschakelweerstand in serie met 3 LEDs aangesloten op een spanningsbron.

Opgave 35. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 36. In de schakeling van opgave 35 staat nu slechts 1 LED in serie in de stroomkring. De overige 2 LEDs doen nu niks. Verander de aansluiting van de bovenste zwarte jumper op het breadboard zodanig dat je 2 LEDs in serie hebt staan in de stroomkring. Doe daarna hetzelfde voor 3 LEDs. Wat neem je waar? Geef een toelichting.

_____

_____

_____

_____

_____

Opgave 37. Bereken de stroomsterkte door het aantal LEDs in serie in de stroomkring waarbij de LEDs nog wel branden.

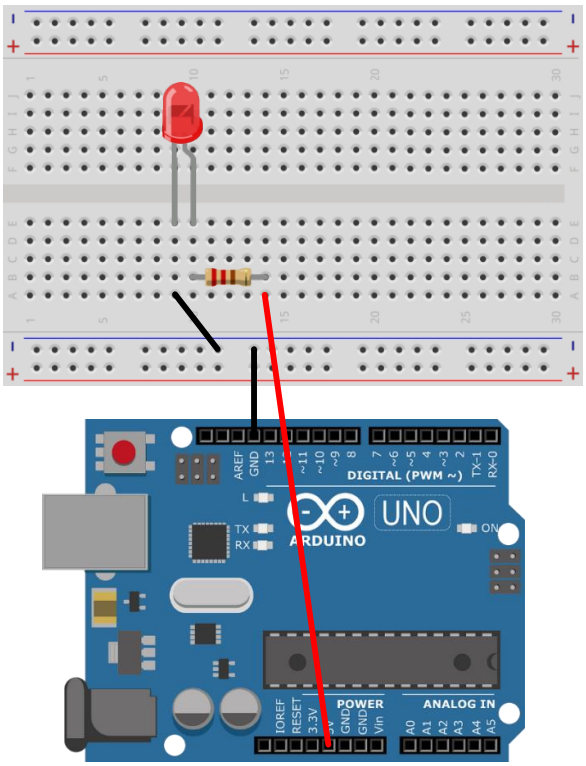
2.8 LEDs op de Arduino – Parallelschakeling

Wat gebeurt er als je meerdere LEDs aansluit op de Arduino? En maakt het dan nog uit of je de LEDs aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *parallelschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 LEDs in *parallel* aansluiten op de Arduino.

Opgave 38. Teken het schakelschema voor een stroomkring met een voorschakelweerstand in (**let op!**) serie met 3 parallel geschakelde LEDs aangesloten op een spanningsbron.

Opgave 39. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 40. In de schakeling van opgave 39 staat nu 1 LED in serie met de voorschakelweerstand in de stroomkring. Voeg nu eenzelfde LED toe op het breadboard zodat beide LEDs ten opzichte van elkaar in parallel staan, maar samen nog wel in serie staan met de voorschakelweerstand. Doe daarna hetzelfde voor 3 LEDs.
Wat neem je waar? Geef een toelichting.

_____

_____

_____

_____

_____

Opgave 41. Bereken de stroomsterkte door de LEDs in parallel in de stroomkring.

Indien je zowel §2.7 als §2.8 hebt gemaakt:

Opgave 42. Branden de LEDs het felst in serie- of parallelschakeling?
Geef een toelichting.

_____

_____

_____

_____

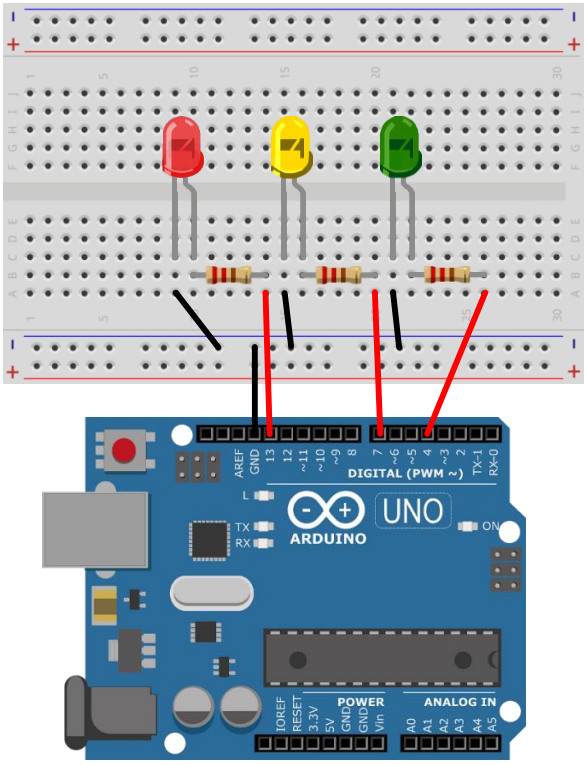
_____

2.9 LEDs afwisselen op de Arduino

Het is mogelijk om LEDs om beurten te laten branden door deze via aparte pinnen (ingangen) aan te sluiten op de Arduino en hiervoor een programma te schrijven. In dit onderdeel ga je dit uitproberen.

➢ Je gaat nu 3 verschillende LEDs achtereenvolgens laten branden.

Opgave 43. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 44. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.

b. Open de Arduino IDE via de snelkoppeling op het bureaublad.

c. Open de voorbeeld-sketch genaamd *Blink*

Bestand → Voorbeelden → 01.Basics → Blink

Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.

d. Omdat deze sketch nog niet doet wat je wil moet je deze aanpassen:

- Eerst moet je de Arduino vertellen welke pinnen je gaat gebruiken en of deze pinnen een ingang (INPUT) of uitgang (OUTPUT) zijn (zie voor uitleg figuur 17). Dit doe je bij void setup(). Je vertelt de Arduino hier dat pin 4, 7, en 13 gebruikt zullen worden, en dat deze zullen dienen als uitgang (OUTPUT). Hierdoor kun je van deze pinnen dan een 5 V uitgangsspanning ontvangen voor de LEDs.
- Vervolgens vertel je de Arduino welke taken hij moet uitvoeren.
  Dit doe je bij void loop(). Je vertelt de Arduino hier welke handelingen hij stapsgewijs moet uitvoeren, zoals het afgeven van een hoge spanning (5 V) aan pin 4 via de regel digitalWrite(4, HIGH); of het nemen van 1 seconde pauze via de regel delay (1000); (zie voor uitleg figuur 17).
- Neem onderstaande regels over in je sketch:

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(4, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(4, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(4, LOW);     // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second

  digitalWrite(7, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(7, LOW);     // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second

  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

e. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.

f. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.

g. Klaar! De LEDs op de Arduino zullen nu achtereenvolgens gaan branden.

3 PROJECTEN MET ARDUINO

Je gaat nu een mini-project doen met Arduino die bestaat uit een situatieschets die je met Arduino moet gaan realiseren. De projecten zijn verdeeld over twee niveaus: beginner en expert.

➢ Je gaat nu achtereenvolgens onderstaande projecten maken.

3.1 VOETGANGERSOVERSTEEKPLAATS – BEGINNER

Aan een drukke weg is voor voetgangers een oversteekplaats gemaakt. Als er geen voetgangers zijn staat het verkeerslicht voor auto's op groen. Voetgangers die willen oversteken kunnen op een knop drukken. Het verkeerslicht voor auto's gaat dan 1 seconde later stapsgewijs van groen, naar oranje, naar rood. Hierbij brandt het oranje licht 3 seconden lang. Het rode verkeerslicht gaat na 10 seconden te hebben gebrand weer op groen zodat de auto's weer mogen rijden.

Opgave 45. Maak de bovenstaande situatieschets met Arduino. **Let op:** zet de Arduino tijdelijk uit.

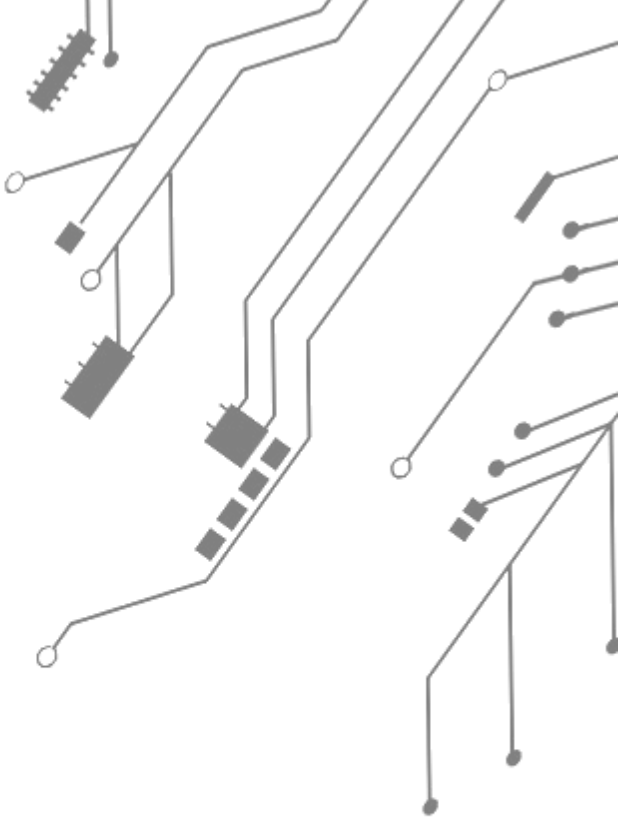
3.2 VOETGANGERSOVERSTEEKPLAATS – EXPERT

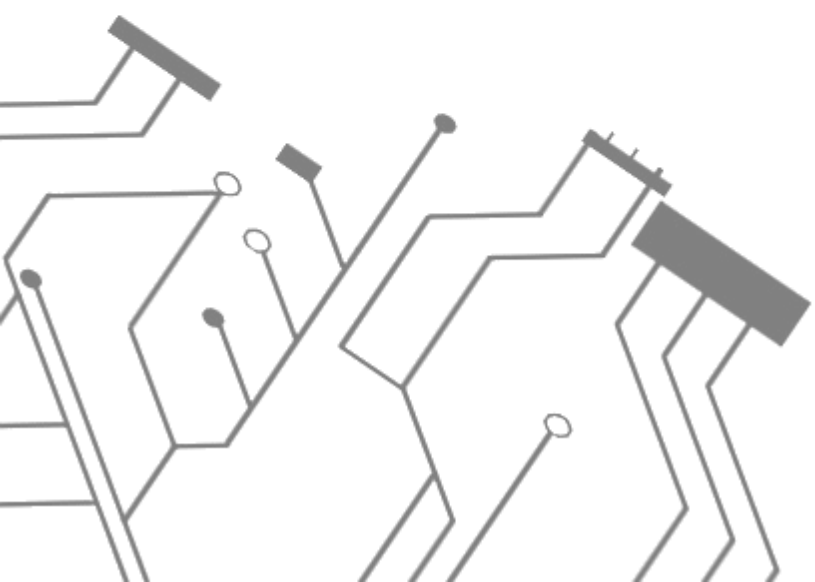
Aan een drukke weg is voor voetgangers een oversteekplaats gemaakt. Als er geen voetgangers zijn staat het verkeerslicht voor auto's op groen en het verkeerslicht voor voetgangers op rood. Voetgangers die willen oversteken kunnen op een knop drukken. Het verkeerslicht voor auto's gaat dan 1 seconde later van groen, naar oranje, naar rood. Hierbij brandt het oranje licht 3 seconden lang. Als het verkeerslicht voor de auto's op rood staat gaat 2 seconden later het verkeerslicht voor de voetgangers op groen. Voetgangers hebben dan 10 seconden de tijd om over te steken. Hierna knippert het voetgangerslicht 5 keer waarna het op rood springt. Dit knipperen gaat afwisselend: 0,5 seconden uit, 0,5 seconden aan. Vervolgens gaat 3 seconden later het verkeerslicht voor de auto's weer op groen.

Opgave 46. Maak de bovenstaande situatieschets met Arduino. **Let op:** zet de Arduino tijdelijk uit.

Kom je er niet uit, gebruik dan onderstaande tips of vraag om hulp:

- Zie §2.3 hoe je het indrukken van de drukknop kunt registreren en gebruiken.
- Zie §2.9 hoe je een stoplicht kunt nabootsen.
- Begin met het programma van §2.3 en verwerk hierin het programma van §2.9.
- Expert: voeg de benodigde hardware en software voor het voetgangerslicht zelf toe.

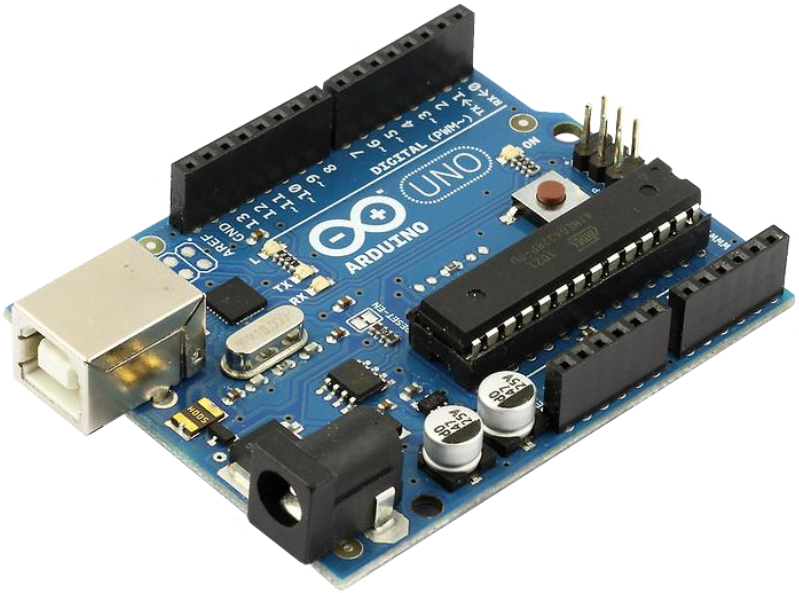




10.7 ARDUINO LESMODULE – DOCENT HANDLEIDING

Arduino lesmodule

Docenthandleiding



DOELGROEP

Deze Arduino lesmodule is opgezet voor het Natuurkunde curriculum van havo/vwo-leerjaar 3 waarin leerlingen 21e-eeuwse kennis en vaardigheden verwerven en versterken, en leerdoelen behalen voor de domeinen Elektriciteit en Technische automatisering.

Let op: van de begeleiders wordt wel enige voorkennis van Arduino vereist (bijvoorbeeld via zelfstudie).

LEERDOELEN

Arduino

- De leerling kan onderstaande Arduino hardware en software beschrijven en hiermee werken:
  - Arduino, inclusief voorbeelden van Arduino projecten
  - Microcontroller, inclusief enkele voordelen en voorbeelden in het dagelijks leven
  - Breadboard
  - Componenten: jumper, LED, weerstand, LDR, en drukknop
  - Arduino IDE, inclusief Arduino sketch, void setup(), void loop(), verifieer, en upload

Elektriciteit | Technische automatisering

- De leerling kan de drie basisdelen van een geautomatiseerd systeem benoemen, inclusief de hiervoor gebruikte componenten om die taken uit te voeren
- De leerling kan voorbeelden van een meet-, stuur-, en regelsysteem noemen met Arduino projecten
- De leerling kan rekenen met spanning, stroomsterkte, en weerstand met behulp van:
  - Wet van Ohm
  - Soortelijke weerstand
  - Serieschakelingen
  - Parallelschakelingen
- De leerling kan elektrische schakelingen tekenen, inclusief serie- en parallelschakelingen

21e-eeuwse vaardigheden

- De leerling kan een probleem herkennen en planmatig oplossen
- De leerling kan problemen zo formuleren dat ze op te lossen zijn met behulp van ICT
- De leerling ontwikkelt kennis en vaardigheden omtrent computer hardware en software
- De leerling kan gezamenlijk een doel behalen en anderen daarbij aanvullen en ondersteunen

TIJDSDUUR

Deze Arduino lesmodule is opgezet voor 4 lessen van 50 minuten. Leerlingen werken hierbij zelfstandig doorlopend aan de lesmodule. De lesmodule is zo ingericht dat bij minder tijd of gewenste leerdoelen, onderdelen overgeslagen kunnen worden. Zo kan de focus meer op Arduino worden gelegd door bijvoorbeeld de serie- en parallelschakeling met weerstanden en LEDs over te slaan, of kunnen Arduino onderdelen overgeslagen worden om juist de focus te leggen op deze serie- en parallelschakelingen.

VOORBEREIDING

	PER:
▪ Computer met Arduino IDE	*Groepje*
▪ Arduino set: Arduino, breadboard, jumpers, LEDs, weerstanden, LDR, drukknop	*Groepje*
▪ Printen: Arduino lesmodule (tip: A5-boekvorm)	*Leerling*
▪ Printen: Arduino lesmodule docenthandleiding	*Docent*

Opmerking: reset de Arduino IDE en Arduino voorafgaand aan de eerste les

OPZET

Focus de aandacht

- Toon een korte video over Arduino projecten, of een eigen Arduino project

Informeer over de doelstellingen

- Benoem dat deze lesmodule op praktische wijze ingaat op de leerstof over Elektriciteit en Technische automatisering, maar ook op kennis en vaardigheden voor de toekomst
- Kondig indien gewenst aan dat leerlingen beoordeeld worden op het werkboek (de Arduino lesmodule), en dat er een klassikale nabespreking volgt via willekeurige groepsleden

Stimuleer de voorkennis

- Benoem enkele voorbeelden uit de al behandelde leerstof die in deze lesmodule terugkomen, zoals: meet-, stuur-, en regelsysteem, wet van Ohm, serieschakelingen, et cetera

Indelen

- Deel leerlingen in tweetallen in
- Laat ieder tweetal tegenover elkaar plaatsnemen (in plaats van naast elkaar)
- Laat per les ieder tweetal afwisselen tussen ‘hardware’ en ‘software’. Dit houdt in dat ze alles wel samen moeten doen, maar dat ze iedere les onderling afwisselen tussen het invoeren van de programma’s op de computer en het bouwen met de Arduino set

Presenteer stimulerend materiaal

- Deel een Arduino set uit per groepje leerlingen
- Deel de Arduino lesmodule op papier uit aan iedere leerling

Zorg voor begeleiding

- Loop rond voor vragen en controleer of leerlingen op de goede weg zijn
- Bespreek bij ieder groepje hoe de samenwerking verloopt

Laat aantonen wat is geleerd

- De Arduino lesmodule voorziet hier grotendeels in, maar stel leerlingen kritische vragen

Geef feedback

- Stimuleer leerlingen door feedback te geven op het proces

Beoordeel de prestaties

- Beoordeel indien gewenst naderhand het ingevulde werkboek (de Arduino lesmodule)

Verbeter het behoud en overdracht

- Houd een klassikale nabespreking over de Arduino lesmodule en bijbehorende leerdoelen

INHOUDSOPGAVE

1 Kennismaken met Arduino 6

1.1 Arduino 6

1.2 Breadboard 8

1.3 Componenten 8

1.3.1 Jumper 8

1.3.2 LED 9

1.3.3 Weerstand 9

1.3.4 LDR 9

1.3.5 Drukknop 9

1.4 Software 10

1.4.1 Arduino sketch 10

1.4.2 Void setup() & void loop() 11

1.4.3 Verifieer & Upload 11

2 Werken met Arduino 12

2.1 Knipperende LED op de Arduino zelf 12

2.2 Een LED aansluiten op de Arduino 14

2.3 Een drukknop op de Arduino 18

2.4 Weerstanden op de Arduino – Serieschakeling 20

2.5 Weerstanden op de Arduino – Parallelschakeling 22

2.6 Een LDR op de Arduino 24

2.7 LEDs op de Arduino – Serieschakeling 26

2.8 LEDs op de Arduino – Parallelschakeling 28

2.9 LEDs afwisselen op de Arduino 30

3 Projecten met Arduino 32

3.1 Voetgangersoversteekplaats – Beginner 32

3.2 Voetgangersoversteekplaats – Expert 34

INLEIDING

We leven in een snel veranderende maatschappij waar digitalisering en technologische ontwikkelingen een belangrijke rol spelen. Als je alleen al om je heen kijkt zie je hoe afhankelijk we zijn van technologie. Van de bewegingssensor in het toilet voor het licht, tot aan het betalen met de pinpas bij de supermarkt voor een zak Apekoppen. Het is daarom belangrijk om voorbereid te worden op deze toekomst door het aanleren van vaardigheden die ook wel bekend staan als 21e-eeuwse vaardigheden. De Arduino is hiervoor heel geschikt, en je zal gaan zien waarom. Het werken met Arduino vereist wel een klein beetje kennis van elektronica en programmeren, maar hierbij zullen we je helpen. Arduino sluit ook heel goed aan bij de hoofdstukken *Elektriciteit* en *Technische Automatisering* uit je lesboek. Deze hoofdstukken zullen we dan ook veel gaan gebruiken voor de opdrachten in deze lesmodule.

➢ De leerdoelen die we willen bereiken in deze lesmodule zijn als volgt:

Arduino

- Je kunt onderstaande Arduino hardware en software beschrijven en hiermee werken:
  - Arduino, inclusief voorbeelden van Arduino projecten
  - Microcontroller, inclusief enkele voordelen en voorbeelden in het dagelijks leven
  - Breadboard
  - Componenten: jumper, LED, weerstand, LDR, en drukknop
  - Arduino IDE, inclusief Arduino sketch, void setup(), void loop(), verifieer, en upload

Elektriciteit | Technische automatisering

- Je kunt de drie basisdelen van een geautomatiseerd systeem benoemen, inclusief de hiervoor gebruikte componenten om die taken uit te voeren
- Je kunt voorbeelden van een meet-, stuur-, en regelsysteem noemen met Arduino projecten
- Je kunt rekenen met spanning, stroomsterkte, en weerstand met behulp van:
  - Wet van Ohm
  - Soortelijke weerstand
  - Serieschakelingen
  - Parallelschakelingen
- Je kunt elektrische schakelingen tekenen, inclusief serie- en parallelschakelingen

21e-eeuwse vaardigheden

- Je kunt een probleem herkennen en planmatig oplossen
- Je kunt problemen zo formuleren dat ze op te lossen zijn met behulp van ICT
- Je ontwikkelt kennis en vaardigheden omtrent computer hardware en software
- Je kunt gezamenlijk een doel behalen en anderen daarbij aanvullen en ondersteunen

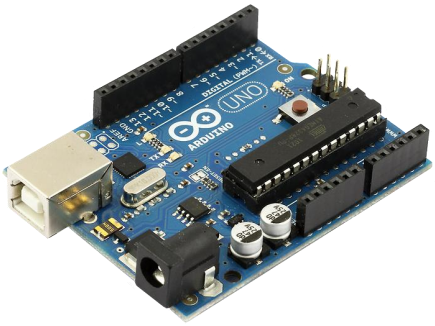
1 KENNISMAKEN MET ARDUINO

Je gaat nu kennismaken met Arduino. Je gaat bekijken wat een Arduino precies is, en hoe je hier mee kunt werken. Na dit onderdeel kun je aan de slag met het maken van je eigen Arduino programma's!

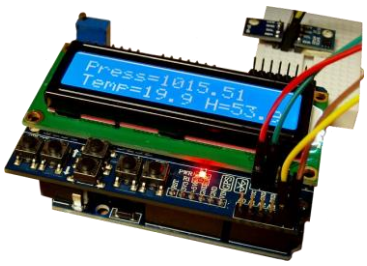
Pak alvast een Arduino-setje zodat je tijdens het lezen alles ook meteen kunt bekijken.

1.1 ARDUINO

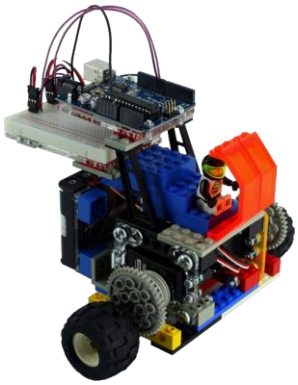
Een *Arduino*, figuur 1, kun je zien als een soort van kleine computer voor iedereen met interesse in het ontwerpen en maken van slimme en creatieve objecten die reageren op hun omgeving. Zo kun je bijvoorbeeld schakelaars, lichtsensoren, bewegingsmeters, afstandsmeters, en temperatuursensoren gebruiken om motoren, lampjes, pompjes en beeldschermen aan te sturen. Arduino projecten gaan van eenvoudige temperatuurmeters, figuur 2, tot aan compleet zelfrijdende robots, figuur 3.



Figuur 1 – Arduino

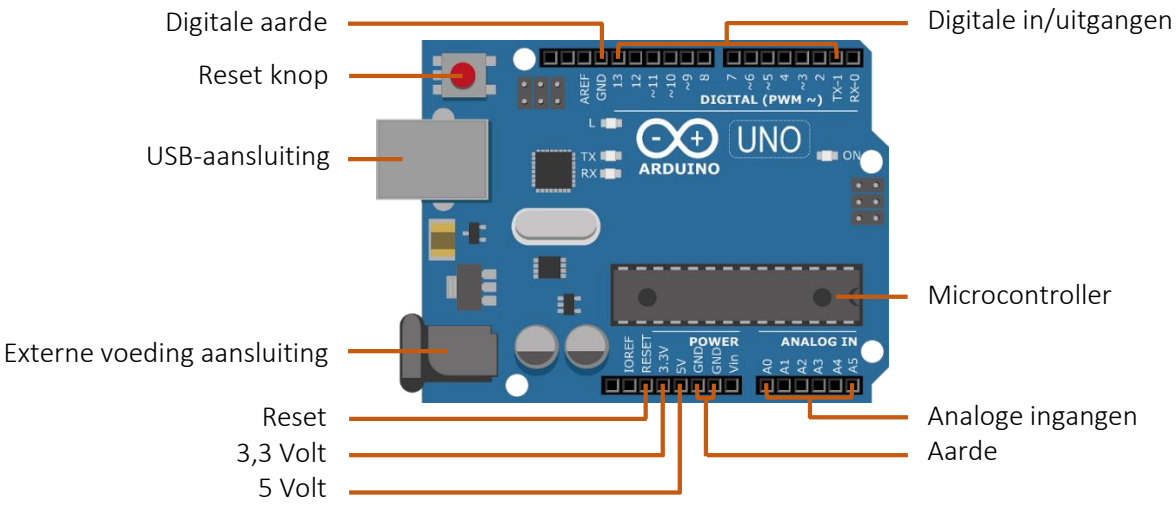


Figuur 2 – Arduino temperatuurmeter



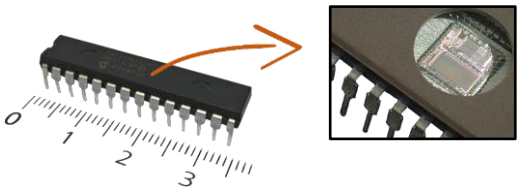
Figuur 3 – Arduino robot

De Arduino zelf is opgebouwd uit een microcontroller, ondersteunende componenten, en een aantal aansluitingen, figuur 4. De *microcontroller* is de computer. Deze is erg klein, figuur 5, en wordt met Arduino voor gebruikers eenvoudig toegankelijk gemaakt. De metalen pootjes van de microcontroller zijn op de Arduino namelijk verbonden met de langwerpige zwarte blokjes met gaatjes boven- en onderaan de Arduino. Op deze manier kun je snel en eenvoudig iets aansluiten en uitproberen.



Figuur 4 – Arduino schematisch weergegeven

Microcontrollers vind je in bijna alle moderne apparaten. Denk aan een televisie, robotstofzuiger, elektrische tandenborstel, of zelfs een zingende wenskaart. Ze worden gebruikt om al deze dingen automatisch aan te sturen, en zijn heel goed in het uitvoeren van één taak.



Figuur 5 – Microcontroller

In het geval van de televisie werkt dit als volgt: stel je selecteert een zender. De microcontroller in de televisie ontvangt het signaal van de afstandsbediening, verwerkt deze, en laat vervolgens de gekozen zender zien op het scherm. Deze drie stappen zijn simpelweg de basis van een geautomatiseerd systeem. Het stroomdiagram per systeemdeel is als volgt:

Tussen deze blokken worden signalen doorgegeven via kleine elektrische spanningen. Indien er geen signaal is, is de spanning 0V. Dit noemen we laag of 0. Bij een maximale spanning, 5 V in het geval van Arduino (ter vergelijking: een AA-batterij geeft 1.5 V), noemen we het signaal hoog of 1.

Opgave 1. Noem nog 3 apparaten binnenshuis waarin zich microcontrollers bevinden.

1. *Thermosstaat*
2. *Afstandsbediening*
3. *Radio*

Opgave 2. Noem 3 voordelen van het gebruik van microcontrollers.

1. *Klein*
2. *Goedkoop*
3. *Weinig energie nodig*

Opgave 3. Noem per systeemdeel uit het stroomdiagram in algemene zin de component die wordt gebruikt om deze taken uit te voeren (gebruik hiervoor je lesboek).

- Invoer: *sensor*
- Verwerking: *verwerker*
- Uitvoer: *actuator*

Opgave 4. Beschrijf aan de hand van invoer, verwerking, en uitvoer de werking van de afstands-bediening bij het selecteren van een andere televisiezender.

Je drukt op de knop van de afstandsbediening (sensor), dit signaal wordt verwerkt door de microcontroller (verwerker), waarna een signaal naar de televisie wordt gestuurd via het infraroodlampje (actuator).

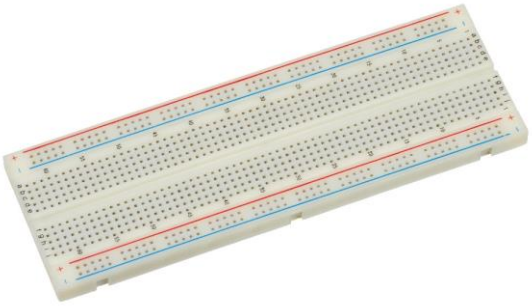
Opgave 5. Met Arduino kun je ontelbare objecten creëren. Noem voor elk van de onderstaande systemen (zie je lesboek) een object die je hiervoor met Arduino zou kunnen maken.

- Meetsysteem: *temperatuurmeter*
- Regelsysteem: *alarmsysteem*
- Stuursysteem: *plantwateringssysteem gestuurd door het vochtigheidsniveau*

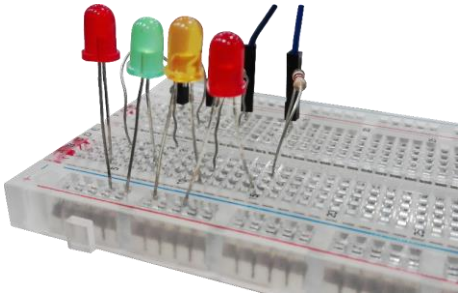
1.2 BREADBOARD

Om met de Arduino elektrische schakelingen te kunnen bouwen wordt het zogenaamde *breadboard* (letterlijk: broodplank) gebruikt, figuur 6. Het breadboard is niets meer dan een aantal rijen met gaatjes, genaamd *pinnen*, waarin je verschillende elektrische componenten kunt vastprikken. Denk bijvoorbeeld aan kabeltjes, LEDs, en weerstandjes, figuur 7. Het breadboard sluit je met kabeltjes aan op de Arduino via de pinnen (gaatjes) in de langwerpige zwarte blokjes boven- en onderaan de Arduino, figuur 4.

Binnenin het breadboard zitten metalen stripjes die de rijen met pinnen met elkaar verbinden. Dit ziet er schematisch uit zoals is weergegeven in figuur 8.

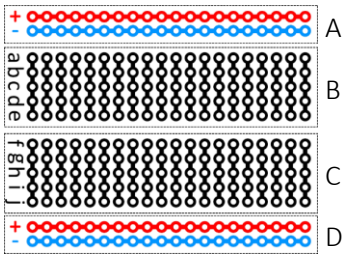


Figuur 6 – Breadboard



Figuur 7 – Breadboard met LEDs

De rijen A en D op het breadboard zijn voor de stroomvoorziening. Hierbij is de (+) rood en de (-) zwart of blauw. De (+) sluit je meestal aan op de 5 V uitgang van de Arduino. De (-) sluit je aan op de GND van de Arduino, wat staat voor ground (Engels voor aarde), ofwel 0 V. Zoals figuur 8 laat zien zijn de pinnen bij A en D per rij verbonden. Sluit je op één pin in deze rij iets aan dan heeft dat invloed op de gehele rij.



Figuur 8 – Breadboard schematisch weergegeven

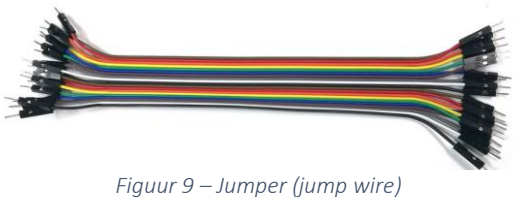
De kolommen B en C worden gebruikt voor de componenten, zoals LEDs. Zoals figuur 8 laat zien zijn de pinnen bij B en C per kolom verbonden. Sluit je op één pin in deze kolom iets aan dan heeft dat invloed op de gehele kolom. De kolommen van B en C zijn niet met elkaar verbonden.

1.3 COMPONENTEN

Er zijn veel componenten te gebruiken voor de Arduino. In deze lesmodule zullen wij de *jumper*, *LED*, *weerstand*, *LDR*, en *drukknop* beschrijven en gebruiken.

1.3.1 Jumper

Een *jumper* (officieel *jump wire*), figuur 9, is simpelweg een elektrische draad (kabeltje) die aan beide uiteinden een pinnetje heeft. Deze gebruik je om de Arduino te verbinden met het breadboard en hierop componenten met elkaar te verbinden, zoals LEDs en weerstanden.



Figuur 9 – Jumper (jump wire)

Er is geen verschil tussen de kleuren van een jumper. Je gebruikt deze enkel voor jezelf om onderscheid te maken in je opstelling, zoals rood voor de (+) aansluiting en zwart voor de (-) aansluiting.

1.3.2 LED

Een *LED (light-emitting diode)*, figuur 10, zendt licht uit als er een elektrisch stroom door de LED loopt. De LED laat stroom maar door in één richting, vandaar de term diode. De lange poot van de LED is de (+) aansluiting en de korte poot de (-) aansluiting.
Let op: sluit je de LED verkeerd om aan, dan doet de LED het niet!

Figuur 10 – LED

LEDs zijn heel bijzonder en niet te vergelijken met bijvoorbeeld een gloeilampje. Een gloeilampje is in principe een stabiele weerstand die door de lopende stroom verhit wordt en begint te gloeien. Een LED heeft deze eigenschap niet. Een LED werkt met een zogenaamde doorlaatspanning. Pas als deze spanning wordt bereikt gaat er een stroom lopen en zal de LED gaan branden, eerder niet. Opvallend is dat voorbij deze doorlaat-spanning de LED (bijna) geen weerstand meer heeft. Je kan de LED dan vergelijken met een zeer goed geleidende stroomdraad. Via de wet van Ohm zie je dan ook dat vanaf deze grens de stroomsterkte in heel korte tijd ontzettend groot wordt.

Figuur 11 – LED doorlaatspanning

De stroomsterkte door een LED is idealiter 20 mA. Bij een hogere stroomsterkte kan de LED al snel kapot gaan. De LED zal daarom altijd in serie geschakeld moeten worden met een weerstand die de stroomsterkte laag houdt. Welke weerstand je hiervoor moet kiezen bereken je via de wet van Ohm. Hierbij geldt voor de spanning over de weerstand: $U_R = U_{totaal} - U_{LED}$. De doorlaatspanning verschilt per LED: rood = 1.9 V; geel = 2.0 V; groen = 2.1 V.

1.3.3 Weerstand

Een *weerstand*, figuur 12, wordt gebruikt om de doorgang van een elektrische stroom te belemmeren. Dit is handig omdat voor sommige componenten de stroomsterkte anders te veel zou zijn, zoals voor de LED. De waarde van een weerstand wordt op de weerstand zelf aangegeven met kleurcodes. Elke kleur staat voor een getal.

Figuur 12 – Weerstand 220 Ω

Zwart	0
Bruin	1
Rood	2
Oranje	3
Geel	4
Groen	5
Blauw	6
Violet	7
Grijs	8
Wit	9

- Ring 1, 2 en 3 geven het 1e, 2e en 3e getal van de weerstand.
- Ring 4 geeft aan met welke macht van 10 het getal van ring 1, 2 en 3 vermenigvuldigd moeten worden.
- Ring 5 geeft het foutpercentage aan, ofwel hoeveel procent de werkelijke waarde af kan wijken van de opgegeven waarde.

Je begint met aflezen vaak bij de ring het dichtst bij de rand, of wat logisch is.
Voorbeeld: rood, rood, zwart, zwart, bruin = 220 · 10^1 Ω met 1% foutmarge.

1.3.4 LDR

Een *LDR (light-dependent resistor)*, figuur 13, is een weerstand die beïnvloed wordt door de hoeveelheid licht die erop valt. Hoe groter de lichtsterkte, hoe lager de weerstandswaarde. De LDR kan hierdoor goed als sensor dienen in schakelingen.

Figuur 13 – LDR

1.3.5 Drukknop

Een *drukknop*, figuur 14, wordt gebruikt om componenten zoals een LED aan te sturen. Wanneer er op de knop wordt gedrukt, worden de naast elkaar gelegen pootjes met elkaar verbonden (de zichtbare pootjes in figuur 14). De pootjes tegenover elkaar zijn altijd met elkaar verbonden, ongeacht of de knop is ingedrukt.

Figuur 14 – Drukknop

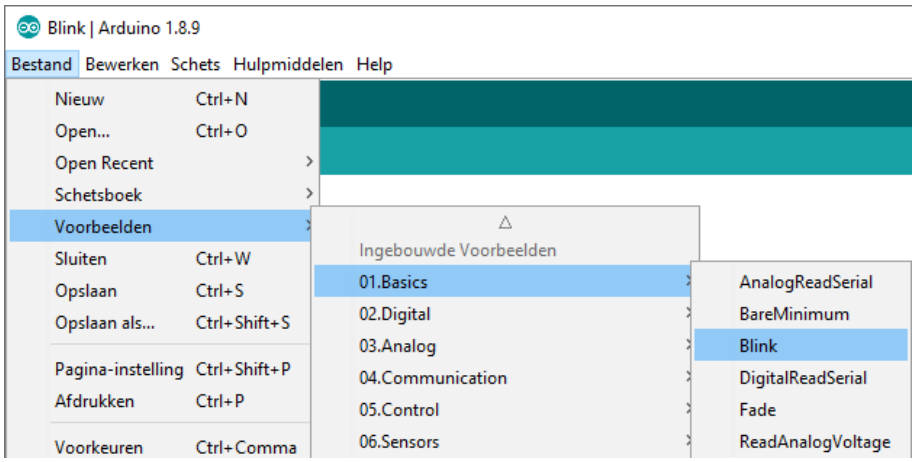
1.4 SOFTWARE

Om de Arduino een opdracht te kunnen laten uitvoeren moet je hem wel eerst vertellen wat hij precies moet doen. Dit doe je door een programma te schrijven waarin alle instructies voor de Arduino staan. Om je hierbij te helpen heeft Arduino speciale software waarmee je gemakkelijk je eigen programma kan schrijven. Ook kan deze software jouw programma meteen op de Arduino zetten. Deze software, figuur 15, heet de *Arduino IDE (Integrated Development Environment)*.

Figuur 15 – Arduino IDE met hierin geopend een lege sketch

1.4.1 Arduino sketch

Een programma die je maakt in de Arduino IDE heet een *sketch*. In figuur 15 staat bijvoorbeeld een lege sketch geopend. In de Arduino IDE zijn veel voorbeeld sketches al voor je gemaakt, zodat je deze niet zelf hoeft te schrijven. Deze sketches, zoals 'Blink' waarmee je een LED kunt laten knipperen, kun je vinden onder Voorbeelden, figuur 16.



Figuur 16 – Arduino IDE sketch voorbeelden

1.4.2 Void setup() & void loop()

De Arduino sketch bestaat altijd uit minstens twee elementen: *void setup()* en *void loop()*, figuur 15. In void setup() staat alles wat van belang is voordat de Arduino een taak gaat uitvoeren, zoals welke aansluiting (pin/gaatje) de Arduino moet gebruiken en wat hij hiermee moet doen. In void loop() staat de opdracht die de Arduino daadwerkelijk moet gaan uitvoeren, zoals het laten knipperen van een LED.

De Arduino zal bij het uitvoeren van een sketch void setup() slechts eenmalig uitvoeren. Void loop() wordt echter, zoals de naam al aangeeft, keer op keer herhaald. Hierdoor kan de Arduino heel goed één specifieke taak uitvoeren. Hoe je void setup() en void loop() moet invullen leer je in de opdrachten.

Standaard termen, zoals void setup() en void loop(), krijgen in de Arduino IDE altijd een specifieke kleur. *Voor meer informatie over deze termen kun je kijken op: https://www.arduino.cc/reference/en/.*

1.4.3 Verifieer & Upload

Als de Arduino sketch klaar is kan deze in principe naar de Arduino worden verzonden. Om ervoor te zorgen dat er geen fouten in de sketch staan laat je deze altijd eerst door de Arduino IDE controleren. Dit gaat via de knop *Verifieer* linksboven in het scherm, figuur 15. Als er geen fouten in zitten geeft de Arduino IDE dit linksonder in het scherm weer met de opmerking *Compileren voltooid*. Als er wel fouten zijn krijg je hierover specifieke informatie te zien die je helpt de fout op te lossen.

Vervolgens kun je de sketch naar de Arduino sturen via de knop *Upload* linksboven in het scherm, figuur 15. Je hoeft de sketch niet eerst nog op te slaan, dat gaat automatisch. Als alles met het uploaden goed is gegaan geeft de Arduino IDE dit linksonder in het scherm weer met de opmerking *Uploaden voltooid*. De Arduino zal nu de sketch gaan uitvoeren.

Let op: *zowel de Arduino IDE, als de Arduino zelf, kunnen nog een sketch van de vorige keer bevatten. De Arduino IDE en Arduino onthouden namelijk de sketch die voor het laatst opgeslagen is. Dit kan dus ook de sketch zijn die niet door jou is gemaakt, maar door een klasgenoot. Open daarom altijd eerst opnieuw jouw eigen sketch en upload deze voor de zekerheid naar de Arduino.*

2 WERKEN MET ARDUINO

Je gaat nu werken met Arduino. Je gaat beginnen met een eenvoudige knipperende LED op de Arduino zelf, waarna je steeds een stapje verder gaat. Na dit onderdeel kun je aan de slag met je eerste project.

2.1 KNIPPERENDE LED OP DE ARDUINO ZELF

Op de Arduino zelf zitten ook LEDs. Deze worden gebruikt om te laten zien dat de Arduino aan staat (ON), data ontvangt (RX), of data verstuurt (TX). Er is echter nog een vierde LED welke je zelf kunt aansturen. In dit onderdeel ga je deze LED laten knipperen door de Arduino te programmeren.

Opgave 6. Zoek de zelf aan te sturen LED op de Arduino en schrijf op welke letter hierbij staat.

De letter L

➢ Je gaat nu de LED op de Arduino laten knipperen met behulp van een kant-en-klare sketch.

Opgave 7. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.
b. Open de Arduino IDE via de snelkoppeling op het bureaublad.
c. Open de voorbeeld-sketch genaamd *Blink.*
   Bestand → *Voorbeelden* → *01.Basics* → *Blink* (zie figuur 16)
   Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.
d. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.
e. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.
f. Klaar! De LED op de Arduino zelf zal nu gaan knipperen.

Je hebt zojuist je eerste programma op de Arduino gezet en de Arduino een opdracht laten uitvoeren!

Voordat we verder gaan willen we je eerst een aantal *algemene regels* vertellen waar je bij het schrijven van programma's rekening mee moet houden:

ALGEMENE REGELS

- Computers lezen een programma stapsgewijs van boven naar beneden, regel voor regel.
- De programmeertaal van Arduino is hoofdlettergevoelig, ofwel: *Arduino* is anders dan *arduino*.
- Functies, zoals *void setup()* en *void loop()*, gebruiken accolades { } om aan te geven wat bij de functie hoort. Ieder open haakje moet altijd worden gesloten (net zoals in je rekenmachine).
- Elke regel die compleet is moet afgesloten worden met een puntkomma ; In de praktijk is dit meestal iedere regel. Uitzonderingen zijn het commentaar // en afsluiten van een functie }.

➢ Je gaat nu stapsgewijs de opzet van de sketch 'Blink' bekijken, figuur 17.

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Figuur 17 – Arduino IDE sketch 'Blink'

//	Dit is de commentaarregel. Alles wat hierachter staat negeert de computer.
void setup()	Deze functie wordt door de Arduino slechts eenmalig uitgevoerd. Hierin staat alles wat van belang is voordat de Arduino een taak gaat uitvoeren, zoals *pinMode*.
pinMode	Hiermee vertel je aan de Arduino welke pin (aansluiting) hij moet gebruiken en wat hij hiermee moet doen. In dit geval wordt Arduino vertelt dat hij de ingebouwde LED met de naam *LED_BUILTIN* moet gebruiken als een uitgang (*OUTPUT*). Bij *OUTPUT* levert de Arduino spanning. Bij *INPUT* ontvangt de Arduino spanning, zoals van een sensor. *Let op: de ingebouwde LED, hier LED_BUILTIN genoemd, is altijd gekoppeld aan pin 13.*
void loop()	Deze functie wordt door de Arduino herhaaldelijk uitgevoerd. Hierin staat de opdracht die de Arduino daadwerkelijk moet uitvoeren, zoals *digitalWrite*.
digitalWrite	Hiermee kun je op een digitale pin, zoals pin 13, een hoge of lage spanning zetten. Hiervoor moet je wel aangeven welke pin je kiest, en of deze een hoge spanning (5 V) of lage spanning (0 V) moet krijgen. De eerste regel hier geeft dus aan dat *LED_BUILTIN* (ofwel: pin 13) een hoge spanning (*HIGH*) moet krijgen. Hierdoor zal de LED branden. *Let op: de genoemde pin moet wel eerst benoemd zijn in void setup() om te werken.*
delay	Hiermee kun je de computer even laten wachten voordat hij de volgende regel uitvoert. Op deze manier kun je dus een knipperende LED maken: LED aan, 1 s wachten, LED uit, 1 s wachten. Deze cyclus, ofwel het deel tussen { }, herhaalt zich steeds weer opnieuw. *Let op: de ingevulde waarden in de functie delay zijn altijd in milliseconde.*

Opgave 8. Verander de sketch zodanig dat de LED op de Arduino steeds 0,10 seconde aan is en dan weer 0,25 seconde uit blijft. Vul hieronder in wat je in de sketch hebt ingevuld.

```
void loop()  {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(100);
  digitalWrite(LED_BUILTIN, LOW);
  delay(250);
}
```

2.2 EEN LED AANSLUITEN OP DE ARDUINO

Het is natuurlijk veel leuker om zelf een LED aan te sluiten op de Arduino. Hiervoor kun je de Arduino gebruiken als spanningsbron, en een stroomkring maken waarin je de LED opneemt. Echter heb je al gelezen in §1.3.2 dat hiervoor ook een weerstand nodig is. Deze noem je een voorschakelweerstand, omdat deze weerstand in de stroomkring altijd in serie vóór de LED moet worden geschakeld.

➢ Je gaat nu ter *bewustwording* eerst kijken naar een situatie *zonder* voorschakelweerstand.

Opgave 9. De weerstand van een LED zou je *voorbij* de doorlaatspanning kunnen *benaderen* alsof het een stroomdraad is, §1.3.2. Hiervoor kun je de soortelijke weerstand gebruiken. Bereken de weerstand van de LED uitgaande dat deze als stroomdraad 2 jumpers lang is (20 cm per jumper), een diameter heeft van 0,5 mm, en gemaakt is van ijzer.

Gegeven: $\rho = 0{,}105\ \Omega\cdot mm^2/m$ *(BINAS tabel 8)*
$l = 40\ cm = 0{,}40\ m$
$d = 0{,}5\ mm$

Gevraagd: $R = ?$

Uitwerking: $$R = \frac{\rho \cdot l}{A} = \frac{\rho \cdot l}{\pi \cdot r^2} = \frac{0{,}105 \cdot 0{,}40}{\pi \cdot (\frac{0{,}5}{2})^2} = 0{,}2\ \Omega$$

Opgave 10. De Arduino levert als spanningsbron een spanning van 5.0 V. De weerstandswaarde van de LED heb je bij opgave 9 berekend. Bereken nu de stroomsterkte door de LED.

Gegeven: $U = 5{,}0\ V$
$R = \frac{0{,}105 \cdot 0{,}40}{\pi \cdot \left(\frac{0{,}5}{2}\right)^2} = 0{,}2\ \Omega$

Gevraagd: $I = ?$

Uitwerking: $$I = \frac{U}{R} = \frac{5{,}0}{\left(\frac{0{,}105 \cdot 0{,}40}{\pi \cdot (\frac{0{,}5}{2})^2}\right)} = 23\ A = 2 \cdot 10^1\ A$$

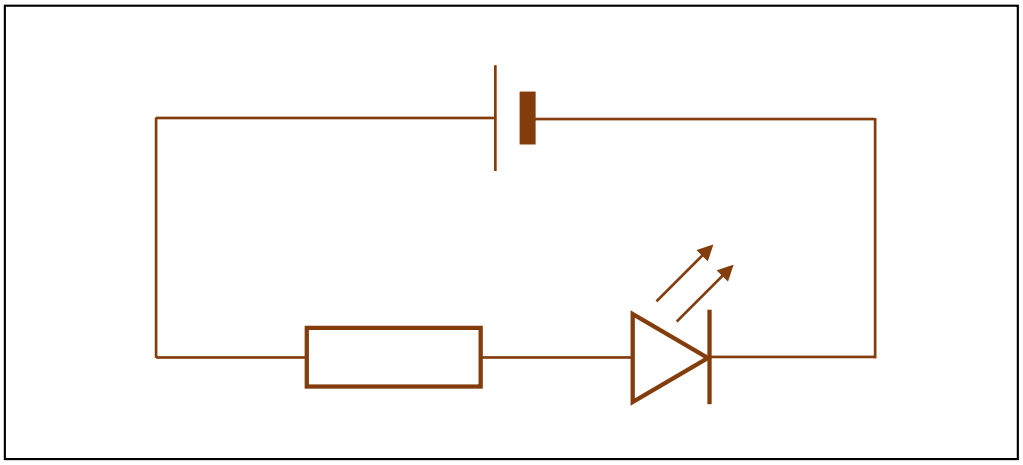
(via afrondingswaarde 0,2 Ω: $I = \frac{5{,}0}{0{,}2} = 25\ A = 3 \cdot 10^1\ A$*)*

Bedenk je nu dat een LED al kapot kan gaan als er meer dan 20 mA (= 0,020 A) doorgaat! Om deze reden moet er dus altijd een voorschakelweerstand in serie worden gebruikt.

Opmerking: in de praktijk zal de Arduino zo'n hoge stroomsterkte echter nooit behalen.

➢ Je gaat nu de *voorbereidingen* treffen om de LED aan te kunnen sluiten op de Arduino.

Opgave 11. Teken het schakelschema voor een stroomkring met de LED en voorschakelweerstand in serie aangesloten op een spanningsbron.



In de stroomkring moet de voorschakelweerstand voldoende groot zijn zodat de stroomsterkte door de LED kleiner is dan 20 mA, §1.3.2. Hiervoor gebruik je de wet van Ohm en regels voor serieschakeling.

Opgave 12. De Arduino levert als spanningsbron een spanning van 5.0 V. Op een rode LED staat 1,9 V / 20 mA (zie voor een toelichting §1.3.2). Bereken de waarde van de voorschakelweerstand die nodig is om de rode LED op de juiste spanning te laten branden.
Let op: vergeet niet de doorlaatpsanning mee te nemen in je berekening, zie §1.3.2.

Gegeven: $U_{LED} = 1{,}9\ V$
$U_{tot} = 5{,}0\ V$
$I = 20\ mA = 0{,}020\ A$

Gevraagd: $R = ?$

Uitwerking: $U_{tot} = U_{LED} + U_R$
$5{,}0 = 1{,}9 + U_R$
$U_R = 3{,}1\ V$

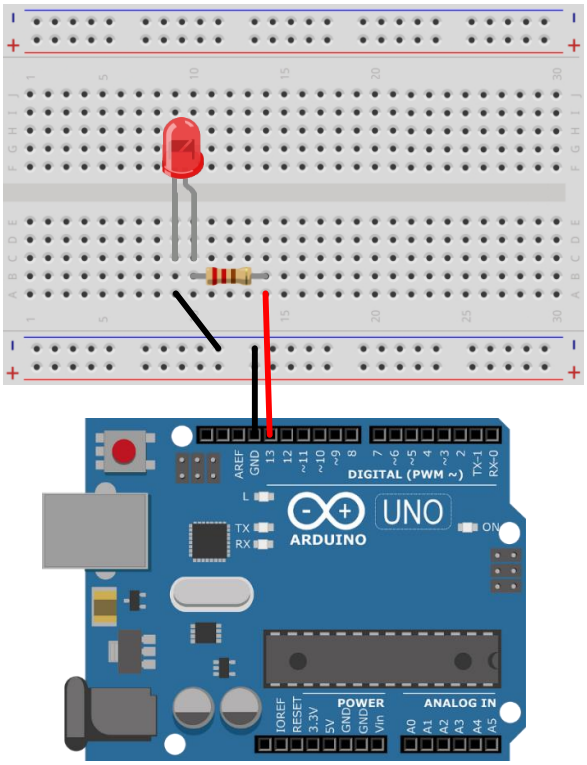
$$R = \frac{U_R}{I} = \frac{3{,}1}{0{,}020} = 155\ \Omega = 1{,}6 \cdot 10^2\ \Omega$$

➢ Je gaat nu de LED aansluiten op de Arduino.

Opgave 13. Zoek in je Arduino setje een weerstand, §1.3.3, die zo dicht mogelijk bij de berekende waarde uit opgave 12 ligt. Let op: de weerstand mag in ieder geval niet kleiner zijn.

Bijvoorbeeld: 220 Ω (kleurcode: rood – rood – zwart – zwart – bruin, zie figuur 12)

Opgave 14. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 15. Voer de volgende stappen uit.
Let op: dit hoeft niet als de sketch 'Blink' uit opgave 7 nog op de Arduino staat.

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.
b. Open de Arduino IDE via de snelkoppeling op het bureaublad.
c. Open de voorbeeld-sketch genaamd *Blink.*
   Bestand → Voorbeelden → 01.Basics → Blink
   Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.
d. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.
e. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.
f. Klaar! De LED zal nu gaan knipperen.

Opgave 16. Verander de sketch zodanig dat de LED niet meer knippert maar continu brandt. Controleer of het werkt en vul hieronder in wat je in de sketch hebt ingevuld.

```
void loop()  {
   digitalWrite(LED_BUILTIN, HIGH);









}
```

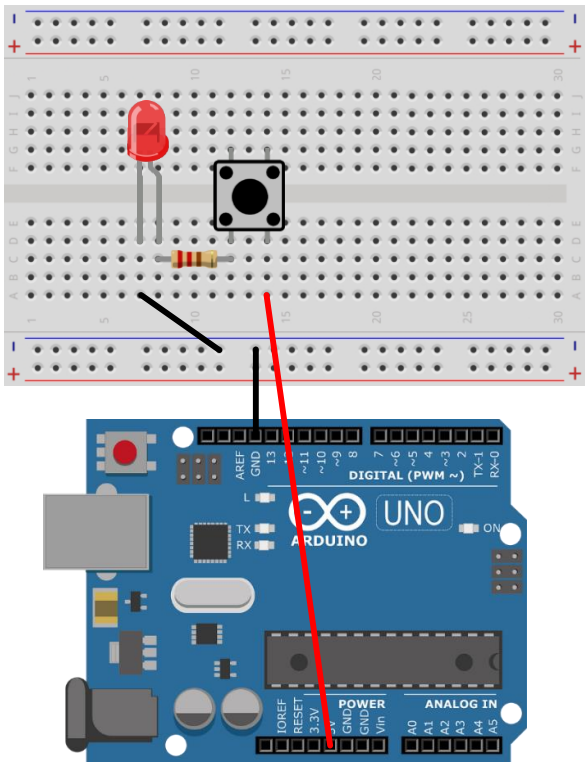
2.3 EEN DRUKKNOP OP DE ARDUINO

Een *drukknop*, §1.3.5, wordt gebruikt om componenten zoals een LED aan te sturen. Wanneer er op de knop wordt gedrukt worden de naast elkaar gelegen pootjes met elkaar verbonden. De pootjes tegenover elkaar zijn altijd met elkaar verbonden, ongeacht of de knop is ingedrukt.

➢ Je gaat nu met een drukknop een LED laten branden.

Opgave 17. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.
Opmerking: zorg ervoor dat de drukknop stevig op het breadboard zit vastgedrukt.

Voor het leren werken met Arduino gebruik je nu in plaats van pin 13 de standaard 5 V pin op de Arduino. Deze aansluiting levert continu 5 V. Dit is ook niet te veranderen, in tegenstelling tot bijvoorbeeld pin 13 (denk aan de knipperende LED). De aansluiting met 5 V is echter zeer handig om een component continu te voorzien van spanning, of tijdelijk in het geval van de drukknop. Je hoeft hiervoor niets te programmeren.



Opgave 18. Wat neem je waar? Geef een toelichting.

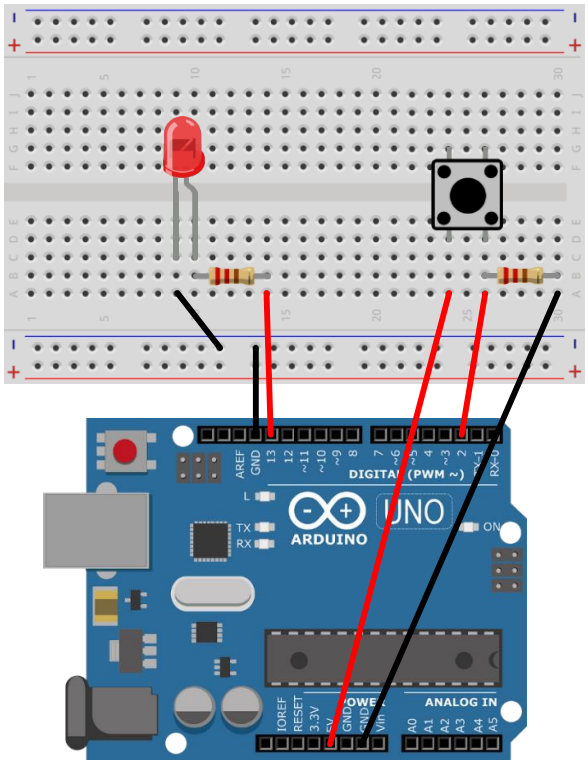
De LED brandt als je de knop indrukt. Als je de knop niet indrukt brandt de LED niet.

Toelichting: Bij het indrukken van de knop wordt de stroomkring gesloten en kan er een stroom lopen waardoor de LED gaat branden. Bij het loslaten van de knop is de stroomkring niet meer gesloten waardoor er geen stroom loopt en de LED niet meer brandt.

Het is ook mogelijk om de Arduino te laten registreren dat de drukknop is ingedrukt. Dit is handig als je de LED continu wil laten branden zonder de drukknop ingedrukt te moeten houden.

➢ Je gaat nu met een drukknop een LED continu laten branden nadat je er eenmalig op hebt gedrukt.

Opgave 19. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 20. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.

b. Open de Arduino IDE via de snelkoppeling op het bureaublad.

c. Open de voorbeeld-sketch genaamd *Button.*
Bestand → Voorbeelden → 02.Digital → Button
Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.

d. Verwijder in de sketch het onderstaande geel gearceerde deel:

```
  } else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
```

e. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.

f. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.

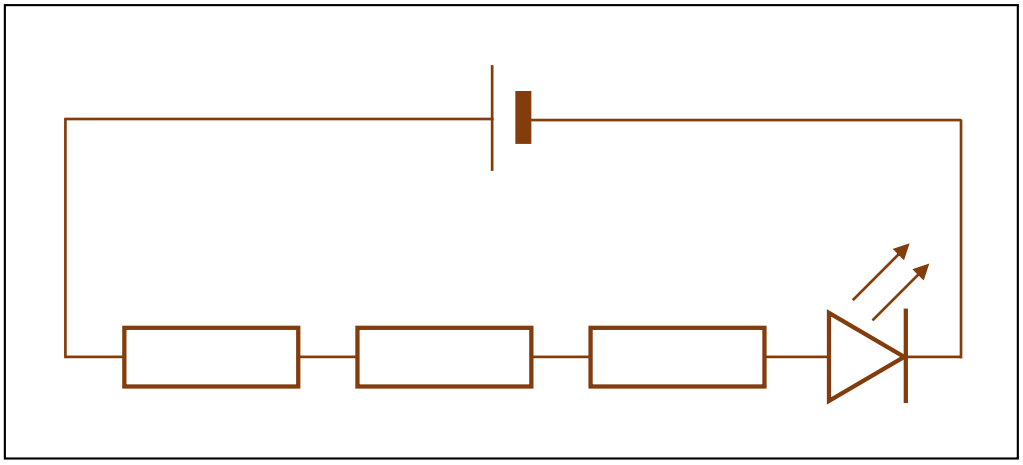
g. Klaar! De LED op de Arduino zal nu bij eenmalig drukken continu blijven branden.
Opmerking: over de inhoud van de Arduino sketch hoef je je nog niet druk te maken.

2.4 Weerstanden op de Arduino – Serieschakeling

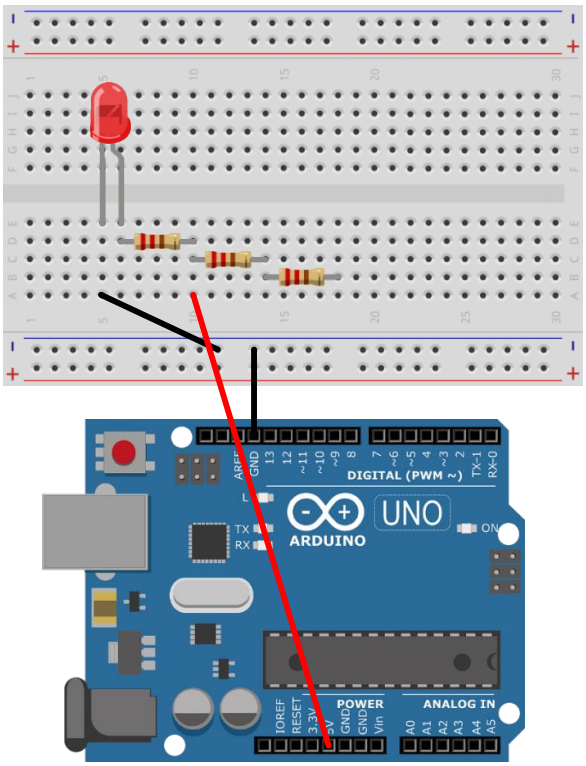
Wat gebeurt er met de LED als je meerdere weerstanden aansluit? En maakt het dan nog uit of je de weerstanden aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *serieschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 weerstanden in *serie* aansluiten op de Arduino.

Opgave 21. Teken het schakelschema voor een stroomkring met een LED in serie met 3 voorschakelweerstanden op een spanningsbron.



Opgave 22. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 23. In de schakeling van opgave 22 staat nu slechts 1 weerstand in serie in de stroomkring. De overige 2 weerstanden doen nu niks. Verander de aansluiting van de rode jumper op het breadboard zodanig dat je 2 weerstanden in serie hebt staan in de stroomkring. Doe daarna hetzelfde voor 3 weerstanden. Wat neem je waar? Geef een toelichting.

De LED brandt steeds minder fel naarmate er meer weerstanden in serie in de stroomkring worden opgenomen.

Toelichting: de totale weerstand neemt toe naarmate er meer weerstanden in serie in de schakeling worden opgenomen. Hierdoor neemt volgens de wet van Ohm de stroomsterkte af, en daarmee de felheid van de LED.

Opgave 24. Bereken de stroomsterkte door de LED bij 3 weerstanden in serie in de stroomkring.

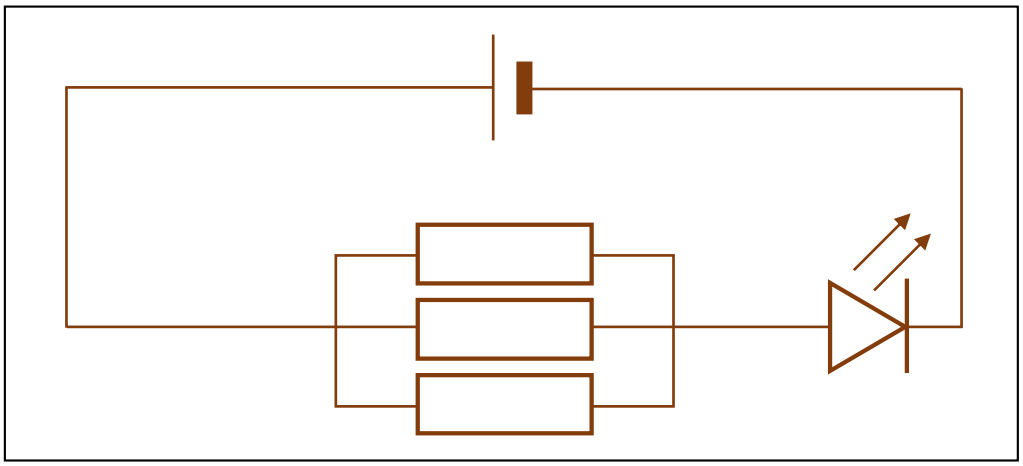
Gegeven:	$U_{LED} = 1{,}9\ V$ $U_{tot} = 5{,}0\ V$ $R_{tot} = R_1 + R_2 + R_3 = 220 + 220 + 220 = 660\ \Omega$
Gevraagd:	$I = ?$
Uitwerking:	$U_{tot} = U_{LED} + U_R$ $5{,}0 = 1{,}9 + U_R$ $U_R = 3{,}1\ V$ $I = \frac{U_R}{R_{tot}} = \frac{3{,}1}{660} = 4{,}7\ mA$

2.5 WEERSTANDEN OP DE ARDUINO – PARALLELSCHAKELING

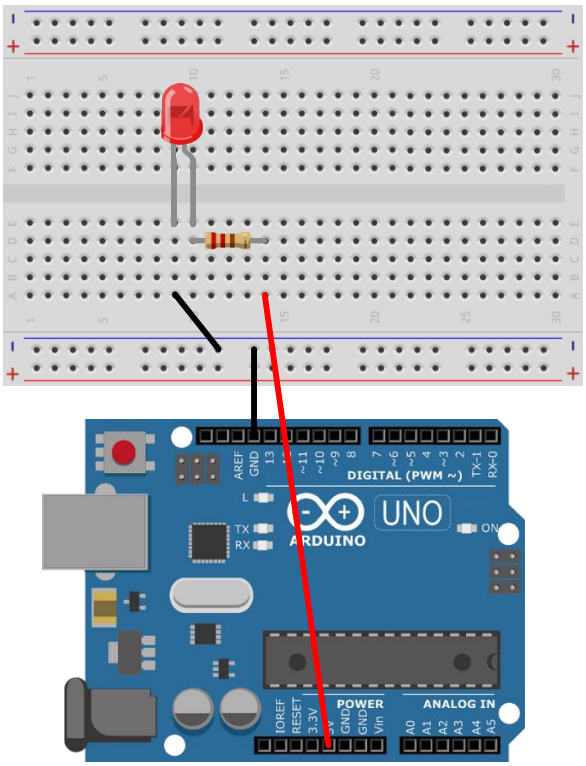
Wat gebeurt er met de LED als je meerdere weerstanden aansluit? En maakt het dan nog uit of je de weerstanden aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *parallelschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 weerstanden in *parallel* aansluiten op de Arduino.

Opgave 25. Teken het schakelschema voor een stroomkring met een LED in serie met 3 parallel geschakelde voorschakelweerstanden op een spanningsbron.



Opgave 26. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 27. In de schakeling van opgave 26 staat nu 1 weerstand in serie met de LED in de stroomkring. Voeg nu eenzelfde weerstand toe op het breadboard zodat beide weerstanden ten opzichte van elkaar in parallel staan, maar samen nog wel in serie staan met de LED. Doe daarna hetzelfde voor 3 weerstanden.

Let op: laat de LED telkens maar 1 seconde branden zodat hij niet kapot gaat!

Wat neem je waar? Geef een toelichting.

De LED brandt steeds feller naarmate er meer weerstanden in parallel in de stroomkring worden opgenomen.

Toelichting: de totale weerstand neemt af naarmate er meer weerstanden in parallel in de schakeling worden opgenomen. Hierdoor neemt volgens de wet van Ohm de stroomsterkte toe, en daarmee de felheid van de LED.

Opgave 28. Bereken de stroomsterkte door de LED bij 3 weerstanden parallel in de stroomkring.

Gegeven:	$U_{LED} = 1{,}9\ V$ $U_{tot} = 5{,}0\ V$ $\frac{1}{R_{tot}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} = \frac{1}{220} + \frac{1}{220} + \frac{1}{220}$ → $R_{tot} = 73{,}3\ \Omega$
Gevraagd:	$I = ?$
Uitwerking:	$U_{tot} = U_{LED} + U_R$ $5{,}0 = 1{,}9 + U_R$ $U_R = 3{,}1\ V$ $I = \frac{U_R}{R_{tot}} = \frac{3{,}1}{73{,}3} = 42\ mA$

Indien je zowel §2.4 als §2.5 hebt gemaakt:

Opgave 29. Brandt de LED het felst bij de weerstanden in serie- of parallelschakeling?
Geef een toelichting.

Parallelschakeling.

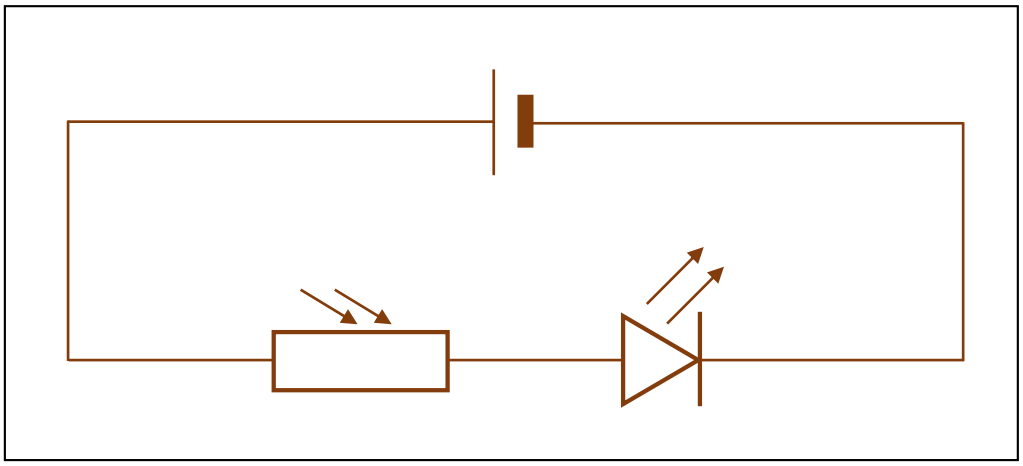
Toelichting: in de parallelschakeling loopt een grotere stroom dan in de serieschakeling, waardoor de LED in de parallelschakeling het felst brandt.

2.6 Een LDR op de Arduino

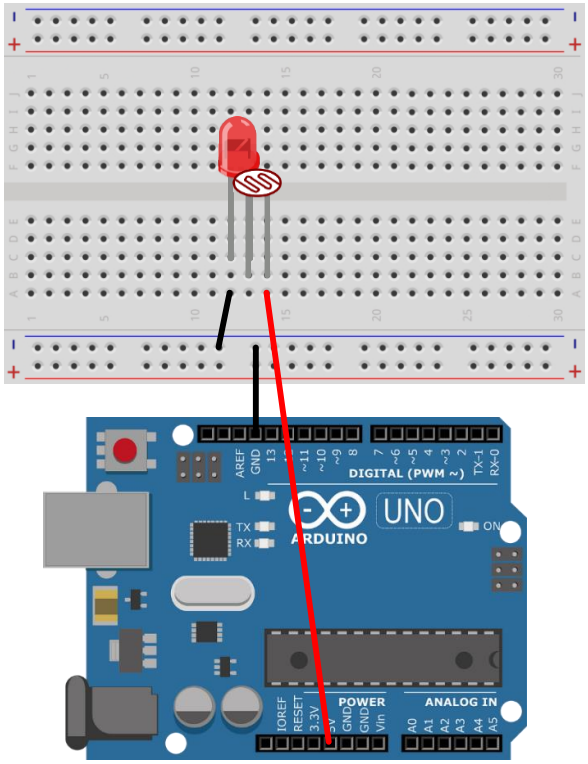
Een *LDR*, §1.3.4, is een weerstand die beïnvloed wordt door de hoeveelheid licht die erop valt. Hoe groter de lichtsterkte, hoe lager de weerstandswaarde. De LDR kan hierdoor goed als sensor dienen.

➢ Je gaat nu een LDR als sensor gebruiken om een LED feller of minder fel te laten branden.

Opgave 30. Teken het schakelschema voor een stroomkring met een LED in *serie* met een LDR aangesloten op een spanningsbron.



Opgave 31. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



➢ Laat nu afwisselend meer en minder licht op de LDR vallen.

Opgave 32. Wat neem je waar? Geef een toelichting.

Hoe meer licht er op de LDR valt, hoe feller de LED brandt.

Toelichting: hoe groter de lichtsterkte op de LDR, hoe lager de weerstandswaarde van de LDR wordt. Hierdoor neemt volgens de wet van Ohm de stroomsterkte toe, en daarmee de felheid van de LED.

Opgave 33. Noem een toepassing waarvoor een LDR zou kunnen worden gebruikt.

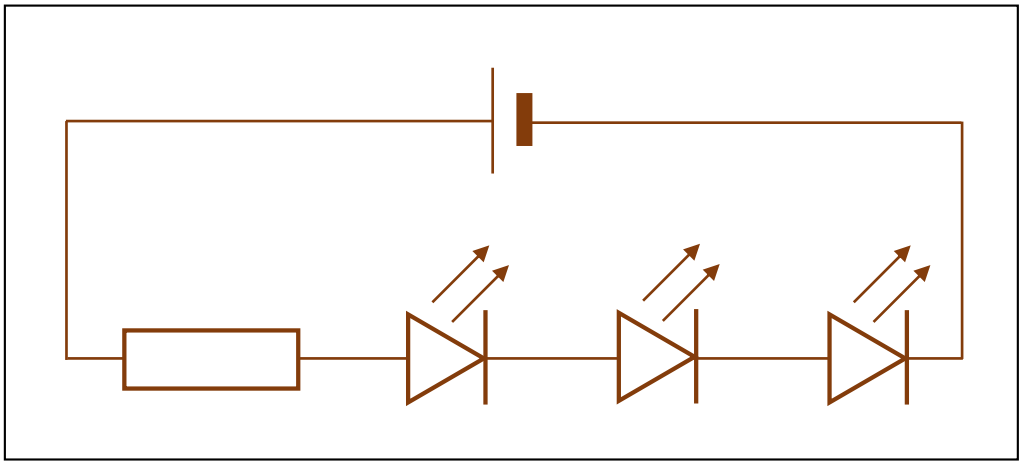
Het automatisch aan laten gaan van een buitenlamp als het buiten donker wordt.

2.7 LEDs op de Arduino – Serieschakeling

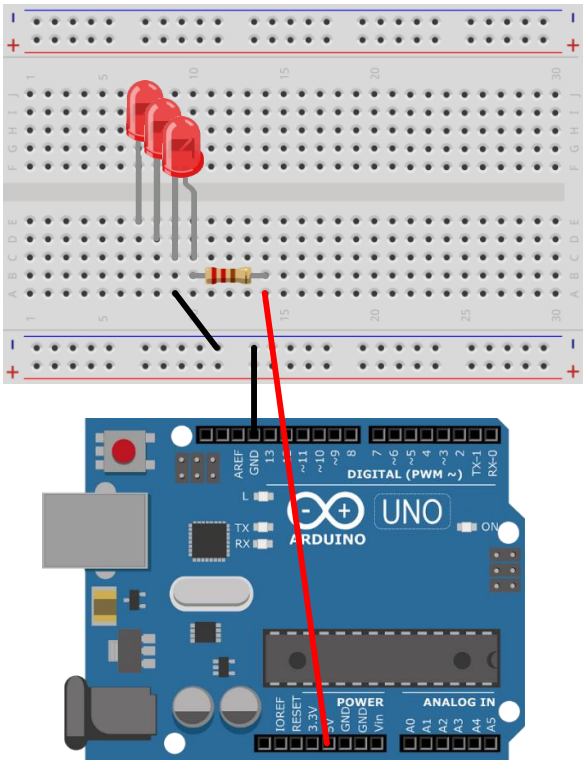
Wat gebeurt er als je meerdere LEDs aansluit op de Arduino? En maakt het dan nog uit of je de LEDs aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *serieschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 LEDs in *serie* aansluiten op de Arduino.

Opgave 34. Teken het schakelschema voor een stroomkring met een voorschakelweerstand in serie met 3 LEDs aangesloten op een spanningsbron.



Opgave 35. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 36. In de schakeling van opgave 35 staat nu slechts 1 LED in serie in de stroomkring. De overige 2 LEDs doen nu niks. Verander de aansluiting van de bovenste zwarte jumper op het breadboard zodanig dat je 2 LEDs in serie hebt staan in de stroomkring. Doe daarna hetzelfde voor 3 LEDs. Wat neem je waar? Geef een toelichting.

De LEDs branden steeds minder fel naarmate er meer LEDs in serie in de stroomkring worden opgenomen. Bij het aansluiten van de 3e LED brandt geen van de LEDs meer!

Toelichting: in een serieschakeling verdeelt de spanning zich over de componenten. Iedere rode LED heeft een minimale spanning van 1,9 V nodig om een stroom te kunnen laten lopen, de zogenaamde doorlaatspanning. Bij het aansluiten van 3 LEDs van 1,9 V, opgeteld 5,7 V, is de door de Arduino geleverde 5 V niet meer voldoende om iedere LED te kunnen voorzien van de benodigde doorlaatspanning. Hierdoor gaat door de LEDs geen stroom meer lopen en zullen de LEDs niet branden.

Opgave 37. Bereken de stroomsterkte door het aantal LEDs in serie in de stroomkring waarbij de LEDs nog wel branden.

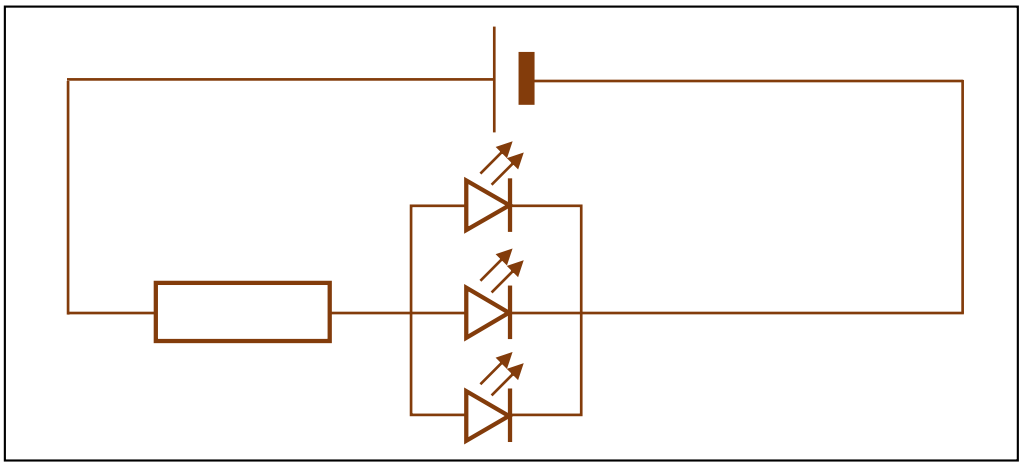
Gegeven:	$U_{LED} = 1{,}9\ V$ $U_{tot} = 5{,}0\ V$ $R = 220\ \Omega$
Gevraagd:	$I = ?$
Uitwerking:	$U_{tot} = U_{LED} + U_{LED} + U_R$ $5{,}0 = 1{,}9 + 1{,}9 + U_R$ $U_R = 1{,}2\ V$ $I = \frac{U_R}{R} = \frac{1{,}2}{220} = 5{,}5\ mA$

2.8 LEDs op de Arduino – Parallelschakeling

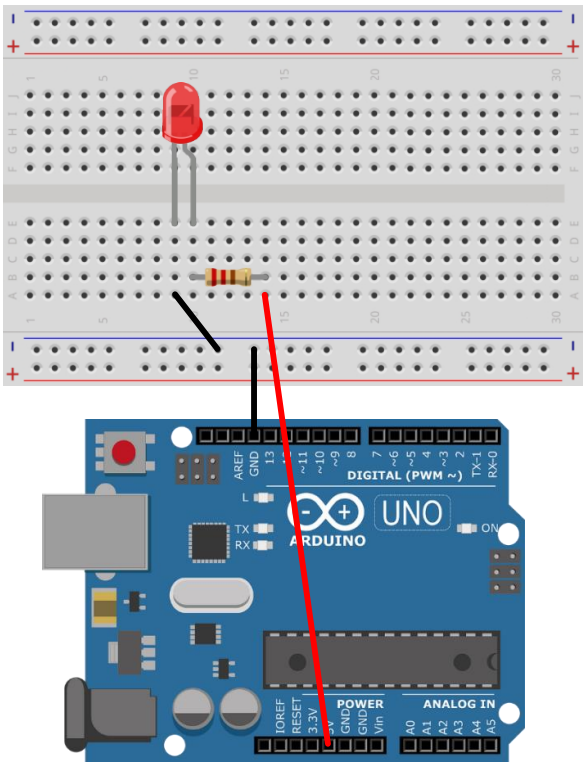
Wat gebeurt er als je meerdere LEDs aansluit op de Arduino? En maakt het dan nog uit of je de LEDs aansluit via een serie- of parallelschakeling? In dit onderdeel ga je antwoord geven op de vragen met betrekking tot de *parallelschakeling*.

➢ Je gaat nu achtereenvolgens 1, 2, en 3 LEDs in *parallel* aansluiten op de Arduino.

Opgave 38. Teken het schakelschema voor een stroomkring met een voorschakelweerstand in (**let op!**) serie met 3 parallel geschakelde LEDs aangesloten op een spanningsbron.



Opgave 39. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 40. In de schakeling van opgave 39 staat nu 1 LED in serie met de voorschakelweerstand in de stroomkring. Voeg nu eenzelfde LED toe op het breadboard zodat beide LEDs ten opzichte van elkaar in parallel staan, maar samen nog wel in serie staan met de voorschakelweerstand. Doe daarna hetzelfde voor 3 LEDs.
Wat neem je waar? Geef een toelichting.

De LEDs branden steeds minder fel naarmate er meer LEDs in parallel in de stroomkring worden opgenomen. Dit is niet heel sterk merkbaar.

Toelichting: in een parallelschakeling is de spanning hetzelfde voor de componenten. Iedere rode LED heeft een minimale spanning van 1,9 V nodig om een stroom te kunnen laten lopen, de zogenaamde doorlaatspanning. In deze situatie is de door de Arduino aan iedere LED geleverde 5 V dus voldoende om te voorzien in de benodigde doorlaat-spanning. De stroomsterkte in een parallelschakeling wordt echter wel verdeeld. In deze situatie is de stroomsterkte nog wel voldoende groot om alle 3 LEDs te laten branden.

Opgave 41. Bereken de stroomsterkte door de LEDs in parallel in de stroomkring.

Gegeven: $U_{LED} = 1{,}9\ V$; $U_{tot} = 5{,}0\ V$; $R = 220\ \Omega$

Gevraagd: $I_{LED} = ?$

Uitwerking:

$U_{tot} = U_{LED} + U_R$

$5{,}0 = 1{,}9 + U_R$

$U_R = 3{,}1\ V$

$$I_{tot} = \frac{U_R}{R_{tot}} = \frac{3{,}1}{220} = 14\ mA$$

Echter, 3 gelijke LEDs in parallel:

$I_{tot} = I_1 + I_2 + I_3$

$I_1 = I_2 = I_3 = \frac{I_{tot}}{3} = \frac{14}{3} = 4{,}7\ mA$

Indien je zowel §2.7 als §2.8 hebt gemaakt:

Opgave 42. Branden de LEDs het felst in serie- of parallelschakeling?
Geef een toelichting.

Parallelschakeling.

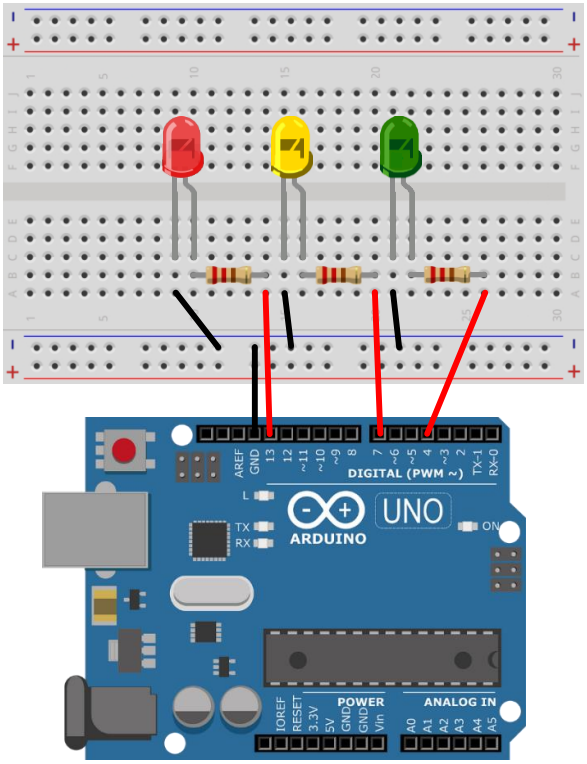
Toelichting: de stroomsterkte wordt in de parallelschakeling weliswaar verdeeld, maar is in deze situatie nog voldoende groot om de 3 LEDs te kunnen laten branden. In de serieschakeling is de geleverde 5 V spanning door de Arduino echter onvoldoende om de 3 LEDs te voorzien in hun benodigde doorlaatspanning, waardoor er geen stroom meer loopt en de 3 LEDs niet branden.

2.9 LEDs afwisselen op de Arduino

Het is mogelijk om LEDs om beurten te laten branden door deze via aparte pinnen (ingangen) aan te sluiten op de Arduino en hiervoor een programma te schrijven. In dit onderdeel ga je dit uitproberen.

➢ Je gaat nu 3 verschillende LEDs achtereenvolgens laten branden.

Opgave 43. Bouw onderstaand schakelschema na. **Let op:** zet de Arduino tijdelijk uit.



Opgave 44. Voer de volgende stappen uit:

a. Sluit de Arduino via de USB-kabel aan op de USB-poort van de laptop.

b. Open de Arduino IDE via de snelkoppeling op het bureaublad.

c. Open de voorbeeld-sketch genaamd *Blink*

Bestand → Voorbeelden → 01.Basics → Blink

Let op: de sketch opent zich in een nieuw venster. Het oude venster mag je sluiten.

d. Omdat deze sketch nog niet doet wat je wil moet je deze aanpassen:

- Eerst moet je de Arduino vertellen welke pinnen je gaat gebruiken en of deze pinnen een ingang (INPUT) of uitgang (OUTPUT) zijn (zie voor uitleg figuur 17). Dit doe je bij void setup(). Je vertelt de Arduino hier dat pin 4, 7, en 13 gebruikt zullen worden, en dat deze zullen dienen als uitgang (OUTPUT). Hierdoor kun je van deze pinnen dan een 5 V uitgangsspanning ontvangen voor de LEDs.
- Vervolgens vertel je de Arduino welke taken hij moet uitvoeren.
  Dit doe je bij void loop(). Je vertelt de Arduino hier welke handelingen hij stapsgewijs moet uitvoeren, zoals het afgeven van een hoge spanning (5 V) aan pin 4 via de regel digitalWrite(4, HIGH); of het nemen van 1 seconde pauze via de regel delay (1000); (zie voor uitleg figuur 17).
- Neem onderstaande regels over in je sketch:

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(4, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(4, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(4, LOW);     // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second

  digitalWrite(7, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(7, LOW);     // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second

  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

e. Klik linksboven op de knop *Verifieer* Wacht op de melding *Compileren voltooid*.

f. Klik linksboven op de knop *Upload* Wacht op de melding *Uploaden voltooid*.

g. Klaar! De LEDs op de Arduino zullen nu achtereenvolgens gaan branden.

3 PROJECTEN MET ARDUINO

Je gaat nu een mini-project doen met Arduino die bestaat uit een situatieschets die je met Arduino moet gaan realiseren. De projecten zijn verdeeld over twee niveaus: beginner en expert.

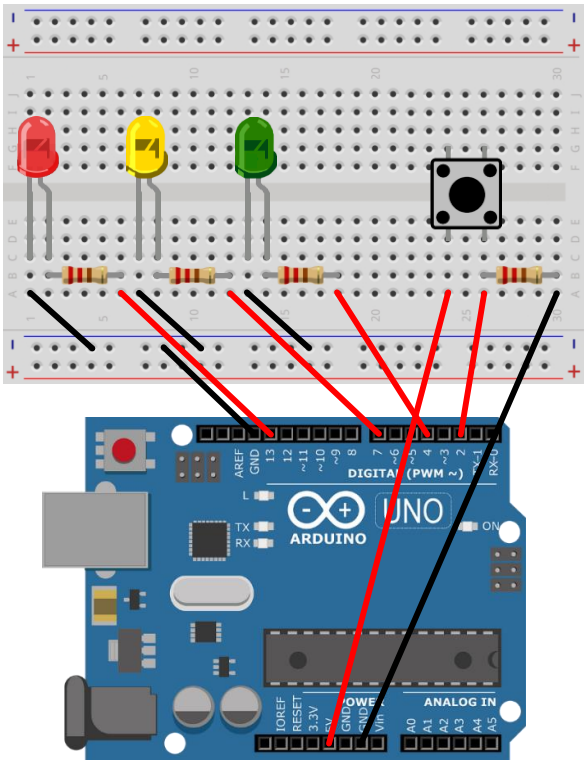
➢ Je gaat nu achtereenvolgens onderstaande projecten maken.

3.1 VOETGANGERSOVERSTEEKPLAATS – BEGINNER

Aan een drukke weg is voor voetgangers een oversteekplaats gemaakt. Als er geen voetgangers zijn staat het verkeerslicht voor auto's op groen. Voetgangers die willen oversteken kunnen op een knop drukken. Het verkeerslicht voor auto's gaat dan 1 seconde later stapsgewijs van groen, naar oranje, naar rood. Hierbij brandt het oranje licht 3 seconden lang. Het rode verkeerslicht gaat na 10 seconden te hebben gebrand weer op groen zodat de auto's weer mogen rijden.

Opgave 45. Maak de bovenstaande situatieschets met Arduino. **Let op:** zet de Arduino tijdelijk uit.

Zie onderstaand schakelschema + sketch.



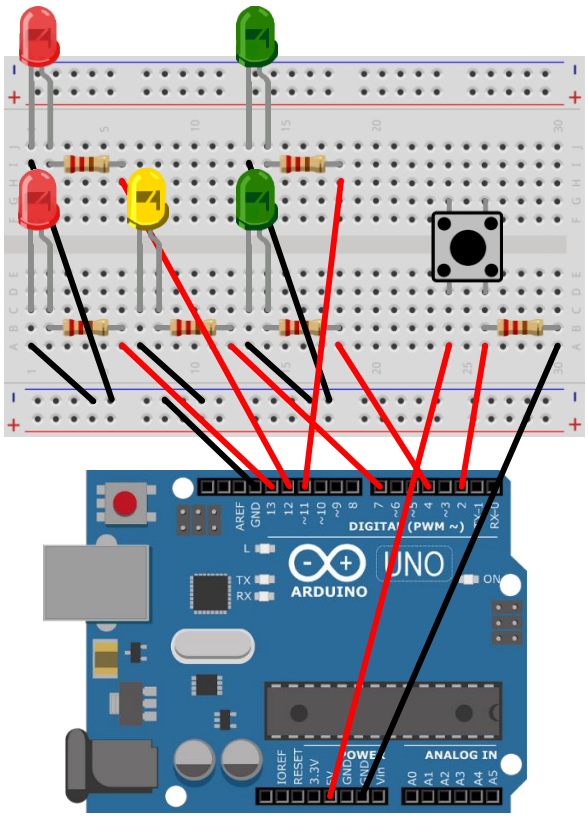
```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;         // the number of the pushbutton pin
const int ledPinGroen = 4;       // the number of the LED pin Groen
const int ledPinOranje = 7;      // the number of the LED pin Oranje
const int ledPinRood = 13;       // the number of the LED pin Rood

// variables will change:
int buttonState = 0;             // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin Groen as an output:
  pinMode(ledPinGroen, OUTPUT);
  // initialize the LED pin Oranje as an output:
  pinMode(ledPinOranje, OUTPUT);
  // initialize the LED pin Rood as an output:
  pinMode(ledPinRood, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  digitalWrite(ledPinGroen, HIGH);       // turn the LED on (HIGH is the voltage level)

  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    delay(1000);                         // wait for 1 second
    digitalWrite(ledPinGroen, LOW);      // turn the LED off by making the voltage LOW

    digitalWrite(ledPinOranje, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(3000);                         // wait for 3 seconds
    digitalWrite(ledPinOranje, LOW);     // turn the LED off by making the voltage LOW

    digitalWrite(ledPinRood, HIGH);      // turn the LED on (HIGH is the voltage level)
    delay(10000);                        // wait for 10 seconds
    digitalWrite(ledPinRood, LOW);       // turn the LED off by making the voltage LOW
  }
}
```

3.2 VOETGANGERSOVERSTEEKPLAATS – EXPERT

Aan een drukke weg is voor voetgangers een oversteekplaats gemaakt. Als er geen voetgangers zijn staat het verkeerslicht voor auto's op groen en het verkeerslicht voor voetgangers op rood. Voetgangers die willen oversteken kunnen op een knop drukken. Het verkeerslicht voor auto's gaat dan 1 seconde later van groen, naar oranje, naar rood. Hierbij brandt het oranje licht 3 seconden lang. Als het verkeerslicht voor de auto's op rood staat gaat 2 seconden later het verkeerslicht voor de voetgangers op groen. Voetgangers hebben dan 10 seconden de tijd om over te steken. Hierna knippert het voetgangerslicht 5 keer waarna het op rood springt. Dit knipperen gaat afwisselend: 0,5 seconden uit, 0,5 seconden aan. Vervolgens gaat 3 seconden later het verkeerslicht voor de auto's weer op groen.

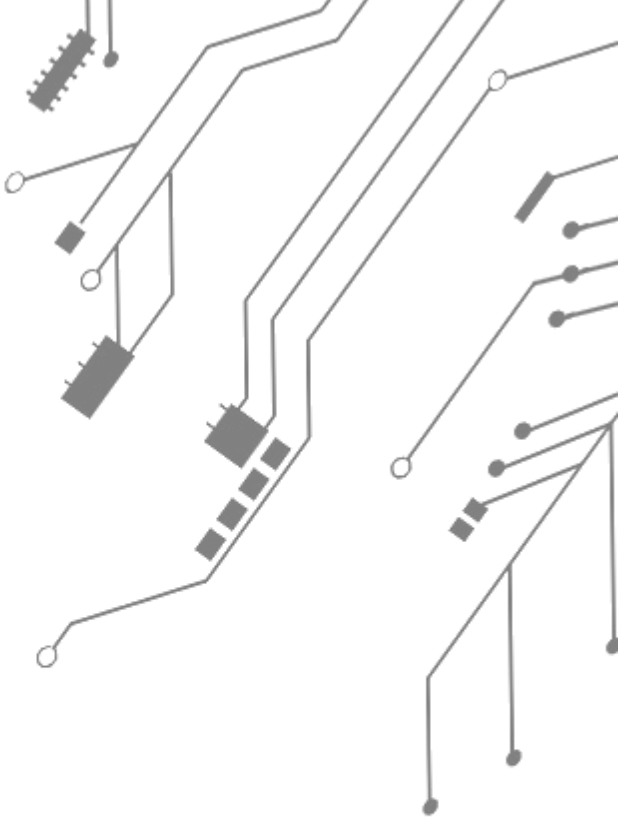
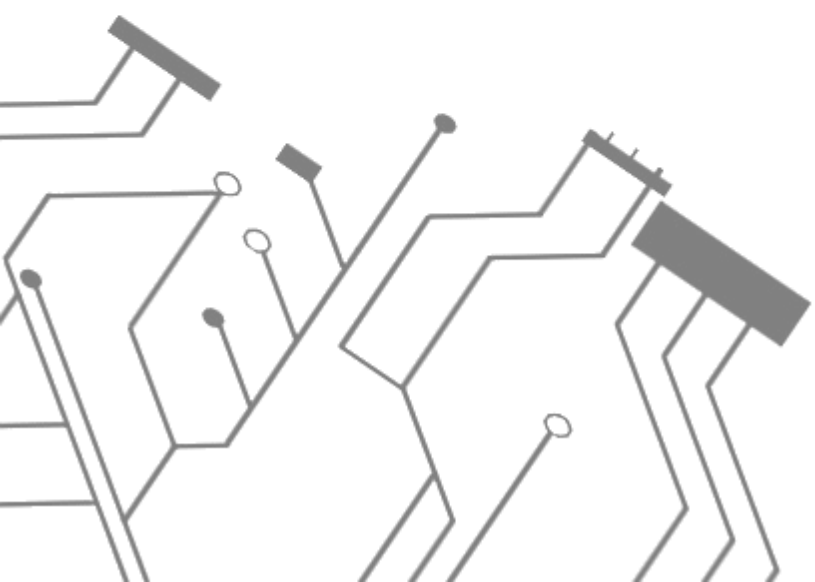
Opgave 46. Maak de bovenstaande situatieschets met Arduino. **Let op:** zet de Arduino tijdelijk uit.

Zie onderstaand schakelschema + sketch.



```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;                  // the number of the pushbutton pin
const int ledPinAutoGroen = 4;            // the number of the LED pin Auto Groen
const int ledPinAutoOranje = 7;           // the number of the LED pin Auto Oranje
const int ledPinAutoRood = 13;            // the number of the LED pin Auto Rood
const int ledPinVoetgangerGroen = 11;     // the number of the LED pin Voetganger Groen
const int ledPinVoetgangerRood = 12;      // the number of the LED pin Voetganger Rood

// variables will change:
int buttonState = 0;                      // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin Auto Groen as an output:
  pinMode(ledPinAutoGroen, OUTPUT);
  // initialize the LED pin Auto Oranje as an output:
  pinMode(ledPinAutoOranje, OUTPUT);
  // initialize the LED pin Auto Rood as an output:
  pinMode(ledPinAutoRood, OUTPUT);
  // initialize the LED pin Voetganger Groen as an output:
  pinMode(ledPinVoetgangerGroen, OUTPUT);
  // initialize the LED pin Voetganger Rood as an output:
  pinMode(ledPinVoetgangerRood, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  digitalWrite(ledPinAutoGroen, HIGH);            // turn the LED on (HIGH is the voltage level)
  digitalWrite(ledPinVoetgangerRood, HIGH);       // turn the LED on (HIGH is the voltage level)

  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    delay(1000);                                  // wait for 1 second
    digitalWrite(ledPinAutoGroen, LOW);           // turn the LED off by making the voltage LOW

    digitalWrite(ledPinAutoOranje, HIGH);         // turn the LED on (HIGH is the voltage level)
    delay(3000);                                  // wait for 3 seconds
    digitalWrite(ledPinAutoOranje, LOW);          // turn the LED off by making the voltage LOW

    digitalWrite(ledPinAutoRood, HIGH);           // turn the LED on (HIGH is the voltage level)

    delay(2000);                                  // wait for 2 seconds
    digitalWrite(ledPinVoetgangerRood, LOW);      // turn the LED off by making the voltage LOW
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)

    delay(10000);                                 // wait for 10 seconds
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(500);                                   // wait for 0.5 second
```

```
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(500);                                   // wait for 0.5 second
    digitalWrite(ledPinVoetgangerGroen, LOW);     // turn the LED off by making the voltage LOW
    digitalWrite(ledPinVoetgangerRood, HIGH);     // turn the LED on (HIGH is the voltage level)

    delay(3000);                                  // wait for 3 seconds
    digitalWrite(ledPinAutoRood, LOW);            // turn the LED off by making the voltage LOW
    digitalWrite(ledPinAutoGroen, HIGH);          // turn the LED on (HIGH is the voltage level)
  }
}
```

Kom je er niet uit, gebruik dan onderstaande tips of vraag om hulp:

- Zie §2.3 hoe je het indrukken van de drukknop kunt registreren en gebruiken.
- Zie §2.9 hoe je een stoplicht kunt nabootsen.
- Begin met het programma van §2.3 en verwerk hierin het programma van §2.9.
- Expert: voeg de benodigde hardware en software voor het voetgangerslicht zelf toe.