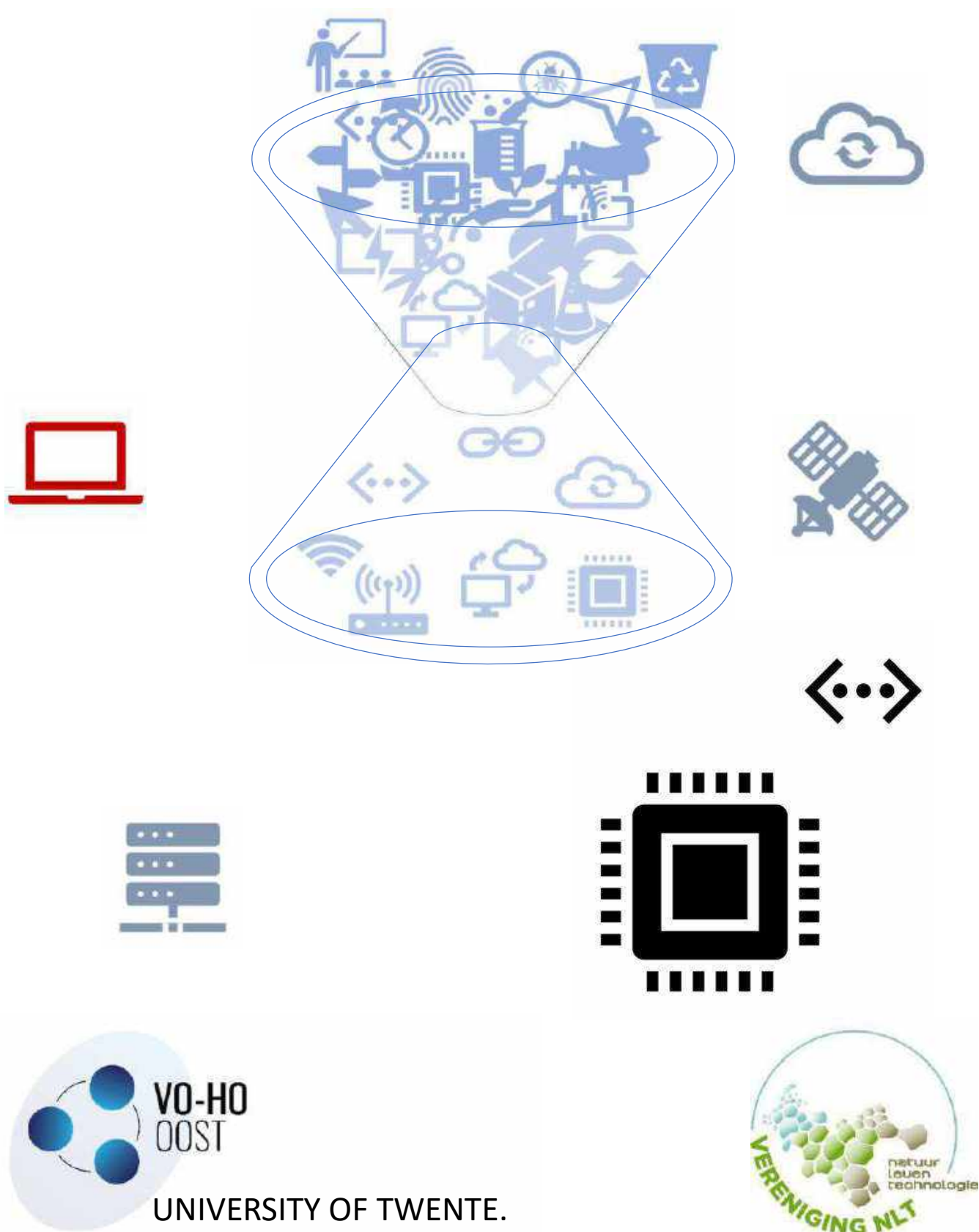


Digitale Technologie

schakelmodule



Inhoudsopgave

COLOFON	4
WOORD VOORAF	5
1 DIGITALE TECHNOLOGIE IS TEGENWOORDIG OVERAL	6
1.1 Scrum	12
2 INTRODUCTIE OP DE MODULE DIGITALE TECHNOLOGIE	14
2.1 Basisbegrippen	15
1. Waarnemen (hoofdstuk 3)	16
2. Verwerken en beslissen (hoofdstuk 4)	18
3. Datatransport en dataopslag (hoofdstuk 5)	19
4. Uitvoeren (hoofdstuk 6)	19
5. Feedback (geen eigen hoofdstuk)	20
6. Verbindingen (hoofdstuk 5)	20
2.2 Basis van de hardware ontdekken met Arduino, Micro:bit of Raspberry pi	24
3. WAARNEMEN EN SENSOREN	29
3.1 De omgeving waarnemen	30
3.2 Kalibreren en ijken	31
3.3 Betekenis van sensoren in een context: het sporthorloge	33
3.4 Onderzoeken en maken	37
4 VERWERKEN EN BESLISSSEN	38
4.1 Fabricage van electronica	39
4.2 Programmeren van een chip	42
4.3 Artificial Intelligence (AI)	46
4.4 Maken en programmeren: digitale thermometer met Arduino	48
5 DATA OPSLAG EN TRANSPORT	51
5.1 De 'cloud'	52
5.2 Verbindingen	54

5.3 Adresseren	55
5.4 Datacenters	57
5.5 Datacenters en 'Cloud'	59
5.6 Beveiliging	59
5.7 Keuzeopdrachten	62
a. Kleine arduino's maken samen big data	62
b. Je eigen Arduino meetproject.	62
c. Satellieten en informatie over de aarde (GIS)	63
d. Verder te verkennen / te onderzoeken toepassingen	63
6 UITVOER: DE ACTUATOR	64
6.1 Invloed op de omgeving uitoefenen	65
6.2 Mechatronica in bedrijf	69
6.3 Maakopdracht met Micro:bit, Arduino of Raspberry Pi.	70
7 EINDOPDRACHT: DIGITAAL DEVICE ONTWERPEN	71
Projecten	71
7.1 Digitale technologie in huis: weerstation voor het binnenklimaat.	71
7.2 Wetropolis: sensoren en actuatoren voor waterbeweging	73
7.3 Digitale devices die een oplossing zijn bij opwarming in de stad	74
7.4 De Q-strip: meten van transpiratievocht	76
7.5 Andere thema's voor de eindopdracht	82
8 VERDER MET DIGITALE TECHNOLOGIE	83
Fijnstof	83
Modelleren	83
Bio-informatica	83
De toekomst van de landbouw	84
Cybersecurity	84
Artificiële Intelligentie (AI)	84
Leeg analyseschema	85
Bijlagen verkrijgbaar in de moduledatabase	85

Colofon

Deze module is bedoeld voor de lessen nlt op havo en vwo. In een later stadium zal voor beide schooltypen een eigen versie beschikbaar komen.

Auteurs

- Klaas Beute (Greijdanus, Zwolle)
- Rudy Jonker (TCC de Thij, Oldenzaal)
- Bart Kappé (Veenlanden College, Mijdrecht)
- Aernout van Rossum (Pre-U, Universiteit Twente; Thomas a Kempis College, Arnhem)
- Jan Jaap Wietsma (Pre-U, Universiteit Twente, nlt vaksteunpunt Oost; Greijdanus, Zwolle en projectleiding)
- Met ondersteuning van Edgar de Wit (Vereniging NLT)

Partners

Deze schakelmodule is ontwikkeld door Pre-U, Universiteit Twente, in opdracht van Vereniging NLT, met financiële en inhoudelijke ondersteuning van

Platform Talent voor Technologie
NLdigital
Dutch Datacenter Association
ISP Connect/ DHPA
CGI
IBM
Human Capital Agenda ICT
Nederlandse AI Coalitie

WisMon
Stichting Globe
Stichting Leerplan Ontwikkeling (SLO)
Stichting Technasium
Vakvereniging I&I
Bèta-outreachfonds van de bèta-decanen van
de gezamenlijke universiteiten

Copyrights

De auteursrechten van dit werk liggen bij Vereniging NLT.

© 2023 Wijk bij Duurstede, Schakelmodule Digitale Technologie, redactieversie 1.0, januari 2023.

Certificeringsnummer 6141-100-VE2-2F2

Bij deze module is online materiaal beschikbaar via

https://maken.wikiwijs.nl/171772/Schakelmodule_Digitale_Technologie_online_materiaal

De bijlagen, waarnaar verwezen wordt in deze module, zijn in deze gedrukte achterwege gelaten.

Alle materialen zijn beschikbaar in de Moduledatabase van Vereniging NLT (<https://module-database.betavak-nlt.nl/>).

Ondersteuning voor deze module wordt gegeven door nlt vaksteunpunt Oost (nlt@beta-oost.nl).

De auteurs hebben zo goed mogelijk de bronnen vermeld en gebruik gemaakt van rechtenvrije afbeeldingen. Mochten er desondanks personen of instanties zijn die rechten menen te kunnen doen gelden op tekstgedeeltes, illustraties enz. van deze module, dan worden zij verzocht zich in verbinding te stellen met nlt vaksteunpunt Oost via het genoemde e-mailadres.

Woord vooraf

Alsjeblieft, de schakelmodule Digitale Technologie.

Ook zonder dat je lid bent van de Vereniging NLT mag je de leerlingversie vrijuit gebruiken, maar we willen wel graag toelichten wat nlt is en wat Vereniging NLT doet.

Wat is nlt

Nlt staat voor Natuur, Leven en Technologie en is een keuzevak voor de bovenbouw havo en vwo. Er is steeds vaker een combinatie van verschillende disciplines nodig om complexe vraagstukken in de wereld op te lossen. Het vak nlt laat zien op welke manier de vakken aardrijkskunde, biologie, informatica, natuurkunde, scheikunde en wiskunde samenkomen om aan deze complexe opdrachten te werken. Het vak is bedoeld als voorbereiding op de keuze voor een studie op het gebied van bèta en technologie.

Het vak nlt is gebaseerd op talrijke modules over zeer verschillende onderwerpen. Het is aan de school om hier keuzes in te maken, waarbij de enige voorwaarde is dat de eindtermen uit het examenprogramma aan bod komen. De modules worden ontwikkeld door docenten in samenwerking met het hoger onderwijs. Daardoor zijn de modules optimaal afgestemd op ontwikkelingen in maatschappij, wetenschap en technologie.

Wat doet Vereniging NLT

Scholen die het vak nlt aanbieden zijn lid van Vereniging NLT. De Vereniging telt ruim 220 leden. Zij zorgt voor de ontwikkeling van nieuwe modules over actuele onderwerpen. Deze schakelmodule over duurzame ontwikkeling is daar een mooi voorbeeld van.

Ook zorgt de Vereniging voor het onderhoud van bestaande modules door ze elke vijf jaar aan te passen aan de nieuwste ontwikkelingen. Kijk voor het huidige moduleaanbod op de website.

Naast modules werkt de Vereniging aan het professionaliseren van docenten en TOA's door middel van conferenties en bijeenkomsten op landelijk en regionaal niveau. Ook nieuwe ontwikkelingen worden opgepakt. Zo werkt de Vereniging actief mee aan het actualiseren van het examenprogramma en het opzetten van nlt in de onderbouw.

Kortom, nlt is een dynamisch vak dat past in het onderwijs van nu en in de toekomst. Kijk voor meer informatie over nlt op www.verenigingnlt.nl. Hier vind je naast een overzicht van alle beschikbare modules, ook informatie over bijv. de activiteiten van de Vereniging en de regionale vaksteunpunten, leerlijnen, toetsing en over het invoeren van nlt.

Deze schakelmodule is ontwikkeld door een auteursteam van het vaksteunpunt NLT - Pre-U, Universiteit Twente. De module vormt de start van een leerlijn digitale technologie, ontwikkeld door Vereniging NLT. In het laatste hoofdstuk staan korte beschrijvingen van modules die in deze leerlijn passen. Vereniging NLT en het auteursteam wensen je veel leerplezier bij het gebruik van deze module.

1 Digitale technologie is tegenwoordig overal

Leerdoelen

Kennis

- 1.1 Bewust worden van de betekenis van digitale technologie in je alledaagse omgeving*
- 1.2 Herkennen van digitale technologie in een context*
- 1.3 Oriënteren op digitale technologie in je omgeving, waardoor je een goed beeld krijgt van beroepen en gebruik.*

Vaardigheden

- 1.4 Overzicht krijgen van de vaardigheden die nodig zijn, en welke je al hebt, voor het gebruik van digitale technologie.*

Digitale technologie is zó gewoon en onmisbaar geworden voor ons dagelijks leven, dat we er amper meer bij stil staan. Computers, navigatie, mobieltjes, online video, streaming, appen, bellen, betalen met de pin: we kunnen niet meer zonder. Het is moeilijk om je een wereld voor te stellen zonder die technologie. Dat betekent ook: in welke richting je ook gaat studeren en werken, je ontkomt niet meer aan digitale technologie. Daarom is het goed om een idee te krijgen wat het is, hoe het werkt, blijft werken en beveiligd wordt, en welke (vaak onzichtbare) invloed deze technologie op ons leven heeft. Om je op weg te helpen hoe digitale technologie gebruikt wordt, beginnen we met een paar voorbeelden.

De tandarts

In de afgelopen tijd ben je vast een keer bij de tandarts geweest. Heb je toen gemerkt hoe de tandarts digitale technologie gebruikt? Waarschijnlijk niet zo bewust. De afspraak is in een digitale agenda vastgelegd. Mogelijk heb je een sms of mailtje ontvangen dat je de afspraak niet moet vergeten. De tandartsstoel wordt met knopjes keurig ingesteld en de tandarts heeft jouw gegevens bij de hand. Uiteraard op een scherm waarin precies te zien is wat er met jouw gebit gebeurd is in de afgelopen jaren, inclusief de röntgenfoto's. Nieuwe foto nodig? Jij klemt een plaatje met een draad tussen je tanden en krijgt een apparaat naast je wang. De tandarts stapt even weg, en bekijkt enkele seconden later de foto op het scherm en kan direct bepalen of er nog iets aan je gebit gebeuren moet. Opzienbarend? Misschien niet, maar wel een groot verschil met dertig jaar geleden. Een afspraak had je op een kaartje, dat je thuis moest bewaren. Met wat geluk werd je (op het vaste thuisnummer) gebeld als herinnering om te komen, maar dat was voor de uitzonderingen. De tandarts had een papieren kaart met daarop een schema van jouw gebit. Tijdens het onderzoek kreeg (net als nu) de assistent(e) gecodeerd informatie waar eventueel een gaatje zat. Dat werd op de kaart aangetekend, en bijgewerkt als het gerepareerd werd. Een foto werd gemaakt met een ingepakt lichtgevoelig negatief. Als je de volgende keer



Figuur 1.1 Digitale foto bij de tandarts

https://www.google.com/url?sa=i&url=https%3A%2Fdepositphotos.com%2Fstock-photos%2Fdental-xray.html&psig=AOvVaw0lt8qXaRphhAsq_hQ4E2r&ust=1623509026466000&source=images&cd=vfe&ved=0CA0QjhqFwoTCJCF_tHoJ_ECFQAAAAAABBD

terugkwam was deze foto ontwikkeld en vastgeniet aan je kaart. In elk geval: het kostte allemaal een stuk meer tijd.

Om een beugel te zetten, was het nodig dat er een gipsafdruk van je kaak gemaakt werd. De orthodontist moest met die afdruk bepalen waar blokjes en draadjes moesten komen. Ook heel tijdrovend. De huidige orthodontist gebruikt steeds vaker digitale 3D gebitsfoto's. Met de computer wordt berekend hoe de tanden moeten verschuiven en hoe de beugel moet worden. Dat wordt uiteraard alvast in een mooi plaatje gezet, zodat je kunt zien hoe het gaat worden. De computer berekent dan precies welke onderdelen voor de beugel nodig zijn, zodat die precies op maat zijn en klaarliggen als de beugel geplaatst gaat worden. Daar komt steeds minder handwerk of berekening aan te pas.

1.1 Opdracht

Noteer aan de hand van de tekst op welke manier digitale technologie bij de tandarts en orthodontist gebruikt wordt. Vul dit aan met tenminste één aspect dat niet in de tekst genoemd staat. Noteer er naast hoe je dit zou doet zonder digitale technologie.

De gegevens die de tandarts verzamelt over jou, en van al zijn/haar andere patiënten, kunnen helpen om de mondzorg te verbeteren. Er zijn duizenden tandartsen die dit doen. Al die gegevens (*data*) vormen een rijke bron van informatie. Als die gegevens via de computer zijn opgeslagen, kun je ze bewerken en doorsturen.

Wie die tot zijn beschikking heeft, kan bijvoorbeeld onderzoeken in welke regio's van het land veel gebitsproblemen zijn, welke behandeling tandartsen inzetten of hoe vaak patiënten de tandarts bezoeken. Je kunt zelf vast nog meer onderzoeksvragen bedenken.

1.2 Vraag

Welke manieren ken je om computers en computergegevens (data) te beschermen?

Beveiliging en privacy

Zodra meer mensen computerdata kunnen bekijken is privacy belangrijk. Je wilt liever niet dat jouw persoonlijke gegevens (bijvoorbeeld over je gebit) zomaar door anderen bekeken kunnen worden.

Het verzamelen en gebruiken van digitale gegevens in de zorg luistert nauw. Wie mag ze inzien, waar is toestemming van de patiënt nodig?

Inloggen, veilige verbindingen, onleesbaar maken van gegevens als ze zonder sleutel bekeken worden (encryptie) en bijhouden wie er in de data kijkt: er is veel nodig om privacy te garanderen.

Landbouw



Figuur 1.2 Drone voor het meten aan landbouwgewassen

<https://agriculturepost.com/7-benefits-of-remote-sensing-gis-in-agriculture/>

Op een gemengd boerenbedrijf worden graan, maïs en bieten geteeld en is een stal met 100 koeien. In het voorjaar bewerkt de boer de akker en zaait de gewassen in. Er wordt waar nodig gewasbescherming gespoten en beregend en gekeken of het gewas zich goed ontwikkelt. Allemaal met het oog. De koeien worden met de melkmachine gemolken, tweemaal per dag. In de zomerperiode zijn de koeien in de wei en worden ze daar gemolken. De melk gaat in melkbussen op de kar mee naar de boerderij. Technologie, en nu ook digitale technologie, heeft dit veranderd en die ontwikkeling gaat nog wel een poosje door.



Figuur 1.3 Een melkrobot

<https://www.boerenbusiness.nl/agribusiness/artikel/10878173/lily-astronaut-a5-goedkoper-en-eenvoudiger>

De koeien worden gemolken met de melkrobot, krijgen hun rantsoen voer doordat ze herkend worden aan een zender om hun nek. De boer bepaalt welk deel van de akker bemest moet worden door satellietgegevens en dronefoto's te gebruiken. Als er geoogst wordt, houdt de oogstmachine precies bij hoeveel elk stukje van de akker opgebracht heeft. Hiermee kan bepaald worden wat er aan de grond moet gebeuren voor het volgende seizoen. De komende jaren zal de landbouw nog meer veranderen, mede door klimaatverandering en eisen rond duurzaamheid en milieu.

Opsporing



Figuur 1.4 Forensisch onderzoek aan digitale opslagmedia
<https://blog.veriato.com/the-evolution-of-digital-forensics>

De politie maakt op uitgebreide schaal gebruik van digitale technologie. Vingerafdrukken of DNA-sporen van een plaats-delict komen in digitale databases. Gegevens uit telefoons of computers van slachtoffers en verdachten leveren belangrijk bewijsmateriaal. Slimme camera's detecteren hardrijders, mensen in de stad of kentekens van gezochte auto's. Een nieuw onderzoekproject gaat van start om deep-fake videobeelden te kunnen opsporen (dit zijn bedrieglijk-echt lijkende gemonteerde beelden, waarin het lijkt dat mensen allerlei dingen doen of zeggen, die ze in werkelijkheid nooit gedaan of gezegd hebben). Het maken van dit soort nepbeelden is steeds gemakkelijker en het resultaat is steeds moeilijker van echt te onderscheiden. Speciale opsporingssoftware moet helpen deze fake te ontmaskeren (*NOS journaal*, 24-5-2021).

De huisarts

Tijdens de Covid-19 pandemie werd de temperatuur van mensen op afstand gemeten om te ontdekken of ze Corona hebben. Als je de huisarts belt, is één van de vragen die je aan de telefoon krijgt: heb je koorts? Om dat zelf te meten, is de kans groot dat je gebruikt maakt van een digitale koortsthermometer. Verderop in de module gaan we die beter bekijken.

1.3 Opdracht

Kies één van de genoemde contexten. Bedenk welke rol digitale technologie nog meer kan hebben in die situatie. Noteer tenminste drie toepassingen.

Big data

Over de hele wereld staan digitale apparaten die informatie verzamelen. Camera's, detectielussen, thermometers, wifitrackers, enzovoorts. Die informatie blijft bewaard in databestanden op computers. Via het internet zijn die gekoppeld. Samen vormen ze de 'big data'. De politie kan er misdrijven mee oplossen, bedrijven vinden zo hun klanten.

Het wordt minder plezierig als de data door criminelen gebruikt wordt om mensen op te lichten, of door de overheid om burgers te volgen of te dwarsbomen vanwege hun overtuiging. Niet voor niets zijn privacy en beveiliging extreem belangrijk.



Figuur 1.5 Digitale koortsthermometer

Svdmolen - Self-published work by Svdmolen, CC BY 2.5,
<https://commons.wikimedia.org/w/index.php?curid=1528736>

1.4 Opdracht: Zelfstandig werkend apparaat – systeem – communicatie – dataopslag

Apparaten of voorwerpen die je misschien zelf gebruikt

Wekker	Electrische tandenborstel	OV-chipkaart
Radio	Waterkoker	Smartwatch
Televisie	Telefoon	Sporthorloge
Computer	Rekenmachine	Electrische fiets
Smartphone	Koortsthermometer	<i>Ander voorbeeld</i>
Beeldscherm	Zelflerende thermostaat	
Electrische lamp	Bankpasje	

Kies twee items uit bovenstaand lijstje. Maak voor beide de volgende opdracht:

- Wat doet of kan het apparaat? Wat doe je er allemaal mee?
- Communiqueert het ding met de omgeving?
- Heeft het apparaat sensoren (ontvangers)?
- Hoe weet je wat het apparaat doet? Bijvoorbeeld: zijn er lampjes, schermje(s), bliepjes via een luidsprekertje?
- Hoe komt het apparaat aan energie?
- Heeft het apparaat andere apparaten nodig om te kunnen werken?
- Welke informatie wisselt het apparaat uit en waarmee?
- Is er in het apparaat privacy-gevoelige informatie aanwezig? Is die beveiligd en hoe?
- Hoe kom je er achter waar de informatie uit het apparaat blijft?
- Slaat het apparaat data op, en zo ja, wat voor soort data?

1.5 Opdracht: Paspoort digitale kennis en vaardigheden

Om een idee te krijgen wat je over het thema digitale technologie weet en nog zou kunnen leren, kun je het volgende lijstje invullen. Bepaal zo (eventueel in overleg met je docent) welke route door de module het meest geschikt voor je is. Vul dit paspoort na afloop van het werken aan de module opnieuw in om te laten zien wat je geleerd hebt.

	Ken ik niet	Ken ik wel	Kan ik gebruiken [Veel, weinig ervaring...]
Arduino			
Raspberry Pi			
Werken met excel			
Tekstverwerker			
Browser gebruiken			
Eenvoudig coderen			
Foto's bewerken			
Robots bouwen (bv lego)			
Drones			
Computergames			
Programmeren games			
Website bouwen			
Zoeken in databases			
Computer bouwen			
Router instellen			
Kunstmatige intelligentie gebruiken			
Zoeken naar computercode			
Code met anderen delen			
Een webwinkel inrichten			
Een computerserver inrichten of beheren			
Software installeren			
Elektronica monteren (solderen, schakelingen etc)			
Programmeertalen zoals: C++, Python			
Apparaten uit elkaar halen, maken, repareren, klussen, sleutelen,...			
Moderator website			
Vloggen			
Micro:bit			

1.1 Scrum

Net als overal in de wereld van ICT is samenwerken belangrijk én ingewikkeld. Het ontwikkelen van digitale technologie vraagt inzet van mensen die allemaal aan een ander stukje van het project werken. Weten wat anderen doen, en vertellen wat jij doet is daarom nodig. Ook als je vast loopt, is het goed om hulp te krijgen. Daarnaast moet voortdurend in beeld blijven wat de gebruiker (klant) wil hebben.

Voor de samenwerking in een project en het zo snel mogelijk opsporen van hindernissen is een methodiek ontwikkeld, die we kennen onder de naam Scrum. Het is een 'agile' methode, wat betekent dat het wendbaar is en zich kan aanpassen aan de situatie. De methode is ontwikkeld in de ICT-wereld voor projecten waar software ontwikkeld wordt en wordt nu bij talloze bedrijven, en ook op school, gebruikt.

Er zijn allerlei termen die gebruikt worden voor stappen in het proces, afspraken en rollen die mensen hebben. De mensen die samen aan een klus werken, het team, heeft een *scrummaster*, die de samenwerking binnen het team organiseert. In deze module zetten we scrum ook in. Het is belangrijk dat de teamleden van zichzelf en elkaar weten wat ze kennen en kunnen.

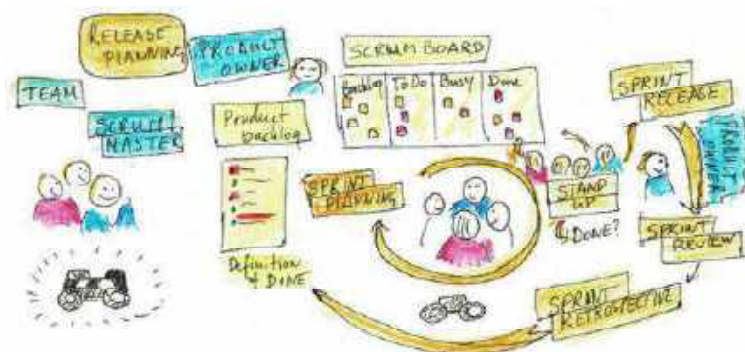
Een digitaal technologieproject begint meestal met een probleem, waar digitale technologie wellicht een oplossing kan zijn. We hebben in deze module ook een aantal eindopdrachten geformuleerd, die je kunt zien als probleembeschrijving. De gebruiker heeft wensen en eisen, geeft een opdracht en verwacht een oplossing. Dat is de 'user story'.

Een goed beeld van het probleem krijgen én nadenken over mogelijke oplossingen, is de eerste fase in elk project. Je lost het probleem van een gebruiker op, want daarvoor komt hij naar jullie toe. Voor je aan het werk gaat moet je precies weten wat de bedoeling is. Als eerste beschrijf je precies wat een gebruiker wil hebben. Er is iemand die daar verantwoordelijk voor is: *de product owner*. Hij heeft contact met de klant. Het eindproduct is er niet in één keer. In korte periodes (sprints) worden tussenproducten gemaakt. De klant kan daar iedere keer op reageren (of de product owner reageert namens de klant).

De eerste stap naar een eindproduct is om duidelijk te krijgen wat er nodig is om het product te maken. Dat wordt de *product backlog*. Daarvan omschrijft het team *taken* en bekijkt het team hoeveel werk elke taak is. Grote taken krijgen veel punten, kleine taken weinig. De taken komen in het vak '*backlog*'

op het scrumbord. Je hoeft niet persé met een papieren scrumbord te werken. Er zijn talloze online varianten van het scrumbord om taken van een team te managen.

Taken worden niet allemaal gelijk aangepakt, maar verdeeld in groepjes die bij elkaar horen, de sprints. Een sprint duurt ongeveer twee weken. Het team staat bij het planbord (scrumbord), en plakt de taken voor een sprint bij elkaar onder 'TO DO'. Dat zijn de taken die er voor deze sprint gedaan moeten worden. Het team verdeelt taken en noteert op de taakbriefjes wie er aan werkt. Taakbriefjes waar aan gewerkt wordt komen onder 'BUSY'. Met één blik op het scrumbord weet je waar je team aan werkt. Vorderingen worden zichtbaar door het verplaatsen van taakbriefjes tijdens



Figuur 1.6 De Scrum-methode voor van een project

de dagelijkse ceremonie: de 'standup'. Taken die afgerond zijn komen bij 'DONE'. Bij taakbriefjes die nog bij BUSY blijven hangen, vragen teamleden elkaar of de taak vordert, of er hulp nodig is. Misschien is de conclusie dat een taak in kleinere stukjes verdeeld moet worden.

Na een sprint zijn de taakbriefjes, als het allemaal soepel gaat, verplaatst naar 'DONE'. Er is dus een stukje van het project klaar. De product owner kijkt of dit stukje van het project geslaagd is en voldoet aan de wensen van de klant. Het team kijkt terug op de sprint (sprint review), en bepaalt of er nog dingen anders moeten of vergeten zijn. Die dingen worden als taken toegevoegd aan de product backlog, en bij een volgende sprint aangepakt. Opnieuw wordt het planbord gevuld met taken voor de volgende sprint en werkt het team opnieuw een cyclus af.

Regelmatig kijken of iets dat je gemaakt hebt af is en werkt, wordt bij de ontwikkeling van computerprogramma's veel gedaan. Werken in kleine stukjes die áf zijn is heel prettig. Daardoor blijft een project wendbaar: 'agile' of 'lean and mean'. Het is prettig wanneer je hulp krijgt als je vast loopt. Als er tijdens een sprint iets anders blijkt te moeten, kan er snel bijgestuurd worden. Alles is tijdens een sprint getest. Aan het eind worden alle onderdelen samengebracht en nog een keer getest. Dan is ook duidelijk of het einddoel bereikt is. Aan het begin is duidelijk wat er gemaakt wordt, niet de exacte stappen om er te komen. Details van taken worden in een sprint bepaald en kunnen elke sprint worden bijgesteld.

Scrum toepassen

In de module kun je scrum ook heel goed gebruiken, bijvoorbeeld om de eindopdracht samen aan te pakken. Precies beschrijven wat er gemaakt moet worden (design), wat daarvoor nodig is en hoe je dat als team aanpakt: dat is waar scrum voor bedoeld is. Je hebt verschillende kwaliteiten in je team nodig, zoals samenwerken, anderen helpen, schrijven, concentreren, overleggen, precies zijn, plannen, stevig doorwerken en dergelijke. De een weet hoe je code schrijft, een ander wat je wilt meten, een handleiding schrijven of praktische verpakking ontwerpen, en een derde kan bruikbare manier bedenken hoe gegevens aan een gebruiker getoond kunnen worden. Elke taak vraagt weer andere kwaliteiten in je team. Naast algemene kwaliteiten is er ook ervaring met het onderwerp van de module. Daarvoor zijn ruwweg drie mogelijkheden:

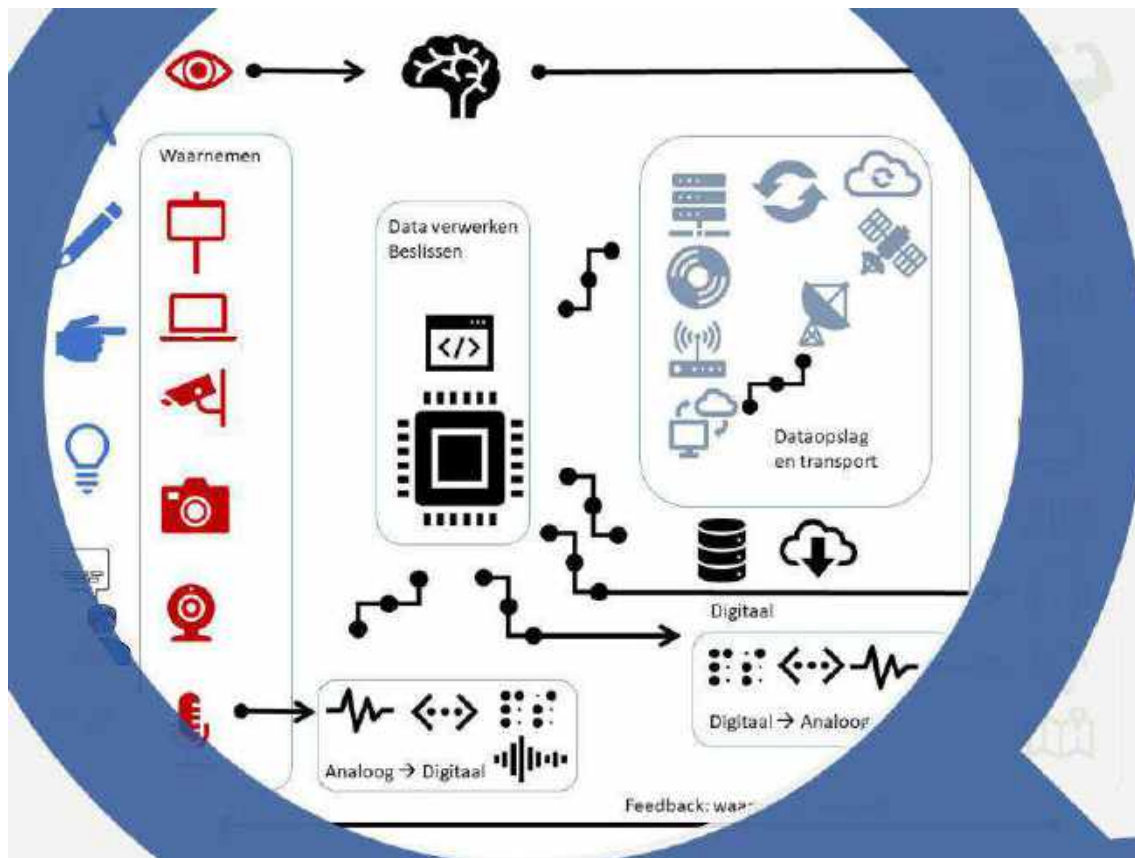
1. *Algemeen gebruiker*: Je bent iemand die digitale technologie gebruikt (bijvoorbeeld je telefoon, websites of programmatuur), maar ze zelf niet installeert of programmeert.
2. *Beheerder en ontwerper*: Je zet computers in elkaar, beheert een website of een server of denkt na over de manier waarop software zou moeten werken.
3. *Codeerder en bouwer*: Je bent bezig met het maken en aanpassen van programma's, bouwen en testen van apparaten, solderen, drivers zoeken en installeren.

Bij de indeling van een team is het nodig om zowel op persoonlijke kwaliteiten als het inhoudelijke profiel (Algemeen gebruiker, Beheerder of Codeerder) te letten. Bij het bestuderen van de hoofdstukken komt dat onderscheid ook van pas. Je zou alles van het hoofdstuk moeten kennen en kunnen, maar het startpunt kan verschillen. Deel A is een goede start voor Algemeen gebruikers; B voor de Beheerders en C voor Codeerders. Uiteindelijk kun je elkaar uitleg geven en helpen bij de theorie en praktische vaardigheden.

Opdracht 1.6: Team samenstellen.

Stel je team (viertal) samen op basis van de *teamkwaliteiten* en inhoudelijk *ABC-profiel*.

2 Introductie op de module Digitale Technologie



Leerdoelen

Kennis

- 2.1 Kennismaken met basisbegrippen die in de digitale wereld gebruikt worden. De volgende basisbegrippen kun je uitleggen en gebruiken: sensor, analoog-digitaal conversie, interface, data, processor, dataopslag, datatransport, datacommunicatie, actuator, cloud, ic, programma, programmeren, algoritme, digitaal-analoog conversie.
- 2.2 Opbouw van een digitaal apparaat herkennen en beschrijven aan de hand een schematische weergave (het analyseschema digitale technologie).

Vaardigheden

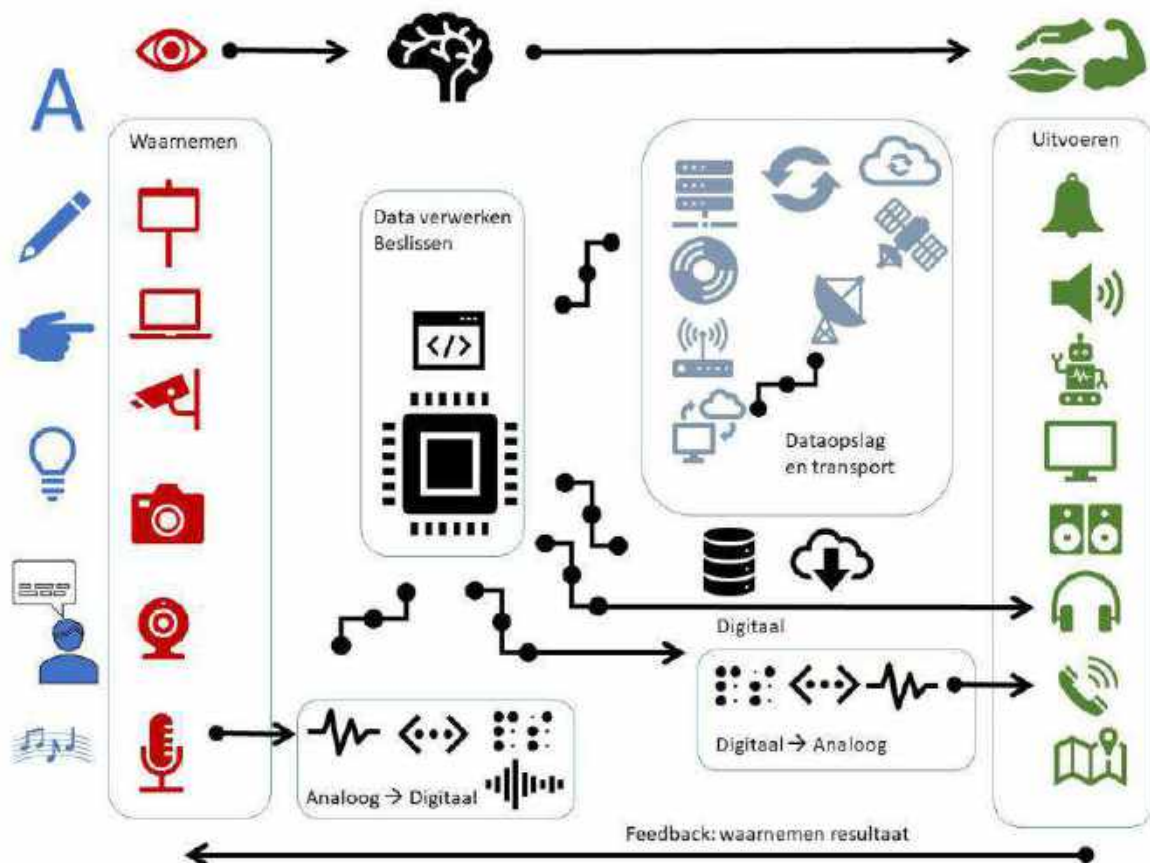
- 2.3 Je kunt de functie van een digitaal apparaat uitleggen aan de hand van de basisfuncties waarnemen, data verwerken, beslissen, dataopslag en -transport, uitvoeren en terugkoppeling.
- 2.4 Je kunt de relatie tussen een fysieke grootheid en een numerieke (digitale) grootheid uitleggen en welke voor- en nadelen het heeft om informatie digitaal te verwerken.



Deel A Uitleg en naslag

In dit gedeelte maak je kennis met de basisbegrippen van digitale technologie en een manier om de onderdelen van een digitaal systeem te kunnen benoemen en herkennen: het analyseschema.

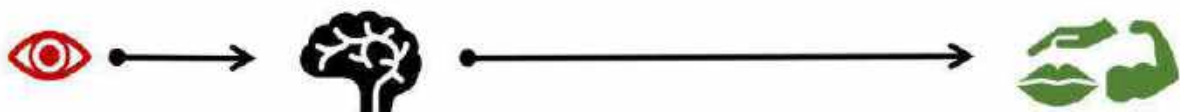
2.1 Basisbegrippen



Figuur 2.1 Schema digitale technologie

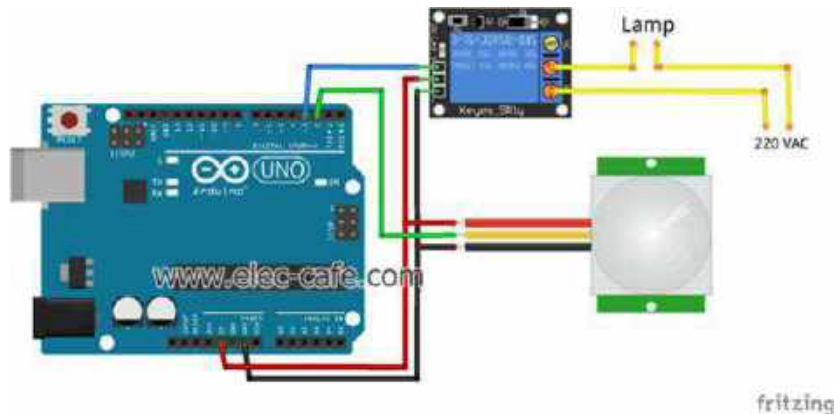
Digitale technologie maakt gebruik van ‘digitale’ apparaten. Digitaal houdt in dat het apparaat signalen gebruikt die uit maar twee verschillende waarden (bits) bestaan: nul of een. Een rijtje bits vormt samen een klein blokje informatie, bijvoorbeeld een getal of een letter.

Alle apparatuur, die bij digitale technologie wordt gebruikt, heeft overeenkomstige onderdelen en functies. Figuur 2.1 geeft hiervan een schematisch overzicht. Sommige onderdelen komen in ieder digitaal apparaat voor. Wanneer onderdelen ontbreken, betekent het vaak dat een digitaal apparaat *andere apparaten* of de mens nodig heeft om te kunnen functioneren. Want: digitale apparaten/ digitale technologie bestaan omdat mensen daarmee werken. Altijd moet de verbinding tussen mens en machine gemaakt worden via interfaces (zoals schermen, knopjes en toetsenborden).



Figuur 2.2 Zintuig - hersenen - ledematen: waarnemen, verwerken en uitvoeren in ons lichaam.

Een digitaal systeem heeft veel overeenkomsten met onszelf. Wij hebben zintuigen om de omgeving **waar te nemen**, hersenen die informatie **verwerken en beslissingen nemen** voor ons handelen, en organen of ledematen die deze handelingen **uitvoeren**, zoals handen, benen of stembanden. Verandering van de omgeving nemen we weer waar met onze zintuigen, de feedback, om ons handelen weer te kunnen bijstellen.



Figuur 2.3 Sensor (voor licht), processor (Arduino) en actuator (lichtschakelaar): waarnemen, verwerken en uitvoeren in een digitaal apparaat.

Met techniek zijn zintuigen, hersenen, organen en ledematen nageemaakt. Ze heten dan: sensoren, processoren en actuatoren. In plaats van zenuwvezels lopen er in een apparaat draadjes, die de signalen doorgeven.

Algemene beschrijving van het schema

In het schema van een digitaal systeem (fig. 2.1) zijn vier belangrijke blokken te onderscheiden. Waarnemen – verwerken en beslissen – dataopslag en datatransport – uitvoeren. We leggen eerst uit wat er mee bedoeld wordt. In de vervolghoofdstukken gaan we elk van deze blokken verder uitwerken.

1. Waarnemen (hoofdstuk 3)



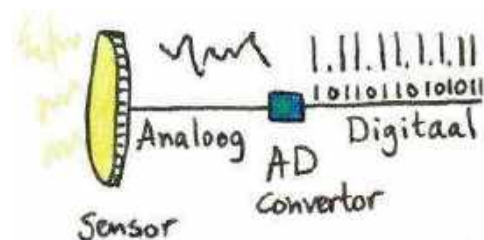
Figuur 2.4
Waarnemen



Figuur 2.5
Conversie

Als mens nemen we onze omgeving, de fysieke wereld, waar. Licht via onze ogen, aanraking met de huid, chemische stoffen in de lucht met onze neus, of bewegingen met ons evenwichtszintuig. Veel apparaten, die wij gebruiken, doen dat ook. Ze zijn uitgerust met *sensoren*. Een telefoon heeft bijvoorbeeld sensoren voor verplaatsing, licht (camera's), aanraking, warmte, geluid, magnetisme of radiogolven. Hiermee kunnen wij het apparaat bedienen, praten, foto's maken of onze positie bepalen. Door gegevens van sensoren op een slimme manier te combineren, kunnen veel dingen via een omweg gemeten worden, bijvoorbeeld het meten van verplaatsing door een camera te gebruiken.

Een *sensor* zorgt dat een fysische grootte (bijvoorbeeld licht of temperatuur) naar een *elektrisch signaal* wordt omgezet. Het signaal uit de sensor is *analoog* (kan oneindig veel waarden aannemen) en wordt vertaald in een *digitaal* signaal (een reeks blokjes met een hoge óf een lage spanning). Die truc wordt uitgevoerd door een *analoog-digitaal omzetter* (AD converter, *figuur 2.6*). Het resultaat is een *binair* signaal (0 of 1).



Figuur 2.6 Analoog naar digitaal omzetting

Mens	Functies	Machine	Functies
Zintuig (uitwendig en inwendig)	Waarnemen (licht, geluid, temperatuur, aanraking, chemische stoffen, beweging; inwendig bijvoorbeeld bloedsuiker, hormonen, kooldioxide)	Sensor	Invoer: waarnemen van de buitenwereld (licht, temperatuur, aanraking, chemische stoffen enz.)
Zenuwcellen, hersenen	Verwerken en beslissen	Processor	Berekenen en informatie uitsturen
Geheugen	Netwerk van verbindingen tussen zenuwcellen	Dataopslag	Opslag van bits in geheugenchip of op harde schijven
Spier en klier	Uitvoeren van bewegingen (dus ook ademen, geluid maken enz.) en afgeven stoffen of hormonen (voor regeling in lichaam)	Actuator	Bewegen, licht geven, geluid maken enz.
Communicatie	Praten, bewegen enz.	Datacommunicatie	Bits via kabels of draadloos versturen

Tabel 2.1 Mens en digitale machine vergeleken

2.1 Vragen

- Deze module gaat over 'digitale technologie'. Er zijn hiervan al enkele voorbeelden langs gekomen. Welke toepassingen kun je nu nog meer bedenken?
- Digitaal staat tegenover analoog. Wat is het verschil tussen beide?
- Welke voordelen heeft digitale technologie ten opzichte van analoge technologie? Welke nadelen kun je noemen?

2.2 Opdracht

Geef voor elk van de genoemde signalen een manier / apparaat om deze analoog te meten en te registreren.

2.3 Opdrachten: Analooq en digitaal

- Analoog naar digitaal conversie en andersom is een belangrijke stap in de digitale wereld. Geef op basis van afbeelding 2.1 drie voorbeelden van analoog-digitaal conversie en drie voorbeelden van digitaal-analoog conversie.
- Noem nog vier andere digitale apparaten, die alleen in combinatie met een digitaal systeem kunnen functioneren.
- Kies één van de apparaten die je bij (b) genoemd hebt. Aan in welk systeem is het apparaat verbonden? Is het apparaat bedoeld voor invoer, uitvoer, opslag of verwerking van digitale gegevens?

2.4 Opdracht: Electr(on)ische apparaten die jij gebruikt: digitaal of niet?

Als je naar school fietst op een regenachtige dag is de kans groot dat je de buienradar op je mobiel in de gaten houdt. Die app werkt alleen als satellieten en grondradars hun data doorgeven aan computers met rekenmodellen, die dit via een website en communicatie met jouw mobiel zichtbaar maken.

- a. Welke apparaten (die gebruik maken van elektriciteit) gebruik jij elke dag? Noteer er tenminste zeven.
- b. Sommige apparaten zijn digitaal, andere analoog, of zijn deze termen niet van toepassing. Schrijf achter elk van de namen van je apparaten of ze digitaal, analoog of geen van beide zijn.
- c. Bepaalde digitale apparaten werken helemaal zelfstandig (hebben geen andere apparaten nodig), sommige hebben andere apparaten nodig om te werken. Geef twee voorbeelden van zelfstandige digitale apparaten, en twee voorbeelden van verbonden (samenwerkende) apparaten.

2. Verwerken en beslissen (hoofdstuk 4)



Figuur 2.7
Verwerken en
beslissen

Digitale signalen lopen als stroompulsjes door metaaldradjes, flitsen als radiogolf of licht door de lucht en door glasvezel. Bij een digitaal signaal maakt het weinig uit hoe groot elk pulsje is: als er maar onderscheid te maken is tussen wel en geen puls (1 of 0). Dat *voorkomt verlies* van het signaal.

Het verwerken van de pulsen in een digitaal systeem gebeurt in heel kleine chips (enkele vierkante mm). Deze *microprocessors* zitten vol *elektronische schakelingen*, geïntegreerde circuits (IC), die miljoenen kleine onderling verbonden onderdelen bevatten. Dat zijn o.a. *transistoren* (schakelaars), *diodes* (die elektronen in één richting doorlaten) en *condensatoren* (voor het tijdelijk opslaan van elektronen).

Een IC zet de spanning op uitgaande metaaldradjes aan en uit op basis van binnenkomende signalen. Wat de IC precies doet, wordt bepaald door de bedrading én door de stand van de schakelaars. Door tevoren de schakelaars in een bepaalde stand te zetten met stroompulsjes, wordt een IC *geprogrammeerd*. Soms ligt dat programma vast: de processor is dan gemaakt om altijd hetzelfde te doen. Andere processoren zijn telkens opnieuw programmeerbaar. Alle schakelaars komen in de beginstand als de IC geen stroom meer krijgt (*reset*). De *software* kan de IC dan opnieuw programmeren. Die van je mobiel heeft telkens andere klusjes, dus vertelt de *app* (applicatie, het programma) wat er moet gebeuren. Wat de IC doet wordt bepaald door de regels (*algoritmen*, zie kader) in de software. Daardoor is een processor flexibel te gebruiken.

2.5 Vraag

Algoritmen (zie kader) kom je op veel meer plaatsen tegen dan alleen in software. Bedenk een paar andere voorbeelden.

Algoritme

Het woord algoritme herinnert aan de Arabische wiskundige Al-guarismi, die ook de basis legde voor Al-gabbr (algebra). Hij beschreef het oplossen van een (reken)probleem in stappen, die samen een algoritme vormen, dat je kunt volgen om een probleem op te lossen.

Dat doe je in je dagelijks leven voortdurend. Als je een boterham met kaas wilt klaarmaken, volg je een aantal stappen. Welke je precies volgt is afhankelijk van de situatie.

Werk je met een groot stuk kaas of plakjes? Gebruik je een kaasschaaf of een mes? Als je de stappen precies noteert voor elke situatie, krijg je verschillende algoritmen. Verschillende algoritmen kunnen dus dezelfde uitkomst hebben.

3. Datatransport en dataopslag (hoofdstuk 5)



Figuur 2.8
Opslag en
transport van
data



Figuur 2.9
Dataopslag

De enen en nullen in digitale signalen worden door de processor voortdurend verplaatst in zijn eigen *werkgeheugen* en als *digitale data* opgeslagen op harde schijven of geheugenchips. Hiermee kan *data* op een ander moment verder worden bewerkt of gebruikt, maar ook verplaatst en ergens anders opgeslagen worden. Opslag in de '*cloud*' komt er op neer dat (een kopie van) die data op de schijven van een of meerdere digitale opslagplaatsen ergens ter wereld terecht komt.

Als je een foto met je mobiel maakt en deelt met vrienden, kopieert en verstuurt je mobiel de data. Dat gaat draadloos via 4G of 5G naar een telefoonmast, of via *wifi* naar een netwerk van koper en glasvezel, en komt via *routers* terecht bij de *server* van de app met opslagruimte in een *datacenter*. Google, Microsoft, Facebook, Zoom en de bedrijven achter al de apps, gebruiken enorme datacenters én supersnelle verbindingen met hun servers om al die digitale data te kunnen ontvangen, opslaan, bewerken én weer door te sturen. Heel dynamisch dus. De foto die je maakt, en die vlak daarna te zien is op de mobiel van je vriendin die naast je staat, heeft al binnen een seconde de wereld rondgereisd.

Het beschermen van computersystemen, *cybersecurity*, is een belangrijk thema bij transport, opslag en gebruik van digitale data. Het is belangrijk dat niet de hele wereld jouw foto's kan bekijken als je dat niet wilt. Ook verwijderen van data blijkt lastig, omdat in een paar seconden een aantal kopieën op allerlei servers staan, waar je geen controle meer over hebt.

2.6 Opdracht

Maak een tekening met de stappen (tussenstations), die een foto aflegt als die van jouw mobiel vervoerd wordt naar de mobiel van een klasgenoot.

4. Uitvoeren (hoofdstuk 6)

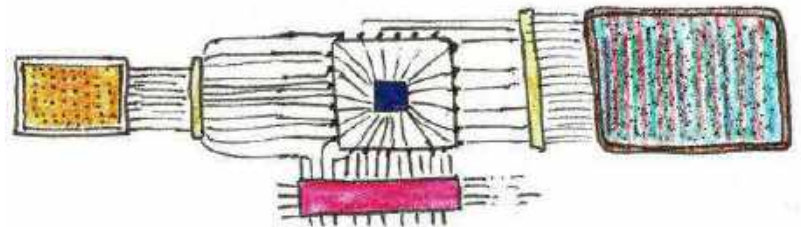


Figuur 2.10
Uitvoeren



Figuur 2.11
Conversie

Aan pulsjes, de bits, in een digitaal systeem heb je alleen maar iets, als je er iets mee kunt doen. Digitale data van een foto wil je graag op een schermje kunnen bekijken. Een rijtje bits kan gebruikt worden om één pixel van een schermje licht te laten geven. Er zijn miljoenen enen en nullen nodig om een afbeelding of tekst op een beeldscherm te laten zien. Maar je kunt ook geluid maken via een stroompje door de spoel in een luidspreker. Op die manier vormen een scherm, luidspreker of motortje de uitvoer van een digitaal systeem en worden daarom *actuatoren* genoemd. Een digitaal signaal wordt vaak direct gebruikt om een beeldpuntje aan of uit te schakelen. Om een luidspreker te laten werken, moet het *digitale signaal* eerst omgezet worden in een *analoog signaal* door een *digitaal-analoog omzetter (DA-converter)*. Dit analoge stroompje kan de luidspreker laten werken om je muziek af te spelen.



Figuur 2.12 Digitale camera: sensor - processor - opslag - actuator (beeldscherm)

2.7 Opdracht

Welke actuatoren zijn er in jouw directe omgeving aanwezig?

5. Feedback (geen eigen hoofdstuk)

In het analyseschema staat het begrip *feedback* (terugkoppeling): je neemt met *sensoren* waar hoe een fysische grootte in de omgeving verandert als een *actuator* actief is. Bij het nauwkeurig en correct uitvoeren van opdrachten, is het enorm belangrijk dat er controle is op de effecten. Als jij water in je glas giet, blijf je kijken wat je doet, om te voorkomen dat je er naast giet of het glas overstroomt. Net als jij je ogen gebruikt om waar te nemen welk effect je handelingen hebben (terugkoppelen), moet ook een digitaal systeem zijn sensoren gebruiken om te ontdekken wat het effect is van de opdrachten die de actuatoren uitvoeren. Die sensoren ontbreken vaak, omdat digitale systemen met en voor de mens werken. Daardoor doet de gebruiker zelf vaak de terugkoppeling, bijvoorbeeld het regelen van het geluidvolume als je muziek beluistert. Hoe automatischer en zelfstandiger een apparaat werkt, hoe belangrijker de rol van terugkoppeling is. Als heel nauwkeurig te voorspellen is wat een actuator doet (bijvoorbeeld: een beeldscherm), is het minder belangrijk (maar wel prettig) om allerlei sensoren te hebben. Je mobiel gebruikt een lichtsensor om te bepalen hoeveel omgevingslicht er is en welke kleur dat heeft. Het scherm is op die manier in fel daglicht en in het donker goed te gebruiken.

2.8 Opdracht: Voorbeelden van terugkoppeling

Geef nog een paar voorbeelden van terugkoppeling die je kent uit de biologie, natuurkunde of scheikunde. Beschrijf bij elk voorbeeld kort wat zou er kunnen gebeuren als die terugkoppeling ontbreekt.

2.9 Opdracht: Terugkoppeling bij temperatuurregeling

Met een systeem waarin een temperatuursensor, processor, temperatuurinstelling, en verwarmers aanwezig zijn kan de temperatuur in een ruimte op peil gehouden worden. Maak een schema waarin je duidelijk maakt hoe terugkoppeling een rol speelt bij het constant houden van de temperatuur.

6. Verbindingen (hoofdstuk 5)

Tussen de vakken in het analyseschema zijn verbindingen getekend. Hiermee wordt aangegeven dat informatie van een vak naar een ander vak wordt overgedragen. Van een onderdeel, zoals een sensor, loopt elektriciteit, via stekkertjes en draadjes of via pinnetjes op



de printplaat, naar de processor. De informatiestroom kan beide kanten op  of in één richting . Tussen computers worden netwerkkabels of glasvezel gebruikt of draadloze overdracht via radiogolven. Een deel van die verbindingen hoort bij het vakje datatransport en -opslag.



Photo by [Mika Baumeister](https://www.vecteezy.com/free-photos) on [Unsplash](https://www.vecteezy.com/free-photos) <https://www.vecteezy.com/free-photos> Pixelpatroon, Wikipedia: https://en.wikipedia.org/wiki/Bayer_filter

Voorbeeldschema: foto's met je mobiel.

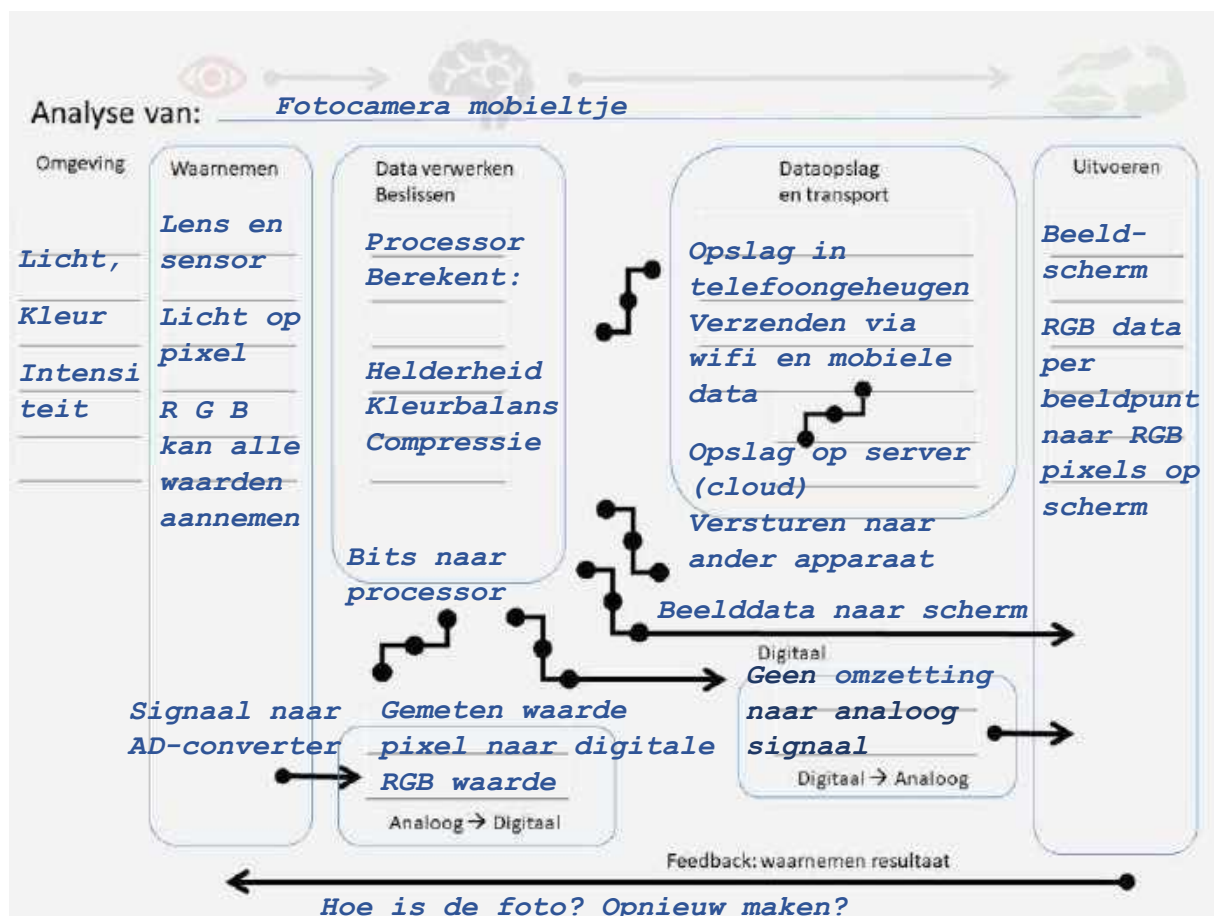
Een mobiele telefoon heeft een camera waarmee je foto's en filmpjes maakt. De camera heeft daarvoor een sensor die via een lens licht opvangt. De sensor is verdeeld in kleine blokjes (pixels), die binnenkomend licht vertalen in een elektrisch stroompje. Naast elkaar zitten een pixel voor R (rood), G (groen) en B (blauw) licht, die samen één RGB beeldpunt vormen. De waarde voor R, G en B is **analoog**, dat wil zeggen dat het alle mogelijke waarden kan aannemen.

In de camerasensor wordt het *analoge* signaal vertaald naar een reeks enen en nullen: een *digitale* waarde. Daarvoor gebruikt de sensor een **analoog-digitaal** omzetter. Het digitale signaal uit de sensor is een enorme tabel van miljoenen getallen, die voor ieder beeldpunt de RGB waarde weergeeft in (een rijtje van 8) bits. Een goede camera heeft al snel 5 tot 10 miljoen beeldpuntjes (megapixel). Voor één RGB beeld moeten minstens 3 (RGB) x 5 miljoen (pixels) x 8 bits verstuurd worden. Ruwe beelden uit de camera gaan digitaal naar de **processor** in de telefoon. Die processor **verwerkt** de informatie en **berekent** bijvoorbeeld hoe het beeld aangepast moet worden. Correctie voor tegenlicht (achtergrond veel te licht) of witbalans (gele foto in kunstlicht) doet de processor. Het fotobestand wordt kleiner gemaakt; als er honderd enen achter elkaar staan, kun je dat verkleinen tot "1, 100 keer". Dat heet **datacompressie**.

Als dat allemaal gebeurd is, gaat de afbeelding naar het **scherm** op je mobiel. Een scherm is een **actuator**: de digitale informatie van de foto gaat naar de beeldpuntjes op het scherm (ook R, G en B). Bekijk het scherm van je mobiel maar eens met een vergrootglas, dan zie je dat.

Tegelijk slaat de mobiel het op bij de foto's (op het **camerageheugen**). Automatische online opslag (de app foto's) of versturen naar een app (Instagram, Whatsapp of iets anders) betekent dat je mobiel de digitale informatie **verstuurt** via een draadloze verbinding (wifi, 4 of 5G) naar een server die het dataverkeer voor de app regelt. Je foto staat zo in enkele seconden op meerdere plaatsen in de wereld. Als je de foto via een app met een vriend(in) deelt, stuurt de server deze onmiddellijk door, en geeft de mobiel of computer een melding. Voor je het weet is het fotobestand bewerkt, opgeslagen, gedeeld en op meerdere schermen getoond.

Met deze informatie is het **analyseschema** (figuur 2.6) als voorbeeld ingevuld. Zo ontdek je hoe je andere digitale systemen kunt beschrijven.



Figuur 2.13 Ingevuld analyseschema voor camera van mobiele telefoon

2.10 Opdracht: Analyse van een digitale koortsthermometer

In de toelichting over de camera van een mobiel, zijn de bouwstenen van de digitale technologie besproken aan de hand van een schema. Je kunt van een digitaal apparaat of systeem nu bekijken welk onderdeel welke functie heeft en hoe de verschillende stappen gedaan worden. Een digitale koortsthermometer is een voorbeeld, waarvan je aan de hand van de toelichting een lege versie van het analyseschema kunt invullen. Wat zit er in en welke functie heeft het? Wat ontbreekt?

Wat ga je doen?

- Haal een digitale koortsthermometer uit elkaar (of gebruik er een die al uit elkaar gehaald is) en bekijk welke onderdelen aanwezig zijn.
- Verdeel de onderdelen over de vier basiscategorieën aan de hand van het analyseschema (leeg schema achterin de lesmodule of verkrijgbaar als los werkblad): waarnemen, verwerken en beslissen, datacommunicatie en -opslag, en uitvoer. Is er omzetting van analoog naar digitaal signaal en omgekeerd? Zo ja, waar?
- Wat voor omgevingssignaal of -signalen registreert de koortsthermometer?
- Welke sensor(en) heeft de koortsthermometer?
- Welke actuator(en) zijn er aanwezig?

Nabespreking koortsthermometer

Een digitale koortsthermometer heeft in het algemeen geen datacommunicatie met de buitenwereld. Een mens bedient het apparaatje en die gebruikt de getoonde gegevens.

De temperatuursensor geeft altijd een waarde aan. Als het meetsysteem aangezet wordt (schakelaar, ook een sensor) wordt de weerstand gelezen en door de AD converter omgezet in een digitaal signaal. De processor bepaalt of de waarde van de weerstand verandert (vergelijk metingen). Als de waarde gedurende 30 seconden niet meer verandert is dat de waarde die op het scherm (uitvoer) moet komen. De processor geeft de opdracht om een piepje (uitvoer via luidsprekertje) te geven. De waarde die op het scherm getoond wordt blijft in een geheugen achter. Indrukken van de schakelaar zorgt dat het apparaat uit gaat. Maar: dit moet vertaald worden in een opdracht aan de processor om te stoppen en het display uit te schakelen. Bij aanschakelen moet de processor alles weer aanschakelen en de laatst gemeten hoogste waarde op het scherm laten zien en na enkele seconden opnieuw beginnen met meten. Voor een koortsthermometer kan ook een minimale waarde ingesteld zijn om een meting te laten starten, bijvoorbeeld 30° Celcius.

Samenwerken met andere apparaten is niet nodig. Het apparaat is ook niet afhankelijk van andere apparaten om zijn werk te doen, het is een zelfstandig, *standalone*, apparaat.

In het vervolg van de module kom je meerdere systemen tegen, die we aan de hand van dit schema gaan beschrijven.

2.11 Opdracht: Oefen met andere apparatuur

Het schema is bedoeld om telkens andere apparaten of systemen te kunnen analyseren. In de volgende hoofdstukken staat telkens één onderdeel centraal. Aan de hand van voorbeelden leer je de bouwstenen van een digitaal systeem kennen, en kun je zelf aan de slag om de technologie in te zetten voor nieuwe toepassingen. Het schema is als werkblad beschikbaar.

Vul het schema in voor één van onderstaande, of kies zelf een apparaat.

- Bankpasje of OV chipkaart
- Bellen via mobiele telefoon
- Rekenmachine

Jaarlijks belanden miljoenen digitale apparaten op de afvalberg (e-waste). Ook op school of bij jou thuis zijn vast oude apparaten te vinden, die binnenkort worden afgevoerd. Ook zo'n apparaat kun je uit elkaar halen en daarmee de opdracht uitvoeren.

Op dezelfde manier kun je andere systemen analyseren. Kies zo'n apparaatje uit en probeer het schema er voor in te vullen. Let op: wanneer er veel vakjes leeg blijven, bedenk dan welke (andere) apparaten er nodig zijn om de functie van ieder vakje te vullen.

Als het om e-waste gaat: wat gooi je allemaal weg en hoe kun je dat hergebruiken?



Deel B Context

Digitale technologie is overal om ons heen. Er zijn diverse manieren ontwikkeld om ermee te experimenteren, zelf te bouwen en programmeren. In dit gedeelte maak je kennis met de meest gebruikte platforms, zodat je daar verderop in de module je eigen projecten mee kunt uitvoeren.

2.2 Basis van de hardware ontdekken met Arduino, Micro:bit of Raspberry pi

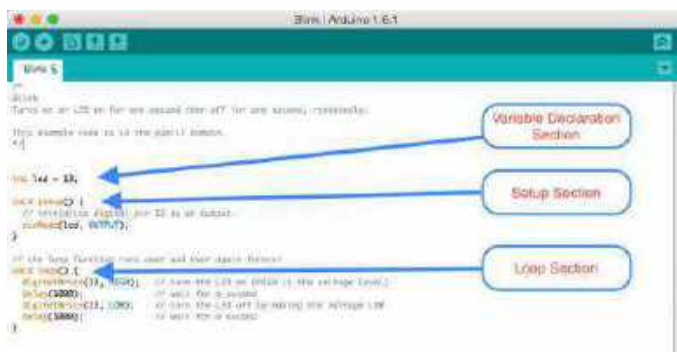
Als je een goed beeld wilt krijgen hoe digitale systemen in elkaar zitten, helpt het om ze zelf in elkaar te zetten en te programmeren. Daar zijn inmiddels veel mooie materialen voor beschikbaar, zoals 'Single Board computers' als Arduino, Micro:bit en Raspberry Pi. Misschien heb je er al wat ervaring mee, want deze kleine apparaatjes worden steeds meer gebruikt op basis- en middelbare scholen.

2.12 Opdracht

Voor we dieper in gaan op minicomputers, is het handig om het analyseschema er weer bij te pakken, en te kijken wat er allemaal in een setje van deze minicomputers aanwezig is. Gebruik voor deze opdracht een setje Arduino, Micro:bit of Raspberry Pi, of een ander systeem dat je beschikbaar hebt.

- Bekijk de componenten en deel ze in, volgens het analyseschema (leeg schema achterin de lesmodule of verkrijgbaar als los werkblad). Noteer de namen van de onderdelen in het schema en schrijf er bij wat de onderdelen doen.
- Voor welke fysische grootheden (bijvoorbeeld licht of temperatuur) zijn er sensoren beschikbaar? Zijn er meerdere sensoren voor dezelfde fysische grootheid?
- In welke onderdelen zitten analoog/digitaal en digitaal/analoog omzetters? Zijn er sensoren die zelf een digitaal signaal lijken te leveren? Wat zegt dat over de AD-omzetter?

Microcomputer en microcontroller



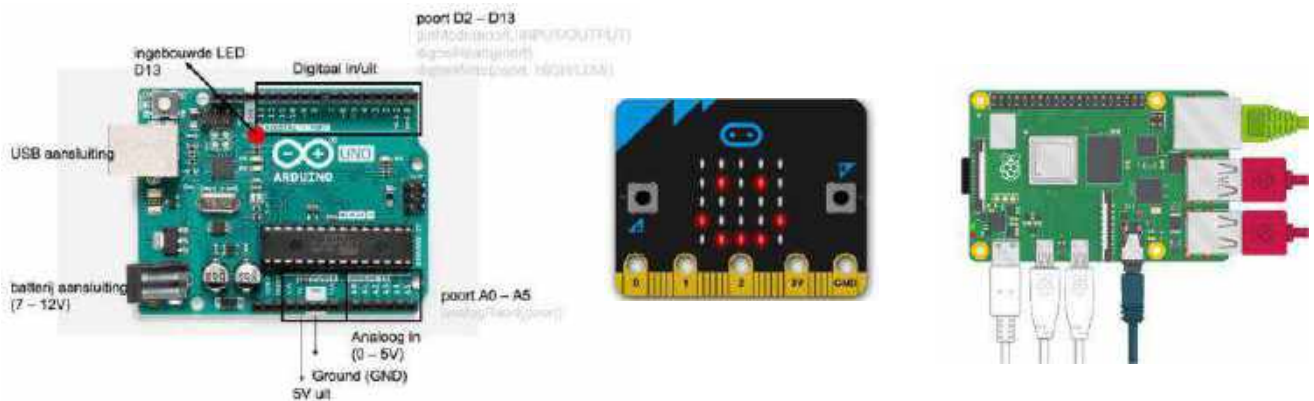
Figuur 2.14 Arduino code

De Raspberry Pi is een *microcomputer*, een kleine computer waar je een scherm en een toetsenbord op kunt aansluiten, die een verbinding heeft met het internet, en die je kunt programmeren. Eigenlijk een gewone computer, maar dan veel kleiner en met een aantal aansluitingen waarmee je iets uit kunt lezen (een druktoets) of aan kunt sturen (een LED). De Arduino en de Micro:bit zijn *microcontrollers*: je

kunt er een programma opzetten, waarmee je een sensor uitleest en bijvoorbeeld een LCD display aanstuurt. Je schrijft dat programma op een computer en 'upload' het op de microcontroller. Je kunt maar één programma tegelijk uitvoeren op deze microcontrollers, maar je kunt 'm wel steeds opnieuw programmeren.

2.13 Opdracht

Zoek uit hoeveel verschillende versies er zijn van deze microcontrollers en microcomputer.



Figuur 2.15 De bordjes van Arduino, Micro:bit (microbit.org) en Raspberry Pi (<https://www.raspberrypi.com/software/>).

2.14 Opdracht

Elk van deze drie kent zijn eigen programmeertaal (talen). Er bestaan overzichtjes op het internet. Geef het aantal programmeertalen van alle drie de types én het linkje naar een website, die deze informatie betrouwbaar en up-to-date weergeeft.

Alle drie gebruiken ze *Open Source* software. Dat betekent dat de computercode voor iedereen toegankelijk is. De programmeurs delen hun code met anderen en ze schrijven samen aan de software, vaak in hun vrije tijd. Die Open Source gedachte is overgenomen door de gebruikers. Als je iets gemaakt hebt worden de resultaten vaak gedeeld met anderen. Overal op het internet zijn voorbeelden te vinden van projecten, die je na kunt bouwen. Zo leer je van elkaars projecten.

2.15 Opdracht: Zoeken naar projecten en software

Er is enorm veel te vinden over bouwen en programmeren van computers. Bekijk een paar van deze sites en noteer in een paar steekwoorden wat je daar kunt vinden.

Alleen Arduino gaat nog een stapje verder in de Open Source gedachte, omdat het Open Source *Hardware* is. Het bouwplan van de Arduino staat online en iedereen mag er een (na)maken. Als je wilt, kun je zelfs je eigen Arduino bouwen (<https://www.arduino.cc/en/main/standalone> of <https://www.instructables.com/How-to-make-your-own-Arduino-board/>).

Zo komt het dat je in China voor heel weinig geld een Arduino kloon kunt kopen, terwijl de Micro:bit en Raspberry Pi alleen gemaakt worden door fabrikanten in de EU. Die laatste zijn vaak wat duurder, maar ook wat steviger en betrouwbaarder.

Programmeren

Een computer werkt met instructies (een programma, software) geschreven in een programmeertaal, zoals Arduino taal, C, Python, machinetaal, Visual Basic enzovoorts. Een programmeur schrijft zulke software.

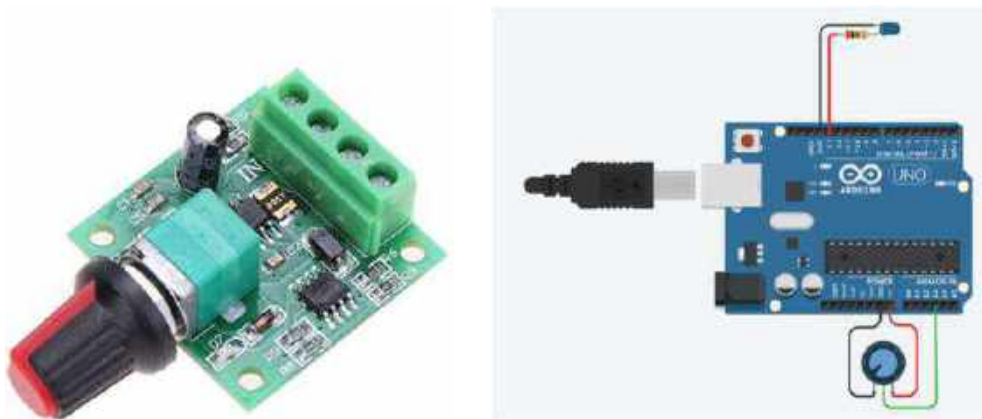
Elke taal definieert <variabelen>, heeft **commando's**, <herhaling> en een duidelijke markering van {begin} en {eind} van een opdracht. In elk goed programma (code) staat uitleg (`//comments//`). De programmeur beschrijft wat het blokje code doet.

Een werkend programma is fijn. Maar de uitleg is nog fijner, omdat iemand anders jaren later het programma kan begrijpen en aanpassen.

Waarom zijn ze handig?

Waarom zijn microcontrollers en microcomputers zo populair? Voor het antwoord op die vraag moeten we terug in de tijd. Vroeger, voor de microcontrollers, maakten elektronische apparaten gebruik van printplaten: plaatjes waarop de weerstanden, condensatoren en transistoren gesoldeerd zijn, onderling verbonden met een netwerk van geleidende koperbanen. Voor elke kleine verandering moet zo'n printje opnieuw gemaakt worden.

Kijk maar eens naar deze dimmer (fig. 2.16), met een printplaatje met elektronica, een potmeter, transistoren en condensatoren. Klein maar fijn, maar niet flexibel. Want stel dat ik mijn LED niet wil dimmen met een potmeter, maar met een drukschakelaar, die door lang en kort drukken de LED aan en uitzet én dimt? Dan heb je een heel ander printje nodig.



Figuur 2.16 Dimmen voor LEDs: een printje met variabele draaiweerstand (links) en digitaal met arduino (rechts)

Een microcontroller werkt veel flexibeler, omdat je 'm kunt programmeren. Je sluit er een LED en een potmeter op aan en schrijft een computerprogramma om de potmeter uit te lezen. Die waarde gebruik je, na wat rekenwerk, om de helderheid van de LED aan te sturen. Maar vervang de potmeter door een drukknop en verander wat aan het computerprogramma, dan kun je de helderheid ook regelen met de tijd die de schakelaar is ingedrukt en hem met een korte klik aan en weer uit zetten.

Leren

Om te leren werken met Arduino, Micro:bit en Raspberry Pi heb je eigenlijk geen lagere of middelbare school meer nodig. Op internet wemelt het van de uitleg, lessen en filmpjes. Op de websites van deze single board computers zijn ook lessen, uitleg en achtergrondinformatie beschikbaar. Als je vastloopt en het niet meer weet, zijn er internetforums waar andere gebruikers je kunnen helpen als je vragen hebt.

Beroepen

In de wereld van de digitale technologie is veel te doen. De kans is groot dat je zelf, via een onverwachte route, in deze wereld terecht komt.

FutureNL heeft hier de serie DigiDoeners over gemaakt. De labels van schoolvakken laten zien dat het heel breed is.

<https://www.lessonup.com/nl/channel/futurenl/series/7b1fc5805d7f73078018673d>

<https://www.lessonup.com/nl/channel/futurenl/series/b8914c0f6056a12da919c22a>

2.16 Opdracht: Wat is er allemaal te leren in die digitale wereld?

Hoe leer je programmeren? Bekijk: <https://www.want.nl/leren-programmeren-10-stappen/>
Hoe doen de professionals dat? Iemand die software schrijft gaat eerst kijken wat er al is: <https://www.want.nl/deze-tweehonderd-websites-apps-en-boeken-moet-je-als-ontwerper-kennen/>.

Zoek op internet nog wat verder. Hoe pakt een professional het programmeren aan? Op welke manier verschilt zijn/haar aanpak van die van jou?

Het leuke is dat je al snel begint met het maken van projecten. Je maakt een kleine schakeling, schrijft wat code, en *voila*, daar heb je een digitale thermometer, een stoplicht of een robot. Er zijn heel erg veel websites waar je leuke projecten kunt vinden, die je kunt maken. Probeer er eens een paar te vinden en maak zo'n project.

Let wel op. Techniek is soms taai. Dingen werken niet, je snapt niet waarom en je zoekt je suf. Dan blijkt dat je ergens iets niet gezien hebt, dat er iets stuk was of dat je een fout(je!) hebt gemaakt met grote gevolgen. *That's the life of an electronics nerd, sorry about that.* Hou vol, blijf nadenken en redeneren, vraag hulp (maar denk vooral eerst zelf na) aan een klasgenoot, een TOA of docent of op Internet. Uiteindelijk komt het bijna altijd goed.

Mocht je je nou afvragen waarom je deze tekst hebt gelezen, omdat alles toch online te vinden is?! We hebben hier een globaal overzicht gegeven en proberen uit te leggen, waarom dingen zijn zoals ze zijn. Wat hier staat vind je dan weer niet online, omdat dat vaak te gedetailleerd is of op een niveau, dat voor beginners te hoog gegrepen is. En natuurlijk, omdat dit het vak nlt is.

2.17 Opdracht: Welke informatie is er? (Klasopdracht)

Maak met je klas een overzicht van websites met tutorials, achtergrondinformatie en projecten. Voor beginners, maar als je wat verder bent, ook voor intermediate of advanced programmeurs. Zet dat overzicht op een plek waar iedereen bij kan.

Wat nu als jullie de zoveelste klas zijn, die dit doen? Je kunt dan laten zien welke website(s) jullie het meeste gebruikten. Welke zijn er nieuw (en misschien wel beter!)? Maak een nieuwe top 10 en vul aan met nieuwe binnenkomers.

2.18 Opdracht: Wat gaat er fout en wat leer je daar van? (Klasopdracht)

Hou voor jezelf bij waar je tegenaan loopt als je een project doet. Wat ging er mis, wat was de oplossing? Zet ook die bij elkaar. Wat ging er het meeste mis? Hoe kan de klas, die na jullie komt, daarvan leren? Wat moeten ze daarvoor precies doen? Zo leer je van elkaars fouten, vooral als die vaak gemaakt worden...

Wat als jullie de zoveelste klas zijn die dit doen? Welke fouten maakten jullie het meeste? Stonden die er al bij? Waarom ging het dan toch mis? Kan het beter beschreven worden? Helpt een uitlegfilmpje? Ook jullie kunnen zorgen dat de volgende klas het sneller leert dan jullie, doordat je ze helpt om de lijst met veel gemaakte fouten aan te vullen en te verbeteren. Op welke manier deel je de benodigde informatie?



Deel C Zelf maken

Je leert veel door dingen zelf te maken. We geven in het laatste deel van elk hoofdstuk maakopdrachten waar je (in overleg met docent en TOA) een keuze uit maakt.

2.19 Opdracht: Van start met je microcontroller

In de handleidingen bij de Arduino, Raspberry Pi of Micro:Bit zijn tientallen opdrachten te vinden waarmee je kunt starten. Als je nooit met zulke apparatuur gewerkt hebt, kun je beginnen om een LED te laten branden, een tekstje op een display te toveren of een motortje te laten draaien.

Afhankelijk van de tijd kun je een paar van deze opdrachten uitvoeren en je verdiepen in de manier waarop de programma's (code) van de microcomputer gemaakt zijn. Gebruik hiervoor de uitleg in de starthandleiding van je apparaat.

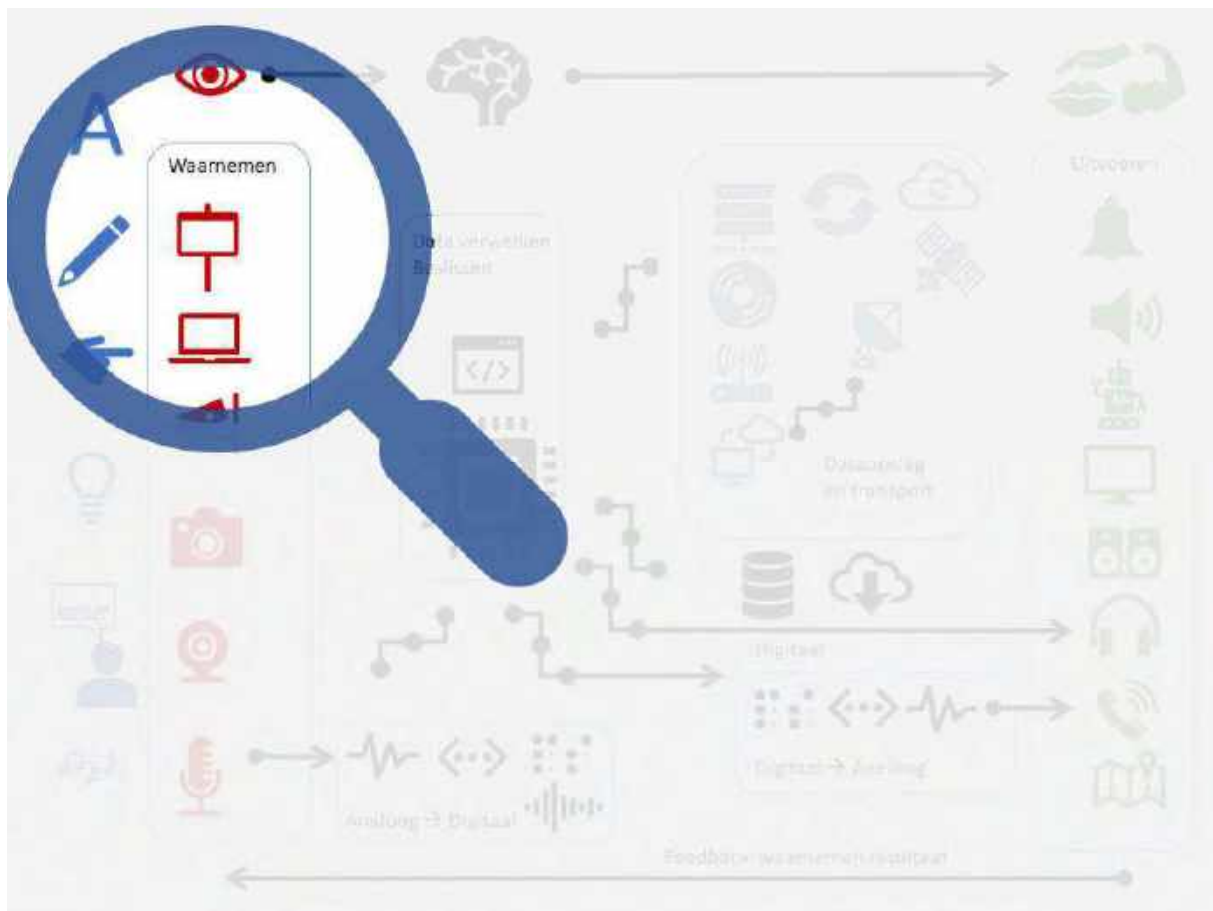
2.20 Keuzeopdrachten

Op de leerlingenwebsite bij deze module zijn praktische uitleg en maakopdrachten te vinden. Als je ergens mee aan de slag wilt, overleg met je docent of TOA wat er op school mogelijk is. De instructie heeft een vaste structuur:

Titel opdracht

1. Korte beschrijving: wat is het en waar is het voor bedoeld? Hoe ingewikkeld?
2. Wat heb je nodig? (materialenlijst)
3. Bouwinstructie
4. Software / code die gebruikt wordt en waar te vinden
5. Wat hebben we geleerd over de componenten, code en principes (verwijzen naar uitleg in de bijlagen).
6. Waar kan (de kennis uit) dit project voor worden gebruikt in de praktijk?

3. Waarnemen en sensoren



Leerdoelen

Kennis

- 3.1 Je kunt uitleggen dat digitale systemen verschillende soorten informatie kunnen waarnemen
- 3.2 Je kunt uitleggen dat voor elk soort informatie verschillende sensoren beschikbaar zijn
- 3.3 Je hebt kennism gemaakt met enkele digitale sensoren (zoals de TMP36)
- 3.4 Je kunt uitleggen dat sensoren vrijwel altijd analoge informatie omzetten in digitale informatie
- 3.5 Je kunt uitleggen dat in het dagelijks leven sensoren een belangrijke rol spelen en kunt aangeven waar je verschillende sensoren tegenkomt

Vaardigheden

- 3.6 Je hebt ervaren dat het noodzakelijk is om sensoren te kalibreren om ze goed te kunnen gebruiken en je weet hoe je dit uitvoert
- 3.7 Je hebt ervaren dat de digitale informatie van sensoren in een digitaal systeem in verschillende typen variabelen opgeslagen kan worden
- 3.8 Je kunt uitleggen hoe informatie in het werkgeheugen van een digitaal systeem is opgeslagen en dat dit in verschillende programmeertalen verrassend veel overeenkomsten vertoont.



Deel A Uitleg en naslag

Introductie, uitleg theorie over sensoren (analyseschema komt telkens terug)

3.1 De omgeving waarnemen

Een digitaal apparaat moet de omgeving kunnen waarnemen. Dat is nodig wanneer een mens het apparaat wil gebruiken. Bij een computer zijn daar het toetsenbord, de camera en muis voor bedoeld. Een druktoets is een sensor, die aanraking waarneemt en die deze informatie doorgeeft aan de processor. Toch kan bij een apparaat zowel invoer- als uitvoerinterface met de mens ontbreken. In dat geval vormt het apparaat een stukje van het geheel, dat met andere apparaten *samenwerkt*. Je kunt een draadloos toetsenbord en een draadloze muis zien als aparte, maar niet los van elkaar werkende, digitale apparaten. Immers: de muis is bedoeld om bewegingen en keuzes van de gebruiker aan de computer door te geven. In een computermuis zitten meerdere sensoren. Samen zorgen ze dat veranderingen in de fysieke omgeving (vingerbeweging, schuifbeweging) worden omgezet in een digitaal signaal dat via een draadje of via radiogolven aan een ontvanger (ook een sensor) in de computer wordt doorgegeven.



Figuur 3.1 Interface:
Invoer in het systeem

Elke *sensor* in de elektronica of digitale technologie zorgt uiteindelijk dat een verandering van een fysische grootte (bijvoorbeeld licht, vochtigheid, temperatuur, beweging of wat dan ook) een meetbaar spanningsverschil oplevert.

Temperatuursensor

Temperatuur heeft invloed op bijvoorbeeld de dichtheid van stoffen. Het vloeibare metaal kwik zet uit als het warm wordt, net als alcohol. Doe de vloeistof in een buisje, zet er streepjes naast en je kunt de krimp en uitzetting gebruiken als maat voor de temperatuur: de thermometer. Elektrische stroom, die door metaal of ander geleidend materiaal gaat, ondervindt weerstand die varieert met de temperatuur. Dat principe wordt gebruikt om een temperatuursensor voor een *elektronische thermometer* te maken. De weerstand is te bepalen door een kleine spanning (bijvoorbeeld 1 V) op de sensor te zetten. Als de temperatuur verandert, zal het spanningsverschil over de weerstand ook veranderen. Voor elektronische metingen wordt tegenwoordig vaak een NTC schakeling als sensor gebruikt.



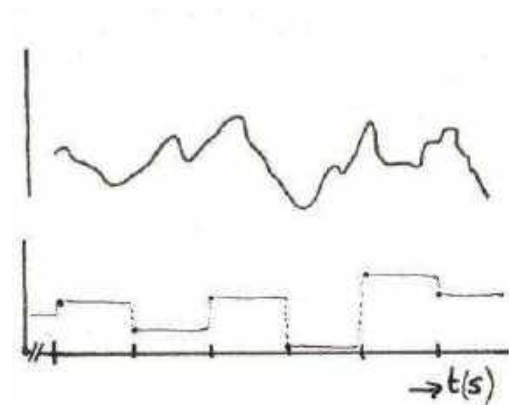
Figuur 3.2
Interface:
uitvoer van het
systeem

De eerste stap voor elektronisch meten is: een analoge fysische grootte (temperatuur) door de (temperatuur)sensor laten vertalen in een analoge spanning (V). Analooq betekent: kan alle waarden aannemen. Bij het aflezen van de sensor noteren we de waarde als een getal (*discreet*). Een gemeten spanning tussen 0,2 en 0,27 V komt bijvoorbeeld overeen met een temperatuur van 30 graden Celcius. Deze waarde moet op het schermje komen. Elke spanning tussen 0,20 en 0,27 V levert dezelfde waarde 30 op. Elke spanning tussen 0,17-0,20 V wordt 31, enzovoorts. Preciezer uitlezen betekent ook dat de temperatuur in kleinere stapjes kan worden weergegeven, bijvoorbeeld 30,7 graden C. We houden rekening met afleesfouten. De afgelezen 30 graden is een *benadering*, waarbij de werkelijke waarde tussen 29,5 en 30,5 graden kan liggen. Bij 30,7 ligt de werkelijke waarde tussen 30,65 en 30,75. Hoe groter de nauwkeurigheid van de thermometer, hoe dichter deze foutgrenzen bij elkaar liggen.

Analoog-digitaal omzetting van signaal uit sensor

Een tweede stap is nodig om van een discreet signaal een *digitaal* signaal te maken. Een spanning in de sensor (bijvoorbeeld 0,25 V) wordt door een schakeling, de *analoog-digitaal omzetter* (AD-converter), vertaald naar een reeks bits (bijvoorbeeld 11011110). Deze schakeling kan bij de sensor ingebouwd zijn, waardoor een sensor direct op een digitale ingang aangesloten kan worden. Veel sensoren geven alleen een variabele spanning (V) af. Dan moet een AD-converter op de processor dit omzetten in een reeks bits (zie figuur 2.6). De chip van de Arduino heeft hier een analoge ingang voor en doet zelf de AD-conversie. Het is dus belangrijk om te weten wat voor soort signaal de sensor verstuurt om de juiste aansluiting te maken.

Hoeveel data?



Figuur 3.4 Sampling: aantal metingen/s moet passen bij de verandering van het signaal

Bij het aflezen van de sensor is naast de gewenste nauwkeurigheid ook van belang *hoe vaak* je dat doet en waar de meetresultaten voor nodig zijn. Nauwkeurigheid wordt ook bepaald door de snelheid waarmee de fysische grootte verandert. Een langzaam veranderende grootte, zoals lichaamstemperatuur, meet je af en toe (alleen als je ziek bent) en dan ook maar een paar keer per dag. Ook de buitentemperatuur zal in één seconde niet sterk



Figuur 3.3 Analoog-Digitaal conversie van signaal uit de sensor

veranderen. Je kunt ermee volstaan om bijvoorbeeld eens per uur af te lezen. We noemen dat bemonsteren of een *sample* nemen. Hoe sneller er verandering is, hoe groter de '*samplefrequentie*' moet zijn. De vuistregel is: als een grootte verandert, moet je twee keer zo vaak een sample nemen als de snelste verandering. Als dat 100x per seconde (100 Hz) is, moet je 200x per seconde (200 Hz) *samplen*. Vaker samplen betekent ook dat er meer data ontstaan.

3.1 Vraag

- Welke fysische grootheden veranderen wel heel snel?
- Leg uit waarom de samplefrequentie minstens tweemaal zo hoog moet zijn als de hoogst voorkomende frequentie?

3.2 Vraag

Welke kenmerken heeft een signaal waarvan Arduino de AD-conversie kan doen? Denk daarbij aan de variatie en grootte van het signaal.

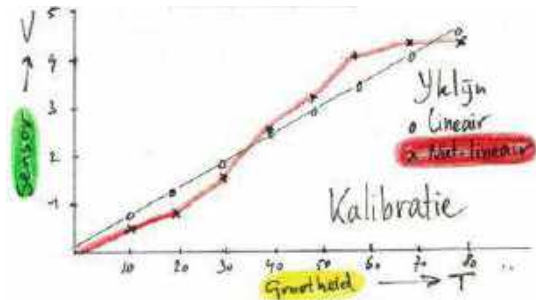
3.2 Kalibreren en ijken

Als je met een metalen draadje of een schakeling van transistors en diodes een temperatuursensor gemaakt hebt, wil je weten welke temperatuur er bij welke spanning hoort. Het bepalen van het verband tussen de temperatuur en de spanning (en omzetting daarvan naar een digitale waarde), noemen we het *kalibreren* van een temperatuursensor. Dat gebeurt bijvoorbeeld met smeltend ijs (dat is 0 graden C) en kokend water (geeft de waarde 100 graden C) of door te vergelijken met een geijkte standaardthermometer. Als je dit bij een serie temperaturen in een grafiek zet, ontstaat een

ijklijn, waarin de relatie tussen de gemeten grootte en de waarde van het signaal staat. Voor andere grootheden als licht of vochtigheid moet een vergelijkbare methode gebruikt worden.

3.3 Vraag

Bedenk hoe je dat *kalibreren* kunt aanpakken voor een lichtsensor.



Figuur 3.5 Signaal (Volt) van sensoren: een met alle waarden in een rechte lijn (lineair) en een niet-lineaire sensor, gemeten bij een reeks temperaturen.

3.4 Opdracht

Met het programma Phyphox (gratis te downloaden en installeren op je device) kun je nagaan welke sensoren er in je mobiel of ander device aanwezig zijn én je kunt ze gebruiken om te meten. Maak (volgens onderstaand voorbeeld) een lijstje van de sensoren die je in je mobiel aantreft. Noteer ook welke fysische grootte je er mee kunt meten. Welke functie denk je dat elk van deze sensoren hebben? Bedenk ook één soort meting die je op deze manier met je device kunt doen.

Voorbeelden van digitale sensoren

Fysieke grootte	Sensor	Toepassing

Beschrijving van diverse sensoren in digitale platforms

Op de leerlingensite van de module

(https://maken.wikiwijs.nl/171772/Schakelmodule_Digitale_Technologie_online_materiaal) is een overzicht opgenomen van sensoren en de manier waarop ze gebruikt kunnen worden. Raadpleeg die informatie bij de maakopdrachten in deel C.



Deel B Context

In dit gedeelte maak je kennis met een sporthorloge, een apparaat dat vol zit met allerlei sensoren. Die verzamelen gegevens en sturen ze door om ze te bewerken, zodat jij er informatie uit kunt halen. Maar hoe gaat dat eigenlijk?

3.3 Betekenis van sensoren in een context: het sporthorloge

Sensoren voor hartslag, snelheid, positie

Je gebruikt steeds meer apparaten die met internet verbonden zijn: muziek luisteren via Spotify, de verwarming aanzetten met een app, video's kijken op je telefoon via Youtube, de lamp aan en uit zetten via wifi of Bluetooth, of betalen via je telefoon. We noemen dit ook wel het Internet of Things (IoT). In huis gebeurt dat vaak al via wifi. Dit is een van de redenen dat 5G wordt aangelegd, zodat alle apparaten heel snel met elkaar kunnen communiceren via een mobiele internet verbinding.

In deze opdracht ga je voor een situatie uitzoeken hoe het werkt: een sporthorloge en een app die de metingen kan vastleggen. Je denkt na over vragen als: via welke stappen wordt de informatie verzonden? Waar is beveiliging van gegevens van belang? Waar wordt de informatie opgeslagen? Welke gegevens zijn nodig om de taak uit te voeren? Etc.

Strava app

We gaan kijken naar een app waarmee je sportprestaties kunt vastleggen. Strava is wereldwijd een van de grootste sportapps, maar er zijn diverse andere die je ook kunt gebruiken. Jullie gaan aan de slag met een sporthorloge. Misschien heb jij zelf of iemand anders in je omgeving een sporthorloge. Een sporthorloge geeft een compleet beeld van digitale technologie, omdat het meet, communiceert, data verwerkt en inzichtelijk maakt. Verder speelt beveiliging van je gegevens een rol.

Sporthorloges zijn in alle soorten en maten verkrijgbaar, met prijzen variërend van 35 tot 900 euro.

Als je geen sporthorloge hebt dan is het installeren van Strava op je telefoon ook genoeg. Naast Strava zijn er ook andere apps. Als je al een andere sportapp gebruikt dan kun je die ook gebruiken voor de opdrachten. Vermeld dat dan wel.

Je onderzoekt een sporthorloge in combinatie met het gebruik van Strava. Hoewel er allerlei verschillende platforms zijn waarop je de gegevens van een sporthorloge kunt verzamelen, is in deze opdracht gekozen voor Strava, omdat dit het meest gebruikte platform is. Maar overal waar je leest over Strava, kun je ook een de naam van een andere, soortgelijke app invullen. Voorbeelden zijn: Nike Run Club, RunKeeper, Endomondo etc. (zie *figuur 3.7 en 3.8*). Bespreek met je docent wat je/jullie gaan gebruiken in de klas. Je hoeft niet perse een sporthorloge te hebben om deze apps te kunnen gebruiken. bij heel veel activiteiten heb je al genoeg aan een telefoon.



Figuur 3.6 Een sporthorloge

Bron: Coolblue
[https://www.androidplanet.nl/nieuws/smartwatch-straling/\(afbeelding,wordt 3.8\)](https://www.androidplanet.nl/nieuws/smartwatch-straling/(afbeelding,wordt%203.8))



Figuur 3.7 Screenshot van Endomondo van een fietsroute van Erik Dekker (tijdens de lock-down, in plaats van Italië) en overzichten van de data die de app verzamelt en presenteert.

3.5 Opdracht: Sport-apps.

Kies twee sport-apps uit (zie het rijtje in bovenstaande tekst). Zoek uit wat de overeenkomsten en verschillen zijn tussen deze apps.

Heb je zelf een sporthorloge? Gebruik dat als voorbeeld. Je kunt anders uitzoeken wat geldt voor bijvoorbeeld de Garmin Vivioactive 3 (of een opvolger hiervan).

3.6 Opdracht: Verschil telefoon en sporthorloge

Zoek uit en beschrijf wat de verschillen zijn tussen een telefoon en sporthorloge (Wat kunnen ze meten? Hoe nauwkeurig zijn ze? Wat is het gebruiksgemak?). Geef bij je opdracht aan welke telefoon en welk sporthorloge je met elkaar vergelijkt.

3.7 Opdracht: Wat kan een sporthorloge meten?

Welke andere onderdelen moet een sporthorloge bevatten om als sporthorloge te kunnen functioneren? Denk aan accu, bluetooth, geheugen, gps etc.. Welke varianten zijn er voor sporthorloges en waarin verschillen ze wat techniek betreft. Maak hiervan een overzichtelijke tabel. Vermeld ook over welk sporthorloge het gaat.

3.8 Vragen

Hoeveel gebruikers heeft Strava wereldwijd? Hoe vaak gebruiken zij Strava per week?

Als er *per activiteit* van een gebruiker 1 mb aan data verstuurd moet worden, hoeveel data moet er dan gemiddeld per seconde verwerkt worden bij de Strava-servers?

3.9 Opdracht: Sensoren in het sporthorloge

Kies een sporthorloge en beschrijf: Welke sensoren zitten er in en op? Heeft het horloge actuatoren, zo ja welke? Hoeveel geheugen heeft een sporthorloge? Wat is de capaciteit van de accu? Wat betekent de capaciteit van geheugen en accu voor het registreren van je activiteiten? Hoeveel activiteiten kun je opslaan en gedurende hoeveel tijd?

3.10 Opdracht: Strava gebruiken

Download Strava (*of een andere app naar keuze) op je telefoon en maak een account. Maak een opname van een sportactiviteit (bijvoorbeeld: hardlopen, naar school fietsen oid.) Je kunt het gemaakte account ook via een desktopcomputer of laptop bekijken. Kijk welke gegevens Strava* allemaal uit de je horloge of telefoon kan halen. Laat met een aantal screenshots zien wat je zo met Strava* kunt doen.

Activiteiten die je met het sporthorloge of je telefoon registreert, levert gegevens op. Dat zijn waarden die uit een reeks sensoren komen. Die gegevens uit je sporthorloge / de sportapp op je telefoon worden via je telefoon naar Strava gestuurd en daar worden de gegevens in jouw account geplaatst. De positiegegevens uit de gps van je device worden op een kaart getekend en de gegevens van andere sporters worden daarmee vergeleken. Vervolgens wordt die informatie weer naar je telefoon of computer gestuurd om het daar te kunnen bekijken en naar de mensen die je volgt gaat een berichtje van wat je hebt gedaan. Vervolgens reageert iemand en die opmerking gaat via internet naar Strava en weer naar jouw telefoon.

Als je een sportactiviteit doet met je telefoon of sporthorloge, kun je de gegevens van die activiteit via een app in kaarten en grafieken te zien krijgen. De volgende gegevens zijn o.a. te zien: de gemiddelde snelheid, hartslag, aantal stappen per minuut, hoogte, afstand en tijd van de activiteit.



Figuur 3.8 Screenshots van de gegevens van een sportactiviteit in Strava: een rondje hardlopen.

3.11 Opdracht: Sensorgegevens

Hoe bepaalt je horloge/ telefoon deze gegevens? Welke sensoren zijn er nodig? Hoe wordt de informatie van de sensoren bewerkt om de gegevens te berekenen/creëren.

3.12 Opdracht: Verwerken van gegevens

Wat is er voor nodig om de plaatjes van afbeelding 3.8 te kunnen maken (route, tempo, hartslag, cadans)? Beschrijf in je antwoord per soort gegevens wat er gemeten wordt en hoe het bewerkt wordt om deze gegevens en grafieken te krijgen.

De verwerking van dergelijke meetgegevens is vaak een *black box*. Je weet ongeveer wat er ingaat en wat er uitkomt, maar wat er in de *black box* gebeurt is onduidelijk. Het onderstaande analyseschema is bedoeld om inzicht te krijgen wat er met de informatie gebeurt en welke processen een rol spelen.

We gaan kijken naar wat de invoer is, wat er op de plaats van de streepjes gebeurt en welke processen spelen in de *black box*.

3.13 Opdracht: Invullen van het analyseschema voor een sporthorloge

Noteer in het analyseschema (leeg schema achterin de lesmodule of verkrijgbaar als los werkblad) zoveel mogelijk processen tussen het sporthorloge, je telefoon en de Strava-server. Hieronder een tabel met voorbeelden van processen die je in het analyseschema opneemt.

Invoer/waarnemen	-	Verwerking	-	Uitvoeren
Een opname van een sportactiviteit starten.	Signaal naar Strava om vanaf nu mij positie met gps bij te houden.	Er is een bestand gemaakt. Met mijn naam, datum en tijd waarin positie en tijd wordt bijgehouden.		Positie wordt geplot op een kaart en de snelheid en gemiddelde snelheid wordt berekend.

Noteer deze processen in de blauw omlijnde vakken in het analyseschema. Gebruik werkwoorden.



Deel C Zelf doen

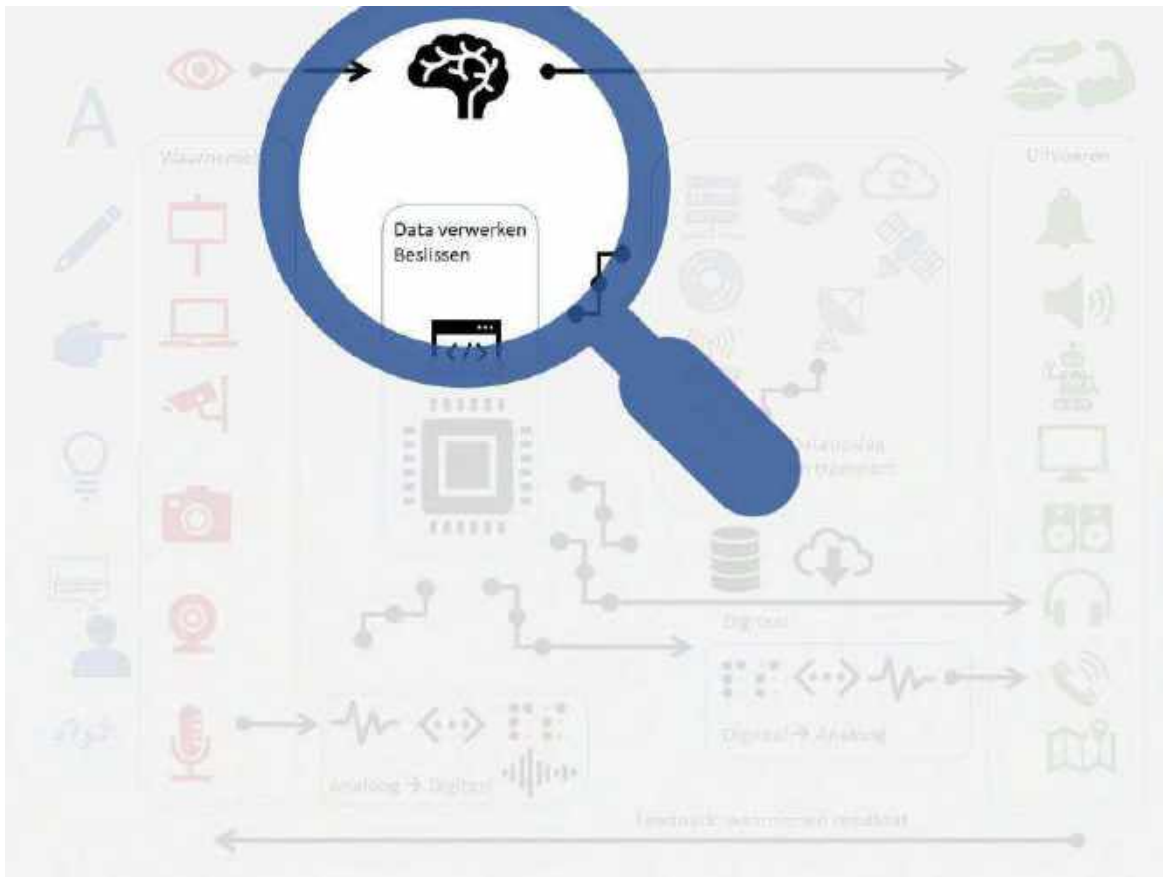
De maakopdrachten bij dit hoofdstuk zijn gericht op sensoren. Zelf sensoren maken, kalibreren of programmeren, of je verder verdiepen in de verschillende sensoren van het sporthorloge. Meer maakopdrachten zijn via je docent beschikbaar.

3.4 Onderzoeken en maken

Met sensoren kun je allerlei onderzoek doen, zelf bouwen en gebruiken. Onderstaand een aantal suggesties. Kies er één uit:

1. Een windmeter maken is interessant en goed te doen, bijvoorbeeld zoals op <https://www.microsoft.com/en-us/education/education-workshop/anemometer.aspx>
2. Onderzoek de eigenschappen van één of meerdere sensoren uit het sporthorloge (wat kan de sensor meten, welke gevoeligheid en met welke nauwkeurigheid?)
3. Haal een digitale koortsthermometer uit elkaar en onderzoek de eigenschappen van de temperatuursensor. Andere kant: kun je de koortsthermometer laten werken met een andere, zelf gemaakte, sensor?
4. Meten van vochtigheid: geleidbaarheid. Gebruik hiervoor de standaard beschikbare vochtigheidssensoren bij Arduino. Kalibreer de sensor voor toepassing in een relevante situatie. Bijvoorbeeld: een tuinder laten weten op welk moment zijn planten water nodig hebben of een waarschuwing dat een ruimte geventileerd moet worden omdat de luchtvochtigheid te hoog is.
5. Windsnelheid: gebruik een rotatiesensor
6. Meten van waterstroming (rotatie, verwarming-temperatuurverschil).
7. Indirecte meting (proxy) van grootheden die niet rechtstreeks meetbaar zijn, bijvoorbeeld met camera's, weerkaatsing van geluid, versnellingssensoren.
8. Maak zelf een vochtigheidssensor met koperdraad, gips en een houtblokje. Meet de weerstand in het koperdraad. Breng water aan op het gips en maak een ijkreeks. Wat betekent de waarde die de sensor oplevert? Hoe kan de gemeten waarde gebruikt worden in de praktijk?
9. De nlt-module *Sportprestaties* beschrijft het maken van een draaihoek-sensor. Hiervoor wordt een potmeter (draaiweerstand) gebruikt, die aangesloten wordt op de computer via Coach. Op welke manier is deze sensor te gebruiken in combinatie met Arduino of een ander microcontrollerplatform?
10. Een eenvoudige versie van de papieren vochtsensor van de Q-strip kan zelf gemaakt en getest worden. Het eerste gedeelte van de opdracht Q-strip (hoofdstuk 7) beschrijft deze werkwijze. Onderzoek de relatie tussen hoeveelheid vocht in de sensor en de waarde die met de Arduino gemeten wordt. Maak daarvan een ijkreeks.
11. Apple heeft een beker ontworpen die bijhoudt hoeveel je drinkt. Handig voor mensen die zó in beslag genomen worden door de computer dat ze alles om zich heen vergeten. Welke sensoren zouden er in zo'n beker aanwezig zijn en hoe werken ze? Pas je kennis toe in het ontwerp van een klein blokje dat je op een willekeurig drinkflesje kunt plakken en registreert wanneer en hoeveel er uit gedronken wordt.

4 Verwerken en beslissen



Leerdoelen

Kennis

- 4.1 Je kunt in grote lijnen beschrijven hoe een computerchip gemaakt wordt en welke materialen daarvoor gebruikt worden
- 4.2 Je kunt uitleggen wat 'programmeren' van een computerchip inhoudt en welke rol transistoren daarin hebben.
- 4.3 Je kunt uitleggen dat computers een computertaal hebben die uit enen en nullen bestaat en dat programmeertaal hierin omgezet moet worden door een compiler

Vaardigheden

- 4.4 Je kunt een programma en bijbehorende drivers voor de Arduino (of vergelijkbare microcontroller) downloaden, bewerken, uploaden en testen.
- 4.5. Je kunt de belangrijkste commando's voor de Arduino die je nodig hebt voor deze module gebruiken en uitleggen waar ze voor bedoeld zijn.
- 4.6 Je kunt uitleggen dat programmeertalen (zoals Matlab, C++, Python) ontworpen zijn voor specifieke toepassingen en dat ze ieder sterke en zwakke punten hebben.
- 4.7 Je bent in staat bronnen te vinden waarmee je verschillende programmeertalen kunt leren.



Deel A Uitleg en naslag

Digitale technologie bestaat niet zonder microchips. Na de uitvinding van de transistor is de electronica in een snel tempo kleiner en krachtiger geworden. We geven je in dit gedeelte een idee hoe die chips gemaakt en geprogrammeerd worden. Als je meer wilt weten kun je terecht in bijlage 3 in de moduledatabase.

Digitale apparaten verwerken informatie en beslissen wat er met die informatie moet gebeuren. Dat gebeurt allemaal in chips, waarvan een processor er ook een is. Dat is de *hardware*. Informatie kan opgeslagen worden, doorgestuurd of naar actuatoren (scherm, printer, luidspreker etc.). De chip kan een vaste bedrading hebben, waardoor telkens dezelfde bewerking wordt gedaan. Een programmeerbare chip kan telkens andere taken of bewerkingen uitvoeren. De instructies hiervoor zijn gecodeerd in de *software*. We kijken hoe die chips gemaakt worden.

4.1 Fabricage van electronica

Computerchips worden met tientallen tot honderden tegelijk gemaakt op ronde schijven zuivere silicium van 1 mm dik (*wafers*). Dat gebeurt in superschone ruimten (*cleanrooms*). Op de siliciumwafer worden tientallen superdunne laagjes aangebracht: metaal, siliciumoxide en mengsels van silicium en andere elementen. Dit is de Metaal, Oxide, Silicium (MOS) techniek, waarmee het overgrote deel van de computerchips en sensoren gemaakt worden. De siliciumwafer is aan het begin 1 mm dik. Ieder laagje is zo'n 30 nm dik. De wafer is na al die laagjes maar 2000 nm dikker, dus 1,002 mm. Een computerchip is meestal niet groter dan 1 x 1 cm, dus er kunnen tientallen tot honderden chips op een enkele wafer gemaakt worden. De wafer wordt daarna in kleine stukjes ('*chips*') gezaagd. Ze worden dan bevestigd in houders met metalen pinnetjes om ze op een printplaat te kunnen aansluiten.

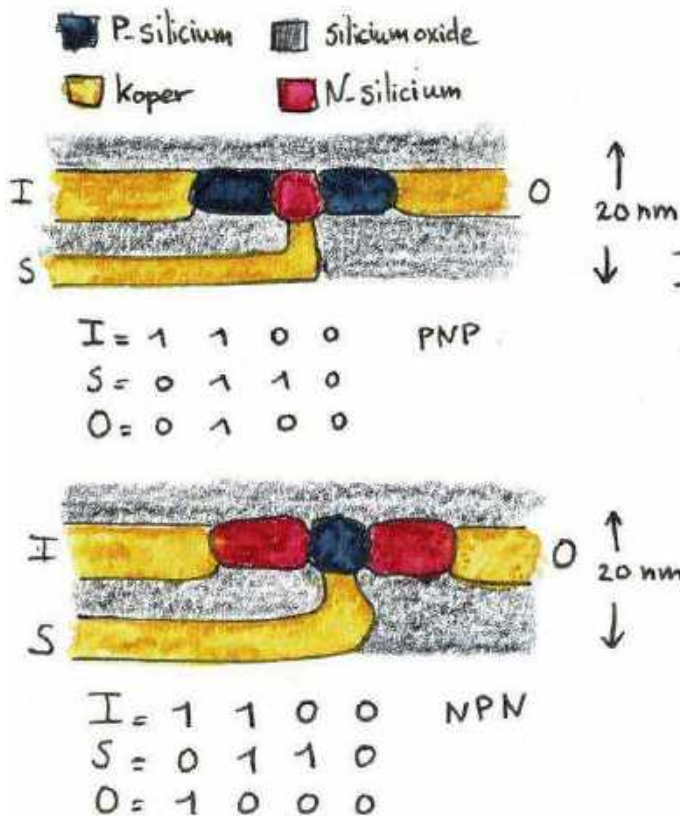


Figuur 4.1 Een productieruimte voor computerchips (*clean room*)

Onderdeeltjes op nanoschaal

De patronen in laagjes van de chip vormen transistoren, verbindingdraadjes en diodes. De lijntjes en blokjes van metaal, oxide en silicium hebben een breedte en dikte van enkele tot tientallen nanometers (nm, een miljoenste millimeter). Stel dat je met verf heel precies een lijn wilt schilderen, dan doe je afplakband (maskeertape) langs de randen. Op een wafer is het allemaal zo klein, dat daarvoor een speciale techniek gebruikt wordt: *lithografie*. De wafer krijgt een dun laagje UV-gevoelige lak. Wanneer deze lak droog is, kan die weer zacht gemaakt worden met UV-licht. Om het UV-licht precies op de goede plek te krijgen gebruikt men een '*masker*'. Een masker is een doorschijnende zwart-wit afbeelding met de verbindingen en onderdeeltjes voor de chip. De afbeelding van het masker wordt met UV op een stukje van 1 x 1 cm op de lak geprojecteerd. Dat gebeurt voor alle chips op een wafer achter elkaar in een chipmachine. Elk laagje op de chip heeft een eigen masker, dus de machine moet een hele serie maskers *na elkaar* op dezelfde wafer projecteren.

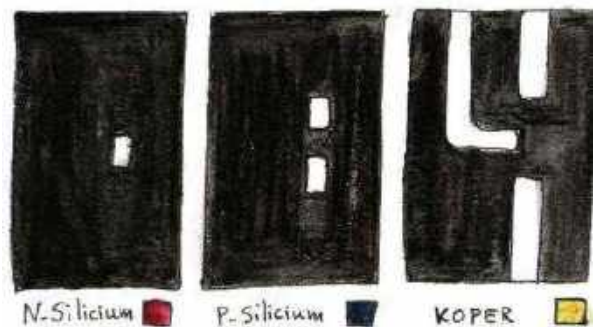
Transistor op chip



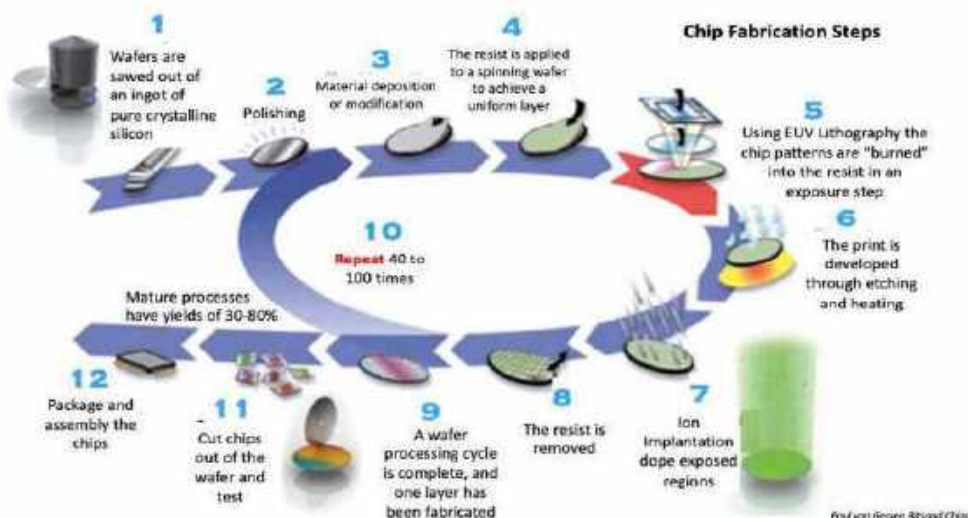
Figuur 4.2 Laagjes metaal, silicium (met n of p toevoegingen) en de verschillende maskers die voor elke laag nodig zijn om een transistor te maken. De structuren zijn enkele nanometers breed.

Realiseer je goed: er zitten tientallen chips op een wafer en elke wafer krijgt tientallen laagjes die tot op de nanometer precies op elkaar moeten zitten. Een wafer moet dan heen en weer geschoven worden tijdens het belichten, naar andere machines voor spoelen, opdampen van metaal en lakken en dan weer terug voor de volgende belichting en je wilt graag heel veel chips achter elkaar kunnen maken.

De machines die dat snel en goed kunnen zijn extreem ingewikkeld en kostbaar. Het Nederlandse bedrijf ASML uit Veldhoven loopt voorop als het gaat



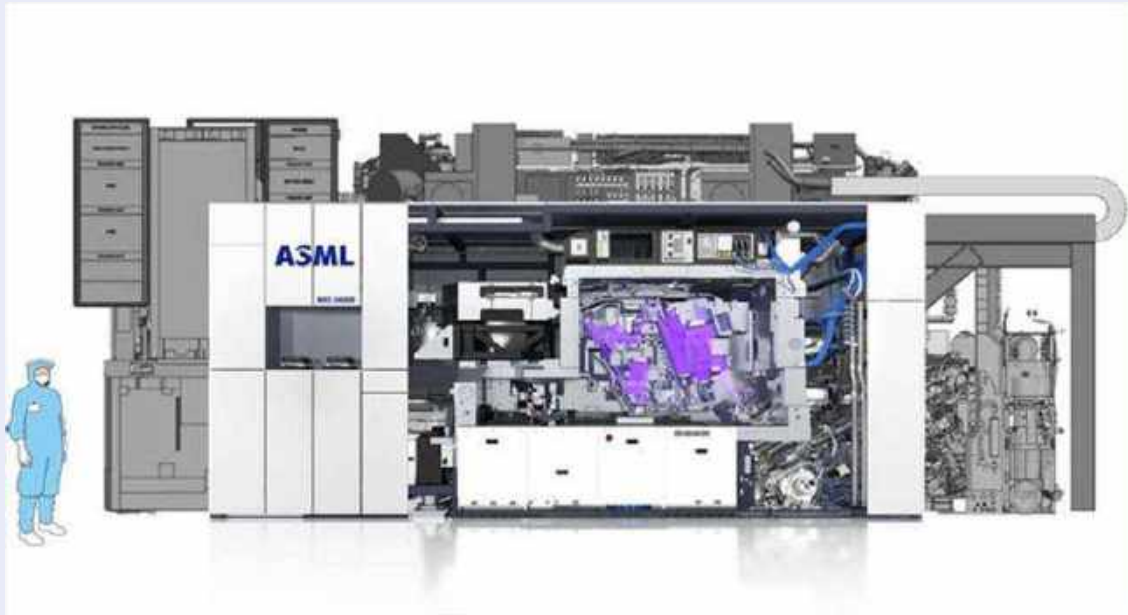
Figuur 4.3 Schema van klein stukje belichtingsmaskers in chipmachine voor onderdelen van één transistor



Figuur 4.4 Productiestappen voor een wafer met microchips

om het maken van deze chipmachines. Ongeveer driekwart van alle chips wordt gemaakt met de machines van ASML (zie kader). Ze staan vooral in fabrieken in Taiwan, Zuid-Korea en de Verenigde Staten. Daar wordt het overgrote deel van de microchips gemaakt. Inbouwen van deze microchips in printplaten gebeurt op veel meer plekken in de wereld.

Hoe werkt dat?



Voor het belichten van chips zijn uiterst precieze Lithografiemachines nodig. Dat zijn zeer dure en grote apparaten, die door 80% van alle chipfabrikanten gekocht worden bij het Nederlandse bedrijf ASML. Een masker met een afmeting van zo'n 30 x 30 cm wordt verkleind afgebeeld op de lak van de wafer. Voor het belichten wordt UV licht gebruikt. Hoe korter de golflengte van het licht, hoe kleiner de afbeeldingen gemaakt kunnen worden. De nieuwste machines die ASML maakt gebruiken EUV (Extended Ultraviolet, met een golflengte van 13,5 nm). De 'lamp' bestaat uit een krachtige laser die minuscule druppeltjes tin verhit die zo EUV gaan uitstralen. Daarmee kunnen lijntjes geprojecteerd worden die enkele nm uit elkaar liggen. Zo kunnen superkleine onderdelen op de chip worden gebouwd. Hoe kleiner de onderdelen, hoe sneller de chips kunnen schakelen. Elektronen hoeven minder ver te reizen, zodat energieverlies en warmteproductie minder zijn. Er kunnen méér bouwsteentjes op de chip gezet worden, zodat die meer bewerkingen kan uitvoeren en ook meer informatie kan opslaan. Volgens de wet van Moore verdubbelt elke 10 jaar de hoeveelheid transistoren op een chip. Met de nieuwste productietechnieken lukt het om die lijn vast te houden. Maar er is een grens aan: de afmeting van een atoom kun je niet kleiner maken. Kijk hier hoe een chipmachine van ASML werkt (link, url).

Buurten bij ASML – Wat is een chip? Hoe werkt een chipmachine?

(<https://youtu.be/5YJHwgoE3pE>)

How it's made Microchips ASML (<https://youtu.be/C4gYf-eaZTE>)

EUV: Lasers, plasma, and the sci-fi tech that will make chips faster | Upscaled

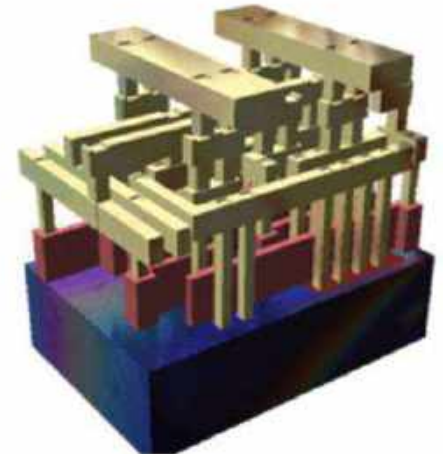
(<https://youtu.be/oliqVrKDtLc>)

De chipmachine moet afbeeldingen kunnen maken waarbij lijntjes enkele nanometers uit elkaar liggen. Op die manier lukt het om de schakelingen op de chip op een fractie van een vierkante millimeter bij elkaar te persen. Met de nieuwste E-UV chipmachines kunnen draadjes en transistoren van enkele nanometers dikte gemaakt worden, waardoor er miljoenen bouwsteentjes op een vierkante millimeter passen. Klein is vooral belangrijk voor de snelheid: de elektronen in de chip leggen dan korte afstandjes af. Méér onderdelen betekent dat de chip meer bewerkingen tegelijk kan doen en meer geheugen heeft.

In de bijlagen in de moduledatabase is veel meer te lezen over technieken die bij de productie van chips gebruikt worden en de techniek achter transistoren en logische schakelingen.

Bouwsteentjes op de chip

De ontwikkeling van de miniatuurversie van de *transistor* (elektronen-schakelaar, uitgevonden rond 1950) zoals die in je telefoon of computer zit laat zien dat het steeds sneller en kleiner kan. In een moderne microprocessors zitten miljoenen transistoren. MOS-techniek zorgt dat die op grote schaal en goedkoop te maken zijn. Met deze techniek worden ook *diodes* gemaakt, die zorgen dat elektronen maar in één richting kunnen lopen. Een bekende diode is de LED (Light Emitting Diode). Dat de elektronen hier maar in één richting doorheen kunnen kun je uitproberen. De LED geeft alleen licht als hij in een stroomkring in de juiste richting in aangesloten: met het lange pootje op de pluspool. Ook de *condensator*, die elektronen kan opslaan en afgeven, is een belangrijke bouwsteen op de chip.



Figuur 4.5 3D-structuur van de laagjes op een chip. De geleidende pilaartjes van metaal met alle componenten van de chip worden laag voor laag opgebouwd, tot een dikte van enkele micrometers. De blauwe onderkant is de silicium wafer.

4.2 Programmeren van een chip

Chip-ontwerpers streven er naar om véél componenten bij elkaar te pakken op één chip. Via contactpuntjes aan de buitenkant van de chip staat die in verbinding met de pinnetjes van de behuizing. Zo staat de chip in verbinding met de andere componenten van het apparaat en krijgt de chip stroom. Hoe ingewikkelder de chip, hoe meer contactpinnetjes er zijn. Door de stroom op bepaalde pinnetjes aan en andere uit te zetten, kunnen de transistoren in een bepaalde stand gezet worden. Chips kunnen dat schakelen alleen doen als er spanning op de voedingspin staat.

Er bestaan twee belangrijke typen chips. *Logische chips* kunnen informatie verwerken en geprogrammeerd worden, zoals de centrale processor (CPU) van een computer. Bepaalde software is vast in het geheugen van de chip opgeslagen (*firmware, embedded software*) zodat die snel beschikbaar is. *Geheugenchips* houden informatie vast. Het snelle DRAM geheugen doet dat alleen als er spanning op de chip staat. Het trage NAND geheugen, zoals een flashkaartje, houdt informatie ook vast als er geen spanning op de chip staat.

4.1 Vraag

Zoek op hoeveel transistoren er op een Intel I7 processor zitten. Hoe groot is één transistor? Om te berekenen nemen we aan dat er 20 lagen op de chip zitten en het oppervlak van de chip 1 cm^2 . Hoe groot is het oppervlak van 1 transistor in nm^2 ?

Een chip verwerkt opdrachten in een vaste regelmaat: de kloksnelheid. Op iedere tik van de klok kan de computer één actie uitvoeren. In een computerchip zit een *kristal* dat met een bepaalde frequentie trilt (in de orde van MHz of GHz) en daarmee bepaalt de chip de tijdsduur en dus zijn kloksnelheid. Die kan bijvoorbeeld verlaagd worden om stroom te besparen. Hoe hoger de frequentie, hoe meer verwerkingsstappen een chip per seconde aan kan.

4.2 Vraag

Hoe groot is de kloksnelheid van de processor die in jouw telefoon zit? Als je het type telefoon kent, kun je opzoeken welke processor daarin verwerkt is. Vergelijk deze kloksnelheid met de Intel i7 processor. Wat valt je op?

Wat zou er nog meer een rol kunnen spelen bij de 'snelheid' van zo'n apparaat?

Hoe moet je het programmeren van een chip nu voorstellen? Programmeren is vergelijkbaar met omzetten van schakelaars. Dat gebeurt bijvoorbeeld in de cockpit van een vliegtuig. De piloten zetten handmatig schakelaars in een bepaalde stand voor vertrek, tijdens de vlucht en na de landing. In het vliegtuig gaan dan allerlei systemen aan of uit. Hoe de schakelaars moeten staan, hebben ze in een checklist staan. Je kunt dat vergelijken met het programma, de software.

Ook het programmeren van een chip is weinig anders dan het omzetten van schakelaars. De software programmeert de chip en zet ook de schakelaars om. Dat zijn in een chip de transistoren. Een pinnetje 'aan' schakelen doe je door er spanning op te zetten (bijvoorbeeld 3V). 'Uit' is dan 0V. Dat is in digitale waarden: 1 en 0.



Figuur 4.6 Het schakelbord in een cockpit vergeleken met de pinnen van een ATMEGA 32P microcontroller

Een programma zorgt er voor dat een chip taken kan uitvoeren. De afzonderlijke stappen in een programma worden *algoritmen* genoemd. Als het beginpunt en eindpunt gelijk is, hoeven de tussenliggende stappen niet hetzelfde te zijn. Ofwel, in veel gevallen zijn er *meerdere* algoritmen te bedenken met dezelfde uitkomst. Een slimme volgorde van handelingen kan een sneller resultaat opleveren. Dat betekent dat het uit maakt *hoe* je het algoritme schrijft en welke algoritmen je voor het programma gebruikt. Dat is de kunst van het elegant programmeren. Terug naar de cockpit: het is wel handig om de knoppen niet kriskras door elkaar te bedienen, dat kost nodeloos extra tijd. Daarom is bij een goede checklist de volgorde zo efficiënt mogelijk.

4.3 Vraag

Aan welke eisen zou een *elegant* algoritme moeten voldoen? Welke technieken zijn er om algoritmen efficiënt te maken?

Op het niveau van de chip moet het algoritme vertellen wat de chip moet doen met de spanning op ieder van de pinnetjes. Maar programmeren door per pinnetje op te schrijven wanneer die 1 of 0 moet zijn, is voor een mens niet te doen. De reeksen 1 en 0 die de chip kan verwerken is *machinetaal* (*machine code*). Dat is voor mensen niet te volgen. Vandaar dat er programmeertalen zijn verzonnen die makkelijker te begrijpen zijn en eigenlijk een hele serie handelingen of algoritmen met één commando kunnen geven (*assembler code*). De computer rekent dit zelf om naar de algoritmen in machinetaal. Dat gebeurt door de *compiler*. De hogere orde programmeertalen zijn voor mensen beter begrijpelijk, maar staan verder af van de machinetaal.

We kunnen niet van buitenaf zien hoe de schakelaartjes in de computerchip staan. Daardoor kunnen we ons ook moeilijk voorstellen wat een programma met een chip doet en zijn we afhankelijk van informatie die we uit de chip terugkrijgen en vertalen naar iets begrijpelijks.

Er zijn tientallen programmeertalen ontwikkeld, die voor andere toepassingen bedoeld zijn. De Arduino-taal, Python of C++ zijn voorbeelden. Bepaalde talen zijn geschikt om apparaten aan te sturen, andere om plaatjes te bewerken of lange rijen getallen te bewerken.

4.4 Opdracht: Programmeertalen

Zoek voor de programmeertalen van Arduino, Python, C++ en Matlab op voor welke toepassing ze gemaakt zijn en wat ze daar geschikt voor maakt. Zoek dit ook uit voor twee andere programmeertalen.

Als een computerprogrammeur een programma maakt dat werkt, maar dat een ander niet kan begrijpen, is dat niet handig. Slordigheid kost heel veel rekentijd, fouten opsporen en verlies van kennis. Daarom is het plezierig om gebruik te kunnen maken van stukken code die goed werken, maar die je niet zelf hoeft te verzinnen. Er zijn allerlei databases met computercode beschikbaar. Programmeurs halen daar *blokken* code uit waarmee ze snel een werkend programma kunnen maken. Je hoeft van de code uit een blok niet precies te weten hoe die in elkaar zit. Als die code ook nog eens als een (gekleurd) blokje wordt weergegeven, zie je echt 'blokken-taal' zoals je die bij veel populaire systemen tegenkomt.

4.5 Opdracht: Blokken programmeren

Blokken programmeren wordt toegankelijk, maar het maken van de blokken is een ingewikkeld proces. De arduino-taal, Micro:bit, M-bot, python, Scratch en Tinkercad maken gebruik van beide technieken. Zoek van beide typen een voorbeeld (blok-programmeren naast regel-voor-regel).

Programmeren van de microcontroller

Als je met een microcontroller gaat werken, zoals Micro:bit of Arduino, dan programmeer je in Arduino-taal. Jouw Arduino-programma wordt door de software in je computer (de *compiler*) omgezet naar machinetaal voor de Arduino en naar het bordje gestuurd. Daarmee gaat het bordje de taak uitvoeren. Zolang er geen nieuw programma komt blijft het bordje dezelfde taak uitvoeren. Als je het geheugen wist (dat kan met de RESET knop), is het programma verdwenen en doet het bordje niets meer. Opnieuw moet je vanuit je computer het programma in machinetaal kopiëren om het bordje te laten werken.

Over computerchips en programma's is veel meer te ontdekken. Bijvoorbeeld: de architectuur van processorchips met 4, 8, 16, tot 64 bit of hoger, parallele of seriële processen, de opbouw van programmeertalen, hoe kunstmatige intelligentie via neurale netwerken geprogrammeerd wordt, opbouw van schakelingen, het bewerken van beelden of de manier waarop een sporthorloge zijn verzamelde data verwerkt en doorstuurt. Hiervoor kun je terecht in verdiepende (vervolg)modules.



Deel B Context: Kunstmatige Intelligentie

Software kan zichzelf leren programmeren. Dit gebeurt bij Kunstmatige Intelligentie of diep-lerende systemen. Die zijn bijzonder geschikt voor het herkennen van patronen.

4.3 Artificial Intelligence (AI)

Bij kunstmatige intelligentie (Artificial Intelligence, AI) maakt de software *zelf* allerlei instellingen die we niet kunnen zien. Dat gebeurt in programma's die kunstmatige neurale netwerken (Artificial Neural Networks, ANN) genoemd worden. Hierdoor kan de software *leren* door voortdurend patronen te vergelijken. De software kan zo bijvoorbeeld beelden herkennen, zoals bij de gezichtsherkenning van een fotocamera of het zoeken naar bepaalde vormen in afbeeldingen. Doordat de computer *zelf* de algoritmen aanpast, is het ook niet meer te achterhalen hoe die precies in elkaar zitten. Daarom verzinnen wetenschappers methoden om de algoritmen na te bouwen en te vergelijken met wat AI heeft gedaan. Hoe AI werkt, kun je in een *vervolgmodule* ontdekken (zie H8).

Kunstmatige Intelligentie: heel geschikt voor het leren herkennen van beelden.

Zoek Wally! Een bekende tekenaar van plaatjesboeken voor kinderen tekende in iedere afbeelding van een boek het figuurtje Wally. Voor de lezers was het de uitdaging om die te ontdekken. We zijn als mensen visueel ingesteld, en heel behendig in het analyseren van beelden, maar het herkennen van wat we zien vraagt oefening. Zeker als het ingewikkelde en niet-alledaagse beelden zijn. In een foto herkennen we snel dat er menselijke gezichten op staan. Maar het is een stuk lastiger om elke afgebeelde persoon te herkennen, daarvoor kennen we te weinig mensen. Computers die gezichten kunnen herkennen hebben de mogelijkheid om in grote databases te zoeken en razendsnel vergelijkingen te maken tussen het afgebeelde gezicht en de afbeeldingen in de database. Door de verbeterde rekenkracht van computers wordt gezichtsherkenning steeds meer toegepast. Beveiligingscamera's, toegangspoortjes, toegang tot je telefoon: AI met gezichtsherkenning is wijdverspreid.



Figuur 4.7 Deze lijkt op Wally

Wat met gezichten kan, is ook mogelijk met alle andere patronen. Herkennen van planten, vogels, verkeersborden, de weg, obstakels, kwaadaardige cellen in een orgaan, pakketjes in een sorteercentrum of de boodschappen in een mandje: voor de computer is het allemaal even ingewikkeld. Voor ons is dat heel nuttig: in de zelfrijdende auto, een app die je vertelt welke vogel je ziet, robots die automatisch je boodschappen of pakketjes bezorgen. In medische centra wordt veel onderzoek gedaan naar het herkennen van afwijkingen in medische beelden. Is dit plekje een onschuldige bultje of een kwaadaardig gezwel? Pathologen moeten die vraag elke dag tientallen keren beantwoorden als ze stukjes weefsel moeten onderzoeken tijdens operaties. Kan de computer artsen helpen om antwoord te geven op deze vraag? Zal hen dat helpen om betere diagnoses te stellen?

Kunstmatige Intelligentie om tumoren te herkennen: het onderzoek van Daan Geijs

Een jaar of tien geleden zat Daan op de middelbare school. Hij had een natuurprofiel, maar volgde niet de juiste vakken om de studie te doen die hij graag wilde: biomedische technologie. Daarom is hij wiskunde B en natuurkunde gaan bijspijken om toch in Twente te gaan studeren. Inmiddels afgestudeerd als biomedisch ingenieur én met een lerarenopleiding natuurkunde, is hij aan de slag gegaan als promovendus bij de medische faculteit van de Radboud Universiteit. Hij werkt aan een

promotieonderzoek om kunstmatige intelligentie (patroonherkenning door computers) in te zetten bij het opsporen van tumoren. Preciezer gezegd: om een systeem te ontwikkelen dat een patholoog ondersteunt bij het bekijken van weefselplakjes om te ontdekken of er kankercellen aanwezig zijn. Bedenk: van ieder verdacht plekje bij elke operatie moet een patholoog een stukje weefsel in plakjes snijden en beoordelen op de aanwezigheid van kankercellen. Dat is veel werk én het vraagt de deskundigheid van een hoog opgeleide specialist. Kan je een computer leren om in foto's van die plakjes weefsel de kankercellen er uit te pikken? Kan een computer ingezet worden om de patholoog het vermoeiende routinewerk uit handen te nemen?



Figuur 4.8 Daan Geijs vertelt over de toepassing van digitale technologie in het ziekenhuis

Daan heeft een aantal filmpjes opgenomen waarin hij vertelt over zijn opleiding, wat hij nu onderzoekt, wat kunstmatige intelligentie is en waar hij na zijn promotie zou kunnen gaan werken. Deze zijn terug te vinden op de leerlingsite

(https://maken.wikiwijs.nl/171772/Schakelmodule_Digitale_Technologie_online_materiaal)

4.6 Opdracht: Werken met Kunstmatige Intelligentie

Bekijk de filmpjes van Daan Geijs en beschrijf in eigen woorden wat je hebt ontdekt over

1. Opleidingen die je na de middelbare school kunt doen
2. Wat Kunstmatige Intelligentie is en hoe dat onderverdeeld wordt
3. Hoe Kunstmatige Intelligentie gebruikt zou kunnen worden in het ziekenhuis
4. Bij welk soort bedrijven en instellingen Daan terecht kan met zijn kennis en ervaring



Deel C Zelf doen

Het maken en programmeren van een koortsthermometer met Arduino is een stapgewijze introductie in de programmeertaal.

4.4 Maken en programmeren: digitale thermometer met Arduino

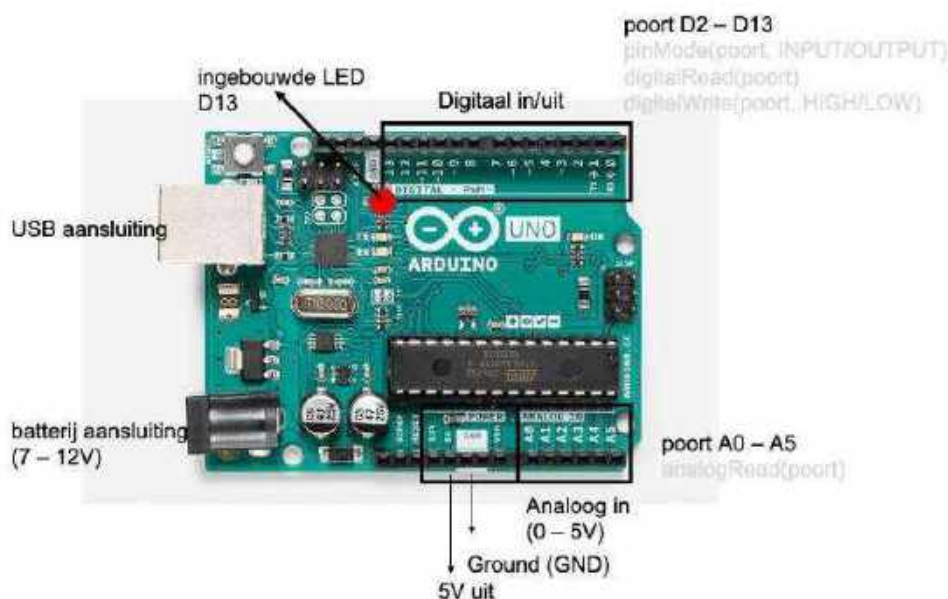
In deze opdracht ga je zelf ervaring opdoen met bouwen en programmeren van een Arduino (of vergelijkbaar device): een digitale (koorts)thermometer. Deze meet de temperatuur en toont de temperatuurwaarde op een display. Misschien denk je: “wat heeft dat voor zin? Ik kan voor weinig geld ook zo’n ding kopen”. Klopt, maar een thermometer is maar een voorbeeld van wat je allemaal met een Arduino kunt maken en doen. Misschien krijg je zelf een heel goed idee en wil je dat gaan maken! Het is dan handig om te weten hoe je dat aanpakt. Daarom hieronder eerst een paar stappen die je in *ieder* project zet. De volledige stap-voor-stap opdracht is in een bijlage beschreven die verkrijgbaar is in de moduledatabase.

Stap 1: Kiezen van de hardware

Voor Arduino zijn veel verschillende onderdelen beschikbaar. Als je ze wilt laten doen wat je wilt, moet je de juiste onderdelen kiezen voor je project. Voor de digitale koortsthermometer wil je temperatuur meten, verwerken en laten zien. Je hebt in elk geval een temperatuursensor, schakelaar, Arduino-bordje met voeding, schermpje (display) en LED nodig.

Wanneer je met andere spullen aan de slag wilt (zoals de Raspberry Pi, Micro:Bit, M-block of nog andere) moet je ook hier de juiste selectie maken voor je project. Vaak zijn sensoren en andere onderdelen door elkaar te gebruiken, mits de aansluitingen passen. Losse steekpennetjes zijn vaak vervangen door connectoren om onderdelen foutloos aan te sluiten.

De Arduino Uno



Figuur 4.9 Het bordje van de Arduino Uno

In figuur 4.9 zie je een afbeelding van een Arduino Uno. Er zijn meerdere soorten Arduino's, maar dit is de belangrijkste. Misschien heb je een kloon die er iets anders uitziet, maar dat maakt niet uit, de aansluitingen zijn hetzelfde. Je moet de Arduino zien als een klein computertje. Op het printplaatje zit een processor (net zoals in je telefoon) voor alle taken. Aan die Arduino sluit je sensoren, display, lampjes etc. aan. Dat doe je op de zwarte insteekpennetjes.

De aansluitingen:

1. USB aansluiting: Hiermee koppel je de Arduino aan de computer
2. Batterij aansluiting: De Arduino kan niet werken zonder spanning. Als de USB is aangesloten, dan krijgt de Arduino daarvan de spanning. Wil je zonder USB werken, dan kun je een batterij/accu/voedingskast aansluiten tussen 7 – 12V
Vaak hebben de sensoren ook een spanning nodig. De 5V aansluiting levert die spanning (5V dus). De GND (= Ground) is de min-kant. Het maakt niet uit of de batterij of de voeding uit USB gebruikt wordt.
3. (rechtsboven) digitale in- en uitgangen: Dit zijn in- en uitgangen die een 0 of een 1 kunnen "lezen" (read) of "schrijven" (write). Dat kun je zelf bepalen. Je kunt hiermee een stand van een schakelaar (aan/uit) lezen (read) of een lampje (LED) aan/uit zetten (write). Hoe dat werkt wordt hieronder verder uitgewerkt. In totaal zijn er 11 van dit soort aansluitingen, genummerd D2 t/m D13 (Digitale poorten).
4. (rechtsonder) analoge ingangen, genummerd A0 t/m A5 (poorten): Dit zijn 6 bijzondere poorten die een spanning (tussen 0 en 5V) kunnen omzetten in een getal. Deze heten AD-omzetters: Analooog (spanning) naar Digitaal (getal) omzetter. Deze moeten we gaan gebruiken om de sensor te gaan lezen en kun je dus ook alleen gebruiken om te "lezen".
5. De Uno heeft een ingebouwde LED, gekoppeld aan D13.

Stap 2: Opzoeken van de juiste drivers

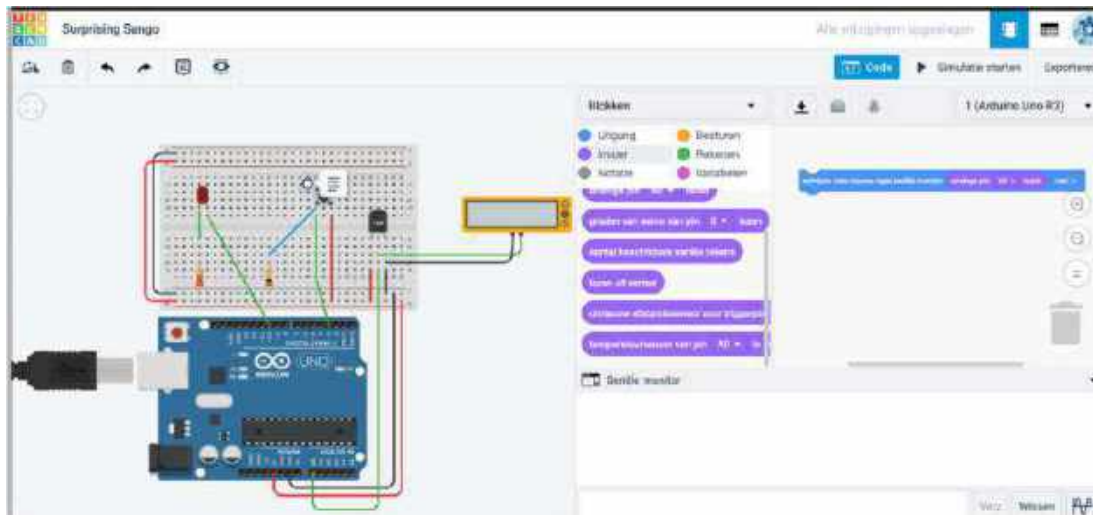
Als je een onderdeel, zoals de temperatuursensor, op een ingang van de Arduino aansluit moet je de Arduino informatie geven. Op welke pinnen is deze aangesloten? Komt er een digitaal of een analoog signaal uit? Wat betekent een waarde die het onderdeel doorgeeft of nodig heeft? Al deze informatie zit verpakt in stukjes code, de *drivers*. Op internet staan bibliotheken (*libraries*) met drivers voor al die sensoren en andere bouwstenen. Om die op te zoeken, heb je de naam van een onderdeel nodig, zoals de TMP36 (temperatuursensor). Wanneer je zelf een sensor of actuator bouwt, moet je zelf dit soort informatie bij elkaar zoeken en in een infobestandje zetten voor de Arduino.

Stap 3: Aan de slag, eventueel zónder te bouwen

Om de Arduino te vertellen wat die allemaal moet doen met die in- en uitgangen moet je deze programmeren. Je moet dus eigenlijk een app maken voor de Arduino. Dat doe je op de computer. Als het programma klaar is, stuur je het via de USB naar de Arduino en die gaat vervolgens doen wat je geprogrammeerd hebt. Het programmeren doe je in de *Arduino IDE*.

Het bouwen en programmeren van een apparaat kan een stuk eenvoudiger als je dit vooraf kunt samenstellen en testen met je computer. Voor Arduino kan dat heel mooi in een online omgeving: TinkerCad (ga naar tinkercad.com en maak een account aan).

Je kunt hier met plaatjes je apparaat samenstellen en de code (sketch) schrijven (in blokkentaal, of in de programmeertaal C). Je kunt het programma opslaan, downloaden en in de echte Arduino laden (*.ino bestand). Je kunt ook de Arduino-IDE (Integrated Development Environment) gebruiken om je sketch te maken of een bestaande sketch aan te passen. Als je het fysieke apparaat bouwt en de sketch laadt (via de Arduino IDE), doet deze wat je al uitgetoetst hebt. Tenminste: dat is wat je kunt verwachten. Uiteraard moet je testen of dat echt zo is. Daarna kun je de echte metingen gaan doen.



Figuur 4.10 In TinkerCad kun je virtueel een device bouwen (links de hardware en aansluitingen) en het programma in blokken weergeven (midden). De coderegels kunnen ook via een venster getoond worden. Met een knop bovenaan het scherm kan de simulatie gestart worden.

Stap 4: Kalibreren, testen en verbeteren

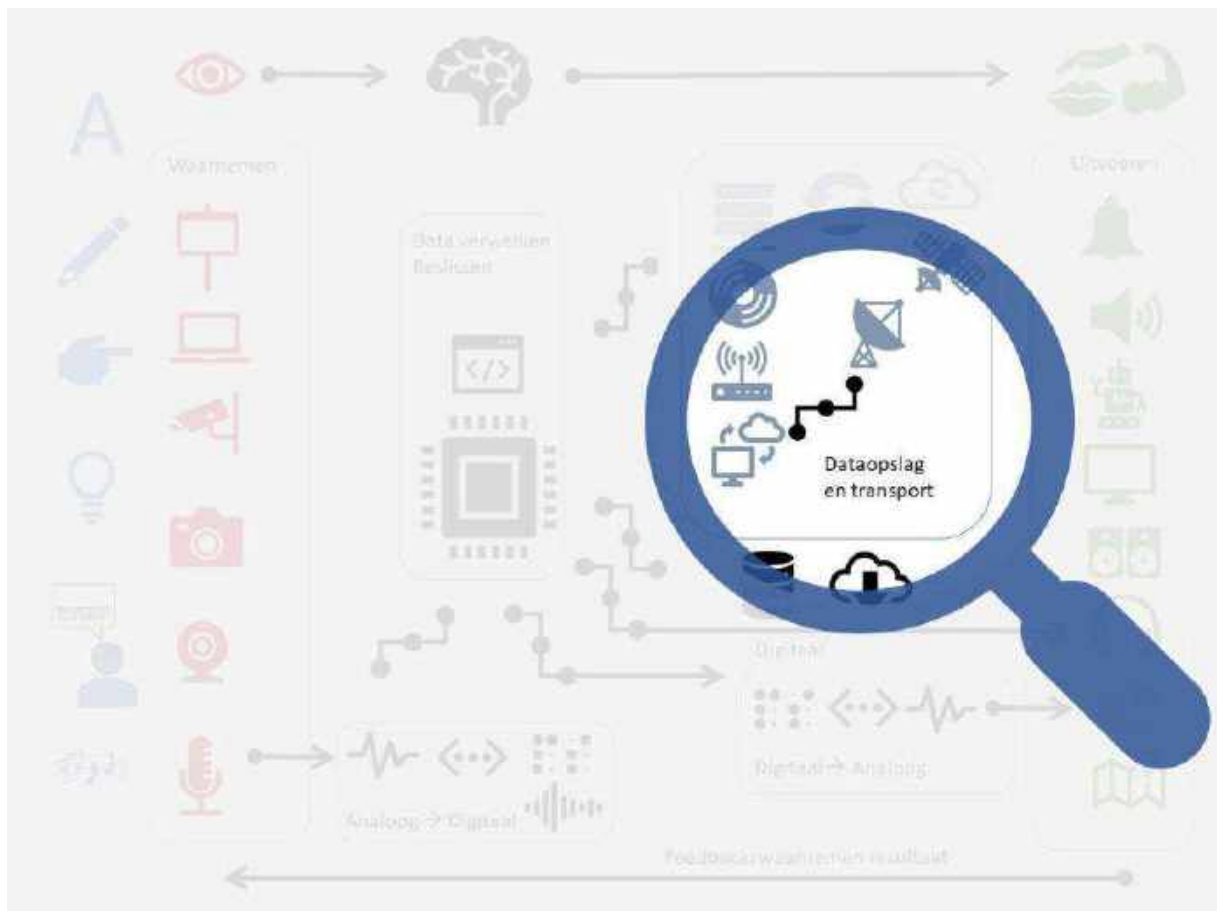
Als je een apparaat hebt gemaakt dat metingen doet, is kalibreren een belangrijk onderdeel. Een waarde uit de sensor (tussen 0 en 1024) moet omgerekend worden naar een correcte waarde op het scherm. In TinkerCad kun je ook het kalibreren van je sensor uitvoeren, zodat je de juiste code kunt schrijven. Uiteraard controleer je die metingen met de fysieke Arduino nog wel.

Wanneer je metingen opslaat of verstuurt, is het ook nodig om te weten wat er gemeten is en op welk tijdstip. Daarvoor moet je ook een timer op de Arduino aansluiten. De gegevens kunnen op een SD-kaartje gezet worden of via een USB-kabel of draadloos via Bluetooth, WiFi of LoRaWan naar een ander apparaat verstuurd. Opslag van data komt in deze opdracht nog niet aan bod, maar wel in die van Q-strip in H7.



Figuur 4.11 Een stukje sketch (code) voor Arduino (in de taal C). Achter de // tekens staat uitleg.

5 Data opslag en transport



Leerdoelen

Kennis

- 5.1 Je kunt uitleggen dat er vuistregels zijn voor het ordenen van gegevens binnen bestanden, het ordenen van bestanden en dat het gebruiken van die vuistregels je tijd bespaart en de kans op fouten vermindert.
- 5.2 Je kunt uitleggen dat metadata de informatie bevat over wat er in bestanden te vinden is en hoe deze geordend zijn.
- 5.3 Je kunt uitleggen dat gegevens fysiek opgeslagen worden op harde schijven (in het klein) of in datacenters (in het groot) en hoe deze opslag functioneert
- 5.4 Je kunt uitleggen dat gegevens op verschillende manieren beveiligd kunnen worden door wachtwoorden en encryptie en dat de wiskunde hier een grote rol in speelt.
- 5.5 Je kunt uitleggen dat het internet een combinatie is van ip-adressen en TCP/IP protocollen en hoe dit werkt

Vaardigheden

- 5.6 Je hebt gewerkt met een apparaat dat informatie naar een website kan sturen zodat je op afstand mee kunt kijken met de sensoren die jouw digitale systeem bevat.



5.1 De 'cloud'

Met alleen maar een mobiele telefoon heb je eindeloos veel informatie beschikbaar. Het versturen en ophalen van data gaat snel, draadloos en onzichtbaar via de 'cloud'. Je deelt bestanden zonder dat je er iets extra's voor hoeft te doen. De naam 'cloud' suggereert dat het ergens in de lucht gebeurt, maar niets is minder waar. Overal in de wereld staan grote en kleine gebouwen zonder ramen, vol met computers en harde schijven: datacenters. Digitale data is heel geschikt om snel en zonder verlies te transporteren. Dat gebeurt met draadloze wifi en 4G / 5G signalen en kabels van je mobiel naar datacenters en weer terug. Goede beveiliging is ook vereist. Achter dataopslag en datatransport zit een heleboel techniek die we in dit hoofdstuk beter gaan bekijken.

Digitale signalen

Digitale technologie gebruikt de binaire code die twee (= bi) waarden kan aannemen: 0 of 1. Dat is in een elektrische schakeling hoge (5V) of lage (0V) spanning, met licht door helder of zwak of zwarte en witte lijnen of vlakjes. Informatie wordt gecodeerd in reeksen nullen en enen.

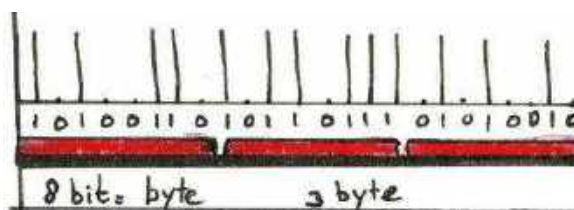


Figuur 5.1 Opslag en transport van data

5.1 Vraag

Zoek enkele voorbeelden van digitale codes die op spullen, sites of drukwerk te vinden zijn. Welke soorten codes kun je terugvinden?

Met een rijtje van 2 binaire waarden kun je 4 combinaties maken: 00, 01, 10 en 11. Dat is 2^2 . Een rijtjes van 8 bits (een byte) wordt veel gebruikt. Dit geeft 2^8 combinaties, ofwel 256. Met langere reeksen kunnen meer verschillende waarden worden gecodeerd. De TMP-36 temperatuursensor stuurt een *analoog* signaal naar de Arduino, dat omgezet wordt in discrete (stapgewijze) decimale waarden 0 tot 1024. (Ga eens na hoeveel bits er voor elke waarde nodig zijn).



Figuur 5.2 Een byte is uit een rijtje van 8 bits. Deze 24 bits zijn in 3 byte verdeeld. 8 bits geeft 256 combinaties.

Omzetting van analoog naar digitaal geeft wel wat *verlies* (immers: bij een reeks verschillende temperaturen geeft de sensor hetzelfde getal). Wanneer er eenmaal een discreet digitaal getal is kan dat goed *behouden* worden. Immers, een spanning van 3, 4 of 5 volt is nog steeds 'hoog' en dus 1. Het maakt ook niet meer uit of dit nu via elektrische spanning, radiogolven of lichtpulsen verstuurd wordt. Data kan zo zonder verlies over lange afstanden worden vervoerd.

Nauwkeurigheid

Precies meten van de temperatuur, bijvoorbeeld met twee cijfers achter de komma, betekent dat je digitale sensor voldoende stapjes moet kunnen weergeven. Een 8-bit temperatuursensor kan niet meer dan 256 verschillende waarden weergeven. Voor een bereik van 0 en 100 °C zijn er iets meer dan 2 waarden per graad beschikbaar. Dat is onvoldoende. Je moet dan bijvoorbeeld een 16-bit sensor gebruiken, die 65.536 waarden kan onderscheiden. Hoe meer bits, hoe nauwkeuriger er te

digitaliseren is. Maar ook: hoe meer rekenkracht *de processor* moeten hebben om een signaal te verwerken. Het verzamelen van data betekent dat je bedenkt: hoe *nauwkeurig* en hoe *vaak* wil ik meten?

5.2 Vraag

Hoeveel data produceert een 16-bit sensor in één dag? Geef je antwoord in Mb.

Data bewerken, opslaan en transporteren



Figuur 5.3 Opslag en uitvoer van digitaal signaal



Figuur 5.4 Verwerking van digitale gegevens in de processor

Het elke seconde uitlezen van de 8-bit sensor levert 1 byte data per seconde, 300 byte in 5 minuten en 86 kilobyte per dag. Dat zijn relatief bescheiden hoeveelheden. Een 5 Megapixel camerasensor levert plaatjes van vele megabytes (Mb, 10^6) waar datatransport en opslag in de orde van gigabyte (Gb, 10^9) of terabyte (Tb, 10^{12}) voor nodig is. Wanneer sensoren aan staan, produceren ze data. Hoe vaak je die data ophaalt en verwerkt bepaal je in de software (hoe nauwkeurig en hoe vaak meet je?). Bij het programmeren daarvan moet je na denken over de hoeveelheid data die *nodig* is en dus de datastroom die ontstaat. Hoe meer data verplaatst moet worden, hoe groter de *bandbreedte* moet zijn om dat snel voor elkaar te krijgen. De bandbreedte wordt meestal uitgedrukt in

bits per second (bps). Ook *compressie* van data is belangrijk. Je vervangt een reeks dezelfde waarden door een (veel kortere) reeks. Daarin staat één keer de oorspronkelijke waarde, hoe vaak die herhaald is en waar het begin en einde zijn. Dat gebeurt met een *compressie-algoritme*. Sommige algoritmes gooien daarmee ook informatie weg. Bij foto's ken je dat als .jpg, bij video als .mp4 en bij geluid als .mp3. Andere algoritmes zorgen dat de oorspronkelijke informatie te herstellen is: 'lossless' compressie. Hier is veel meer over te leren in andere modules (zie hoofdstuk 8).

5.3 Opdracht

- Test eens uit wat de bandbreedte (bandwidth) is van de internetverbinding van een mobiel en een laptop (via Wifi én met een bekabelde Ethernet aansluiting). Ga daarvoor met de mobiel of laptop naar de website [speedtest.net](https://www.speedtest.net).
- Vergelijk de bandbreedtes met elkaar. Verklaar het (eventuele) verschil dat je ziet.

In de computerwereld draait alles om data, datatransport en dataopslag. Aan alleen maar een reeks enen en nullen heb je niet zoveel. De processor moet weten waar de reeks enen en nullen vandaan komt, wat er mee moet gebeuren en waar die na bewerking naartoe moet. Daarom is het belangrijk dat bekend is waar die data over gaat. We noemen dit *metadata*. Vergelijk het met de informatie:

Geheugen

Digitale systemen gebruiken geheugenchips en opslagmedia (harddisk, optische schijven, solid state disks of flash geheugenkaarten) om bits weg te schrijven. Een geheugen in de computer werkt met *transistoren* (schakelaars) of *condensatoren*. Een aantal *transistoren* kunnen samen zo verbonden worden dat bij de uitgang van de schakeling spanning blijft staan (1) of juist niet (0). Deze flip-flop schakeling vormt dan een geheugencel. Met een paar miljard van deze schakelingen bouw je een geheugenchip. Dit kan ook met *condensatoren*. Als je daar elektronen in opsluit, blijft de spanning hoog (1). Een lege condensator heeft dan de waarde (0). Elektronen lopen langzaam uit de condensator, zodat de geheugenchip telkens moet controleren (actief, dynamisch) of de spanning hoog genoeg is (en voegt zo nodig lading toe). Het geheugen is wel supersnel, waardoor dit gebruikt wordt als werkgeheugen (Dynamic RAM).

‘ja’. Daar heb je weinig aan als je de vraag en de context niet kent. Het antwoord wordt anders als je de vraag weet: wil je suiker in je thee? Een *database* bevat tabellen met data én metadata. Daar kun je in zoeken, data uit lezen, bijschrijven en wissen.

Een bestand (*file*) heeft allerlei informatie dat vertelt waar het bestand over gaat en hoe het in elkaar zit. Vaak zijn files zo groot dat ze in blokjes moeten worden opgeslagen en bewerkt. De processor heeft van elk blokje een *adres* en zet dat in het geheugen. Blokjes worden van het ene naar het andere deel van het geheugen verplaatst of in de processor bewerkt en weer opgeslagen. De harde schijf gebruikt ook *adressen* om bij te houden waar ieder datablokje is opgeslagen. Op een harde schijf hoeven de blokjes van één file niet achter elkaar te staan. De schijf zet ze neer waar ruimte is. De schijf moet dus precies bijhouden welke datablokjes samen het digitale bestand vormen dat je hebt opgeslagen.

De computer schrijft en leest de hele tijd van werkgeheugen naar massageheugen (harde schijf of ssd) of stuurt en ontvangt de blokjes data over de netwerkverbinding en het internet. Bij het heen en weer sturen over het netwerk moet er nóg meer informatie toegevoegd worden om het datablokje op de juiste plek te krijgen. Voor die taken heeft de computer een netwerkkkaart of wifi-controller. Je mobiel gebruikt daar een wifi, bluetooth of 4/5G controller voor. Die is in staat de data heen en weer te sturen naar het netwerkknooppunt (router, switch) en vandaar naar het internet.

Voor het heen-en-weer sturen van datapakketjes moeten computers verbinding met elkaar hebben, met koperdraad, glasvezel of draadloos.

5.4 Vraag

Hoeveel opslag heeft jouw mobiel? Maak onderscheid tussen werkgeheugen (RAM) en het opslag (ssd) geheugen. Hoe zit dat met je computer?

5.2 Verbindingen

Metaaldraad

Elektriciteit loopt via metaal en dat is dus ook de manier om elektrische digitale signalen te vervoeren. De snelheid waarmee dat gebeurt is pakweg 0,1 x de lichtsnelheid. Een elektrisch signaal verzwakt over grotere afstanden, waardoor er versterkers nodig zijn. De standaard voor digitale elektrische signalen is de UTP-kabel, die bijvoorbeeld ook gebruikt wordt om in huis router en computer te koppelen.

5.5 Vraag

Hoe werkt die UTP kabel? Waarom heeft die meerdere draden?

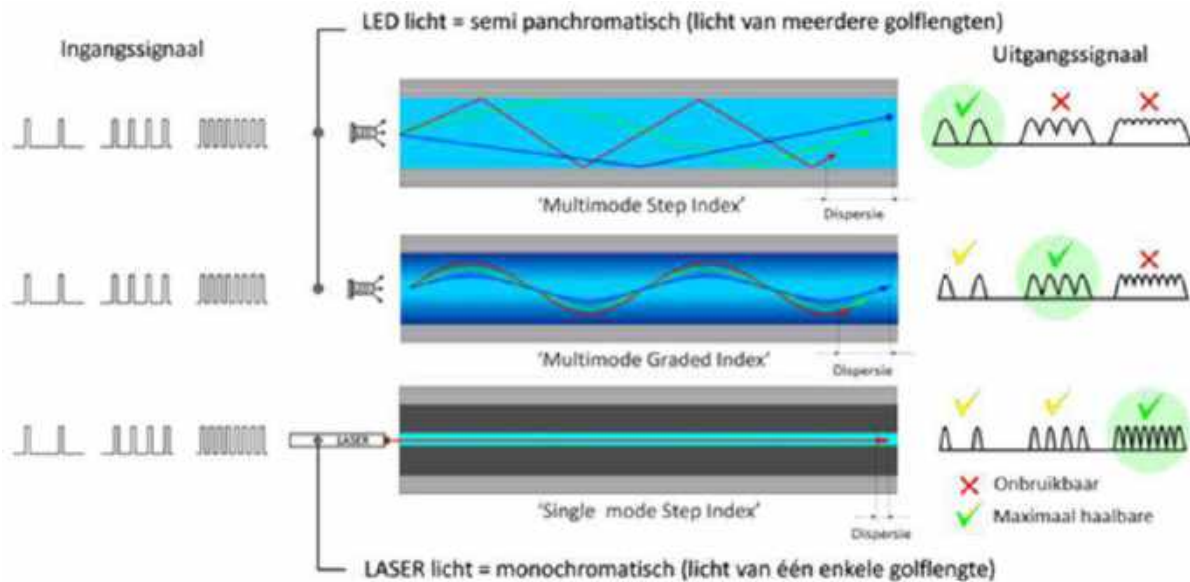
Sneller: glasvezel

Het sneller en over langere afstanden transporteren van digitale informatie gaat tegenwoordig via glasvezel. Hoewel met een iets lagere snelheid dan lichtsnelheid in vacuüm, is het transport wel sneller dan in metaal. Er is bovendien minder verzwakking van het signaal over langere afstand.

5.6 Vraag

Zoek op over welke afstand een glasvezelkabel een signaal kan versturen zonder versterker. Zoek ook op welke techniek(en) men toepast om verschillende datastromen in één glasvezel te vervoeren.

Voor de glasvezel worden lasers gebruikt (die een digitaal signaal omzetten in lichtpulsen) en fotodiodes (die het digitale lichtsignaal weer in stroom omzetten). De lasers kunnen heel klein zijn, en zelfs als onderdeel van een computerchip worden gebouwd. Daardoor lukt het ook om tussen computers de glasvezel voor communicatie in te zetten. Een stap verder is de ontwikkeling van een optische computer waar alle bewerkingen met licht gebeuren.



Figuur 5.5 Kwaliteit van datatransport met licht in glasvezels

Meer bronnen over glasvezel:

<https://www.deingenieur.nl/artikel/nederlandse-bedrijven-pionieren-met-chips-op-licht>
<https://www.deingenieur.nl/artikel/fotonica-sector-krijgt-duw-in-rug>
<https://www.deingenieur.nl/artikel/doorbraak-silicium-zendt-licht-uit>
<https://nl.wikipedia.org/wiki/Glasvezel>
https://en.wikipedia.org/wiki/Laser_diode

Nog sneller? Draadloos.

Wifi, bluetooth, mobiele data (3, 4 en 5G) of satellietcommunicatie werken met radiogolven die zich met lichtsnelheid verplaatsen. De omzetting van een digitaal signaal naar een radiogolf en terug kost tijd en rekenwerk, waardoor weer snelheid verloren gaat. Er zijn zendmasten en schotelantennes nodig, die weer met (glasvezel) kabels aan datacenters gekoppeld zijn.

5.7 Opdracht

Er is veel te vinden over draadloze communicatie. Als je in een groepje werkt: kies per groepslid één van de technieken uit en verzamel daar informatie over. Maak daarvan een overzichtelijke tekening / kleine poster om uit te leggen hoe de techniek werkt. Geef informatie over snelheid, bandbreedte (hoeveel data per seconde) en afstand die overbrugd wordt. Hoe groot zijn zender en ontvanger van het signaal? Worden er gevaren of schadelijke effecten genoemd?

5.3 Adresseren

Elke computer in een netwerk heeft een eigen adres: een toegewezen IP (v4 of v6) en een MAC (uniek apparaatnummer). Een datapakketje krijgt adressen mee: waar komt het vandaan (afzender IP) en waar moet het naartoe (ontvanger IP). Het internetprotocol (TCP/IP) beschrijft hoe pakketjes

over het netwerk vervoerd moeten worden. Andere protocollen (als http://) sturen de datapakketjes in de computer zelf naar de juiste plek (bijvoorbeeld de webbrowser).

Het eerste knooppunt leest het IP adres en bepaalt naar welk volgend punt het verstuurd wordt. Het eerste stukje van het adres ("niet mijn netwerk") is al voldoende om het door te sturen. Een netwerk in huis heeft vaak een IP-adres dat begint met 192. Dat betekent dat de router het pakketje niet naar buiten stuurt, maar in het eigen netwerk zoekt. Je internetrouter thuis heeft voor ieder apparaat een eigen laatste getal in de IP-reeks: bijvoorbeeld 192.168.2.0 voor de router en 192.168.2.12 voor jouw computer. De router die in huis staat, heeft een IP-adres voor binnen (LAN) én een voor buiten (WAN). Als je een website bezoekt, gaat je browser data van een webserver ophalen. Je tikt een adres in, bijvoorbeeld <https://www.verenigingnlt.nl>. Met dat adres kun je nog niet de juiste server bereiken. Daarom gaat het pakketje eerst naar een DNS-server (dynamische naamserver), die van de website-naam een IP-adres maakt. Nu kan het verzoek (ophalen van de webpagina) bij de juiste webserver worden bezorgd. De webserver stuurt vervolgens pakketjes terug naar jouw computer. Op die route liggen allerlei tussenstations (knooppunten, *hubs*) die supersnel het IP-adres lezen en het pakketje doorsturen. Dat pakketjesverkeer gaat zo snel, dat je niet merkt dat de computers waar ze vandaan komen aan de andere kant van de wereld kunnen staan. Voor dat dataverkeer zijn overal ter wereld knooppunten en datacenters ingericht.

Internet knooppunten

Het Internet Knooppunt Amsterdam, op de campus van de Universiteit van Amsterdam, was in de beginjaren van het Internet één van de zeven wereldwijde knooppunten (*hubs*). Nog steeds is het een van de grootste hubs in de wereld, mede doordat de belangrijkste telecommunicatiekabel tussen Europa en de VS bij Amsterdam uitkomt.

De basis van het vrij toegankelijke internet is het nauwkeurig zonder selectie doorgeven van de datapakketjes die verzonden worden. Veel landen respecteren die vrijheid niet en blokkeren (delen) van internetdiensten, sociale media of zoekmachines. Het komt er op neer dat de knooppunten in die landen selectief het verkeer van een deel van de IP-adressen niet doorlaten. Er zijn technieken als VPN om veilige verbindingen op te zetten, waarbij een apparaat voor de buitenwereld een ander IP-adres toegekend krijgt. Een thuiscomputer kan op die manier een IP-adres krijgen alsof die bij een bedrijfsnetwerk hoort. Toegang tot andere apparatuur binnen het netwerk, zoals een printer of opslagservice, kan op die manier ook geregeld worden.

5.8 Opdracht

Vraag aan je docent, TOA of ICT-beheerder of je op school kunt kijken waar de netwerkrouers en servers staan en hoe de dataopslag geregeld is.



Deel B Context: datacenters en beveiliging

Opslaan en beveiligen van belangrijke digitale data is van groot belang. Datacenters zijn bedoeld voor snel toegankelijke, maar veilig bewaarde data.

5.4 Datacenters

Voor alles wat je online met je mobiel doet zijn andere computers nodig. Ze worden 'servers' genoemd, omdat ze bedoeld zijn om jou van dienst te zijn: ze slaan jouw informatie op en sturen je de informatie die je wilt. Alle servers met opslagruimte van online diensten staan in honderden datacenters over de hele wereld.

5.9 Opdracht

Voor welke apps moet je online zijn? Welke data stuur je dan van je mobiel naar een datacenter en terug?

Rondkijken in een datacenter

Direct naast het rekencentrum van de Universiteit van Amsterdam staat een groot datacenter van Equinix, met vele verdiepingen waar servers van allerlei bedrijven opgesteld zijn. Kijk hier voor een virtuele rondleiding door dat datacenter: https://equinix-cdn.s3.amazonaws.com/virtual-tours/Amsterdam_AM4/index.htm.

Snelheid is alles in deze digitale wereld. Voor een webwinkel is het belangrijk dat bestellen en betalen zonder vertraging verloopt. In een datacenter kan een webwinkel zijn servers vlakbij die van een bank plaatsen. Uitwisselen van data tussen de servers gaat dan met zo min mogelijk vertraging. Grotere afstand betekent ook: schakelkastjes (hubs, switches) of andere servers en dus vertraging. Voor de bezoeker van de site betekent het een snelle afhandeling van opdrachten. Het is dus aantrekkelijk voor een bedrijf om haar servers ergens te plaatsen waar zoveel mogelijk connectiviteit is. In Nederland vind je deze connectiviteitsknooppunten in Amsterdam, maar ook in Rotterdam, Eindhoven en Groningen.

5.10 Vraag

Noem een paar voorbeelden van webwinkels. Denk even terug aan je laatste online bestelling. Welke stappen (denk je) dat er zitten tussen de bestelknop en jouw digitale betaling zit (het moment dat het geld van je rekening af is)?

Veiligheid is ook alles in de digitale wereld. In een datacenter zijn de volgende zaken goed geregeld: constante voeding (stroom voor de servers en schijven, noodstroom als de netstroom uitvalt), koeling (een constante temperatuur rond de 20 graden), snelle verbindingen (glasvezels en korte afstanden tot andere servers) en een streng beveiligd gebouw (mensen kunnen er niet zomaar bij). Een datacenter biedt deze veilige voorziening en bedrijven kunnen daar hun computers neerzetten,

Digitale haven

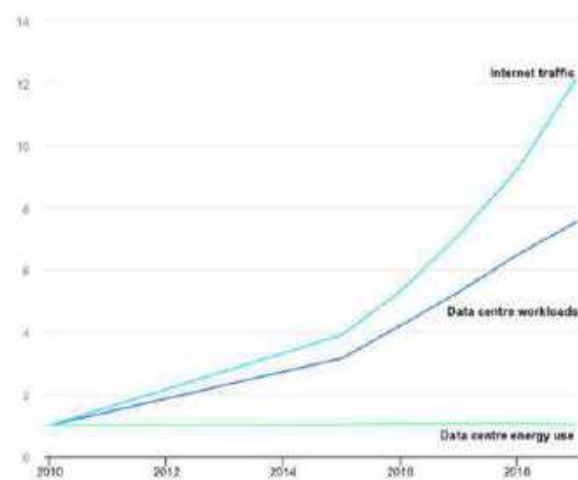
Nederland is wereldwijd een belangrijke plek voor de digitale wereld. Een centrale verbinding in het Internet is de zeekabel tussen Europa en de Verenigde Staten. Die komt in Amsterdam binnen, onderin het rekencentrum van de universiteit van Amsterdam. Daar is de Amsterdam Internet Exchange, AMS-IX gevormd, één van de allereerste en belangrijkste knooppunten van het Internet. Vanaf Nederland wordt veel data getransporteerd naar plekken elders in Europa of de wereld. Om deze reden wordt het daarom ook wel een digitale haven of digitale mainport genoemd.

door ruimte in het datacenter te huren. Datacenters die ruimte verhuren aan klanten voor het plaatsen van servers, worden ook wel *colocatie* datacenters genoemd. Weten hoe een datacenter werkt? Zie <https://www.dutchdatacenters.nl/datacenters/hoe-werkt-een-datacenter/>

Sommige techbedrijven hebben zoveel servers dat ze eigen datacenters bouwen, denk aan Google en Microsoft. Dit soort datacenters worden *hyperscale* datacenters genoemd. Wanneer een programma op een computer in een datacenter draait en de data daar ook staan, dan hoeft je op je eigen computer alleen maar wat instructies naar de computer van het datacenter te sturen. Je eigen computer hoeft dan niets op te slaan of te bewerken. Je kunt op die manier je telefoon gebruiken om allerlei plaatjes te bewerken, omdat het echte rekenwerk in het datacenter gebeurt. Dit werken in de 'cloud' gebeurt niet in de wolken, maar gewoon in een gebouw zonder ramen, ergens op een industrieterrein, een datacenter dus.

Elektriciteit

Datacenters gebruiken veel stroom. Zoveel zelfs dat ze windmolenparken in de buurt hebben om groene stroom te leveren. Tegelijkertijd zien we dat het energieverbruik van datacenters al tien jaar stabiel is, terwijl het internetverkeer en de werkbelasting voor de datacenters enorm gestegen zijn. Dit is voor een belangrijk deel te danken aan het samenvoegen van kleine inefficiënte serverruimtes die bedrijven zelf ingericht hadden, tot professionele, grote en efficiëntere datacenters die we nu kennen. Ook het gebruik van meer Cloud-applicaties heeft bijgedragen aan het gelijk blijven van het stroomverbruik. Het restproduct van stroomverbruik in een datacenter is restwarmte. Naast besparing op stroomverbruik (bijvoorbeeld door harde schijven te vervangen door solid state drives) wordt ook gekeken of de restwarmte nuttig gebruikt kan worden.



Figuur 5.6 Energieverbruik van datacenters, in vergelijking met dataverwerking en internetverkeer.

5.11 Opdracht

Duurzaam maken van digitale technologie heeft veel te maken met het verbruik (en de gebruikte bronnen) van energie en grondstoffen en kringlopen.

- Welke manieren kun jij bedenken om digitale technologie zelf duurzamer te maken? Je kunt kijken naar het verminderen van negatieve effecten, zoals hoog stroomverbruik, digitaal afval en opraken van grondstoffen. Hierbij kun je uitgaan van een voorbeeld, zoals een datacenter, je (school)computer, telefoon, minicomputers als Arduino en allerlei printers of digitaal gereedschap. Ook kun je je richten op de samenhang in het hele systeem.
- Op welke manier draagt het gebruik van digitale technologie zelf bij aan duurzame ontwikkeling? Denk bijvoorbeeld aan online samenwerken, slim plannen van routes of apparaten die zichzelf kunnen uitschakelen als ze niet gebruikt worden.

5.5 Datacenters en 'Cloud'

De standaardopslag in datacenters is vaak de 3,5 inch harde schijf, waarvan er tienduizenden samengepakt zitten in rekken, die samen aangestuurd worden door server-computers. De processor kan wat meer verwerken dan in je computer thuis, maar de opbouw is hetzelfde. In plaats van één harde schijf heeft een server er tientallen. Ook thuis wordt data vaak apart van de computer opgeslagen op een zogenaamde NAS (Network Attached Storage), een kastje met een minicomputer, harde schijven en een netwerkaansluiting. Met wat handige software kan zo'n kastje ingericht worden om films, documenten en foto's op te slaan. Misschien wel het belangrijkste: door een serie harde schijven samen te voegen kan een array worden gemaakt, waarmee verlies van gegevens voorkomen wordt (in geval een harde schijf kapot gaat). Een systeem als RAID is een handige manier om met minder opslagruimte toch van alle data een kopie te hebben.

5.12 Opdracht

Zoek uit wat RAID is en hoe het werkt. Welke vormen van gegevensbescherming kom je nog meer tegen?

Data redundancy, ofwel voorkomen van dataverlies als er iets stuk gaat, is belangrijk voor veiligheid van data. Minstens zo belangrijk is het voorkomen van diefstal, brand, stroomuitval en andere zaken waardoor data onbereikbaar wordt. Veel bedrijven zijn volledig afhankelijk van goed werkende servers die constant beschikbaar zijn. Een hokje met computers in een bedrijfspand is niet handig en het risico van uitval of dataverlies door calamiteiten is te groot. Vandaar dat bedrijven hun computers en schijven het liefst in een veilige en optimale omgeving zetten: datacenters.

5.6 Beveiliging

Het sturen van kleine datapakketjes van en naar aparte computers is probleemloos als iedere computer op het internet netjes doet wat de bedoeling is. Helaas is de wereld niet zo eerlijk. Pakketjes worden onderschept en gelezen, je krijgt pakketjes aangeboden die schadelijk zijn en ga zo maar door. Via internet kan informatie uit een computer gelezen en gewijzigd worden.

We zijn zó afhankelijk van goed werkende computers, die onderling veilig data kunnen uitwisselen, dat beveiliging misschien wel het meest belangrijke onderwerp in de IT-wereld is. Hoe zorg je voor veilige computers?

5.13 Vraag

Je gebruikt op je computer en telefoon verschillende vormen van beveiliging. Welke? Noem er tenminste drie.

Poorten

Zoals een gebouw meerdere deuren heeft, geldt dat ook voor een router. Routers en computers beschikken over poorten. Daardoor kunnen datapakketjes intern precies afgeleverd worden. Pakketjes voor de webserver gaan via een standaard poort (bijvoorbeeld 72), dat hoeft niet in het pakketje beschreven te zijn. Maar als een server niet alle pakketjes wil hebben of heel nauwkeurig alleen van één computer de pakketjes wil doorgeven, wordt gewerkt met lijstjes afzenders en poortnummers. Iets als 19.02.202.10:5002 is een IP adres en een poortnummer. Je moet dus vertellen waar je pakketje terecht moet komen. Dat is een van de vormen om internetverkeer te beveiligen. Je kent zelf wel de gebruikersnaam-wachtwoord toegang. Een poort maakt het iets veiliger. Als je ook nog tevoren hebt afgesproken welke computers contact met elkaar mogen hebben (en de webserver kent jouw computer IP), wordt het nog wat meer beveiligd.

Wanneer je pakketjes, voordat ze verstuurd worden, door elkaar schudt (*versleutelen, encryptie*) en bij de ontvanger weer sorteert (met dezelfde sleutel), kun je informatie nog een stapje veiliger versturen. Het is belangrijk dat verzender en ontvanger hebben afgesproken welke sleutel ze gebruiken (maar uiteraard de sleutel niet in het pakketje verstoppen).

Voor de beveiliging wordt veel kennis van wiskunde gebruikt. Priemgetallen, reeksen willekeurige getallen, wachtwoordregels of datacorrectie zijn veelgebruikte methodes. Bijvoorbeeld: een controlegetal de som van alle getallen in de reeks moet even zijn. Als dat niet zo is, wordt de controlebit 1, anders 0. Ook hier kun je je met vervolgmodes uitgebreid in verdiepen (zie hoofdstuk 8).

Hacken en cyberaanvallen

Binnendringen van een computer of webserver betekent dat je door de beveiliging moet zien te komen. Daarvoor moet je een heleboel informatie hebben. Een gebruikersnaam, het wachtwoord, het juiste veiligheidscertificaat, juiste IP adres, encryptiesleutel: je kunt het heel moeilijk maken om binnen te komen. Een moeilijk te raden wachtwoord helpt. Bedenk dat hackers niet altijd zelf wachtwoorden gaan intypen. Ze hebben daarvoor een programma dat gebruik maakt van allerlei lijsten. Uit een woordenboek geplukt, willekeurig samengesteld of gejat uit een slecht beveiligde database of 'eerlijk' gekocht op een criminele marktplaats (dark web).

Maar: snelle computers kunnen ook zelf wachtwoorden samenstellen en uitproberen. Wachtwoorden van meerdere karakters kunnen in heel korte tijd gemaakt en uitgeprobeerd worden. Hoe langer het wachtwoord, hoe langer de computer er over doet en hoe lastiger het wordt. Een slecht beveiligd achterdeurtje kan zo de toegang tot belangrijke informatie zijn. Apparaten die je met internet verbindt (slimme thermostaat, webcam, koelkast) zijn heel vaak voorzien van een simpel te raden of standaard wachtwoord. Hackers zoeken vaak naar dit soort apparaten om je netwerk binnen te komen. Immers: het apparaat heeft een IP-adres en is daarmee te vinden voor computers. Via zo'n apparaat kun je andere apparaten in het netwerk bereiken: je bent binnen via de achterdeur.

5.14 Opdracht: Sterke wachtwoorden

Wachtwoorden bestaan meestal uit de 256 verschillende ASCII tekens (letters, cijfers, leestekens).

- a. Als een wachtwoord 5 tekens lang is, hoeveel verschillende wachtwoorden kun je dan vormen? Bereken dit ook voor een lengte van 10 tekens.
- b. Stel: een vlotte hack-computer kan 1 wachtwoord per seconde maken en uitproberen. Hoe lang doet deze dan over het hacken van jouw mailbox?

Via de voordeur binnenkomen gebeurt ook vaker dan je zou willen. Een bericht met een bijlage of een link die je opent, kan een pakketje afleveren met gevaarlijke software: een virus of gijzelsoftware. De software kan in je telefoon of computer informatie zoeken en doorsturen (bijvoorbeeld inlog van je bankrekening) of alle bestanden blokkeren (en tegen betaling weer vrijgeven, *ransomware*).

5.15 Opdracht

Zoek op internet tenminste twee voorbeelden van hacken, falende beveiliging of dreiging van cyberterreur. Waar gaat het mis in die situatie, wanneer die beveiliging niet goed is?

Routers en beveiliging

Routers en hubs zijn ook kleine computers met chips en software en met algoritmen die door mensen geprogrammeerd zijn. Het netjes doorsturen van pakketjes, zonder ongevraagd kopiëren elders heen te sturen, is iets wat je van een router verwacht. Maar het is niet ondenkbaar dat de chips waar de computer in de router mee gemaakt is, al van zichzelf schakelingen hebben die gemaakt zijn om pakketjes te kopiëren en ook naar een ander adres te sturen. De schakelingen in de chips zijn zo ingewikkeld, dat het niet eens hoeft op te vallen. Intussen doet de router andere dingen dan de bouwer van het apparaat bedoelt, alleen al doordat de chips die geplaatst zijn anders werken of méér doen, dan in de specificatie staat (wat ze volgens de beschrijving zouden moeten doen).

Je kunt dit zien als valsspelende postbodes die brieven onderweg kopiëren of verwisselen en pas daarna bezorgen. We moeten vertrouwen op wat de chipfabrikant levert. Als de apparaten ook niet door jezelf gemaakt worden, is het helemaal lastig om exact te controleren wat er allemaal in zit.

In de tech-industrie is het maken van chips en apparatuur op een paar plekken in de wereld gecentraliseerd. Het allergrootste deel van de spullen wordt in Azië in elkaar gezet, ondanks dat het om merken uit Amerika of Japan gaat. Een monopolie op de fabricage én de grondstoffen daarvoor maakt ons allemaal uiterst afhankelijk en daarmee ook de mogelijke spionage die ermee gepaard gaan.

Het nieuwe 5G netwerk vraagt om heel veel nieuwe apparatuur: 5G zenders en snelle netwerkkapparatuur om die enorme datastromen (in datapakketjes) goed te kunnen vervoeren. Allerlei bedrijven hebben daar apparatuur voor ontwikkeld. Maar de bedrijven die eerder deze apparatuur leverden, beschikken niet meer over de snelste technologie en kunnen de slag verliezen van nieuwe aanbieders uit Azië. Helaas is daarvan niet duidelijk of er in de apparatuur ook spionagetechnologie is ingebouwd. Vandaar dat bepaalde aanbieders geweerd worden uit de kern van onze data-infrastructuur.

5.16 Opdracht

Probeer te achterhalen welke verdenkingen tegen aanbieders van 5G-technologie er zijn en welke rol dat speelt in de besluitvorming in Nederland.



Deel C Zelf doen

Zelf doen en maken is niet alleen knutselen en testen, maar ook het opvragen, bewerken en gebruiken van data. Je kunt hier verschillende kanten mee op. Kies één of meer opdrachten als afsluiting van dit hoofdstuk.

5.7 Keuzeopdrachten

a. Kleine arduino's maken samen big data

Fijnstof, partikels in de lucht van 1 tot 500 nm, kunnen met een fijnstofsensoren gemeten worden. Je moet denken aan roetdeeltjes uit auto's of vliegtuigen of deeltjes uit de rook van een fabriek.

Hoewel stuifmeel (pollen) er niet bij hoort, is dat ook iets dat in de lucht zweeft en door ons ingeademd wordt. De vervolgmodule Fijnstof gaat hier in detail op in.

De Arduino die hiervoor gebruikt wordt, meet via de fijnstofsensoren en geeft data draadloos door via het Internet of Things op het 2G netwerk. De hoeveelheid data van één sensor gaat nog wel, maar honderden over het hele land leveren een aardige massa data op. Om daar iets aan te hebben, moet je de data voorzien van gegevens over de locatie, tijdstip, structuur enzovoort en netjes ordenen in een database. Als die metadata in orde is, kun je de gegevens uit de database lezen en bijvoorbeeld combineren met gegevens van een digitale kaart. Je kunt dan een nieuw plaatje berekenen waarop de fijnstofwaarde in ons land op een bepaald moment te zien zijn. Als dat allemaal via een website te raadplegen is, moet de webserver deze data van verschillende bronnen ophalen en combineren. Kijk op www.rivm.nl/fijnstofmeting-en-globe.nl voor meer informatie.

b. Je eigen Arduino meetproject.

Arduino is prima geschikt om metingen op te slaan en door te sturen. De thermometer die je in H4 gemaakt hebt, kun je voorzien van opslag (kaartje) of draadloze verbinding (wifi, bluetooth, LORA-WAN). Zoek op sites van Arduino-spullen wat je hiervoor nodig hebt. Wellicht zijn die spullen ook op school beschikbaar. Vraag gerust aan je docent of TOA!

Voorzie je Arduino-thermometer van een manier om de gemeten data te kunnen opslaan of versturen. Zorg dat de Arduino stroom krijgt uit een batterij.

- Opslaan: voorzie je Arduino van een klok en een opslagmedium, zoals SD-kaartje
- Versturen: voorzie je Arduino van een klok en wifi of bluetooth

Zoek de juiste drivers en herschrijf de sketch voor je uitgebreide Arduino thermometer.

Gebruik je device om gedurende een aantal uren temperatuur te meten. Temperatuur in een klaslokaal of je eigen lichaamstemperatuur bij het sporten of slapen. Bij de laatste twee situaties moet je ook nadenken hoe je de sensor op een goede manier met je lichaam kunt verbinden. De opdracht met Q-strip (H7) geeft ook een opstap naar het zelf maken van een vochtsensor. Gebruik die kennis om een portable device te maken waarmee je tijdens het sporten het zweten kunt meten.

<https://www.arduino.cc/en/Guide/ArduinoUnoWiFiRev2>

<https://www.arduino.cc/en/Reference/WiFiNINA>

<https://store.arduino.cc/explore-iot-kit>

c. Satellieten en informatie over de aarde (GIS)

In bijlage 2 in de moduledatabase staat een opdracht over de digitale wereld achter aardobservatiesatellieten. Beelden die Landsat dagelijks maakt zijn via internet te bekijken, gecombineerd met geoinformatie (GIS). Je kunt zo de actuele situatie en verandering in gebieden op aarde onderzoeken. Ook Sentinel levert interessante informatie op. Je maakt kennis met databases en technieken om data uit allerlei bronnen overzichtelijk te presenteren.

d. Verder te verkennen / te onderzoeken toepassingen

Er is veel aan de orde gekomen in dit hoofdstuk, maar niet zo uitgebreid. Er valt nog veel te ontdekken. Verdiep je in één van onderstaande thema's. Met de kennis van Arduino of andere platforms kun je zelf ook aan de slag om met dit thema iets te bouwen.

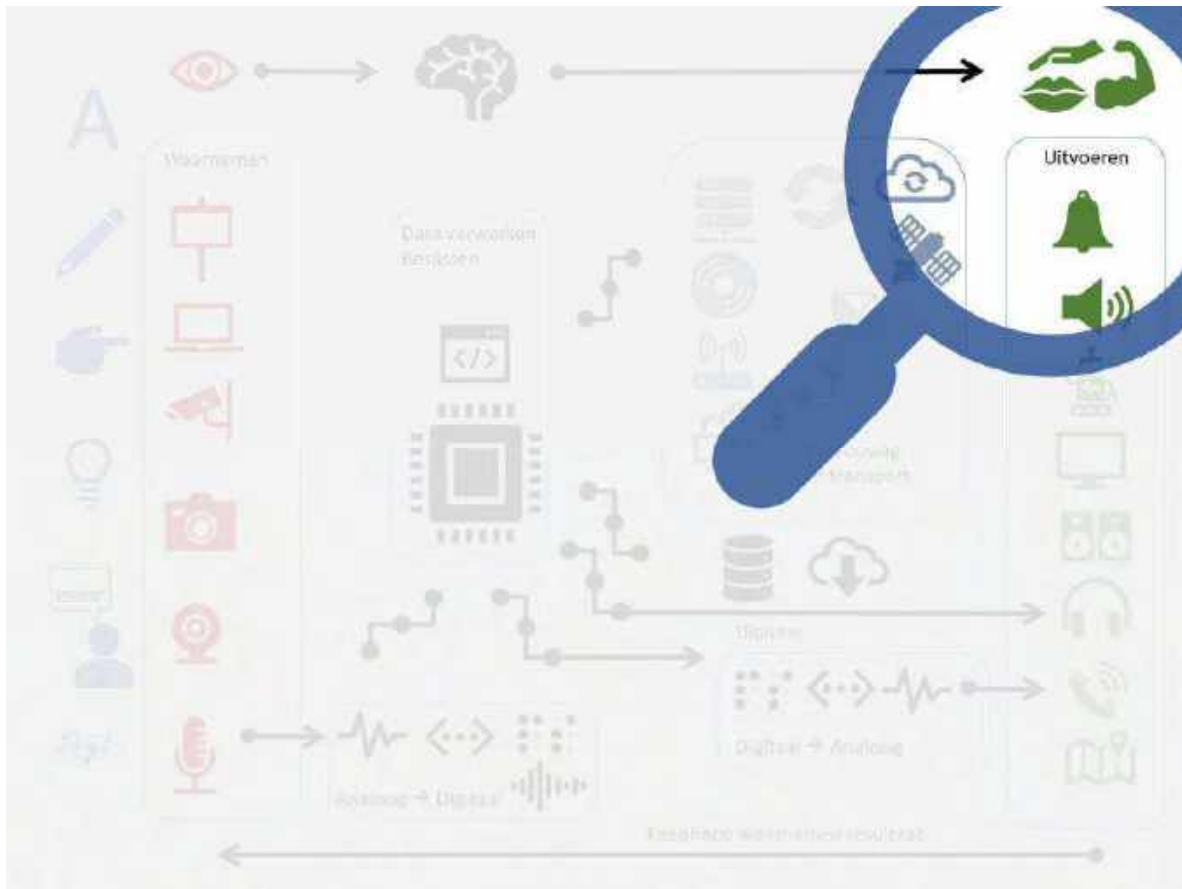


Figuur 5.7 Landsat satelliet die dagelijks honderden foto's van het aardoppervlak maakt

Maak daarvan een presentatie, informatieve poster of een schriftelijk verslag.

- 5G communicatie
- Glasvezel en optische communicatie
- Een digitaal geheugen
- Datacenter: hoe is dat ingericht
- Telefonie, videobellen
- Streaming media
- LORA netwerken

6 Uitvoer: de actuator



Leerdoelen

Kennis

- 6.1 Je kunt uitleggen dat digitale systemen invloed kunnen hebben op hun omgeving door middel van actuatoren
- 6.2 Je kunt uitleggen dat actuatoren zoals lampjes, speakers, motoren, RF zenders, servomotoren in veel verschillende digitale systemen voorkomen en dat je die (relatief) eenvoudig zelf kunt bouwen

Vaardigheden

- 6.3 Je hebt ervaren dat je in je programmeercode in je digitale systeem soms bepaalde bibliotheken moet gebruiken om actuatoren te kunnen gebruiken
- 6.4 Je hebt ervaren dat je voor het aansturen en monitoren van actuatoren online veel informatie kunt vinden op websites als stackoverflow en reddit.
- 6.5 Je hebt kennism gemaakt met het aansturen van actuatoren en dat dit gevolgen heeft voor het stroomverbruik van je digitale systeem en dat ontwerpkeuzes dus invloed hebben op de duurzaamheid van je systeem
- 6.6 Je hebt ontdekt hoe goed jouw actuatoren functioneren en hun omgeving beïnvloeden en kunt beoordelen of jouw digitale systeem voldoet aan de ontwerpseisen die je hebt gesteld.



Deel A Uitleg en naslag

Actuatoren: manieren waarop een digitaal systeem invloed kan uitoefenen op de buitenwereld

6.1 Invloed op de omgeving uitoefenen

Aan digitale apparaten hebben we weinig als ze niets in buitenwereld kunnen veranderen. Daar horen jij en ik ook bij. Daarom heeft een digitaal apparaat onderdelen die kunnen bewegen, geluid maken of licht geven: de actuatoren. Soms zijn dat onderdeeljes die al verwerkt zijn op een chip of printplaatje, zoals een lichtgevende diode (LED) die aangaat als de chip stroom krijgt, zodat we daaraan kunnen zien of het apparaat ‘aan’ staat of dat er data ontvangen of weggestuurd wordt.

Een digitaal systeem zonder actuatoren kan uiteindelijk niet communiceren of dingen doen. Pixels op een scherm aan- of uitschakelen is een manier van communiceren. Een triltoon van je mobiel is eigenlijk een klein motortje dat even aangaat. Zo zijn er allerlei dingen bedacht om licht, geluid of beweging te maken. Uiteindelijk gebeurt dat bijna altijd door elektrische spanning aan of uit te schakelen. Wat we als mensen met onze handen doen, laten we nu door machines uitvoeren.

Digitale actuatoren zijn verwerkt in de machine en worden door de processor(en) aangestuurd. Ze gebruiken informatie uit databestanden en input uit sensoren. Daarmee kunnen ze hun klus doen. Naast de actuatoren heeft een systeem ook veel sensoren die waarnemen wat het systeem zelf doet en veranderingen in de fysische grootheden waarnemen in de omgeving van het systeem. Hierdoor kan het systeem terugkoppeling ontvangen en zichzelf bijsturen. Hoe ingewikkelder de omgeving en hoe minder voorspelbaar, hoe belangrijker het is om die terugkoppeling te gebruiken.

Maar bewegingen en verplaatsingen, zoals zelfrijdende auto's of robotarmen, vragen veel meer controle en waarneming, omdat de buitenwereld een stuk minder voorspelbaar is én omdat het apparaat zelf beweegt. Terugkoppeling is het gebruik van informatie over het effect om de handeling te kunnen bijsturen. Zelfsturende (autonome) systemen zijn heel ingewikkeld, omdat ze vol sensoren zitten om de terugkoppeling naar de actuatoren te kunnen uitvoeren.

6.1 Opdracht: Terugkoppeling

Teken een eenvoudig schema voor een zelfrijdende auto of een autonome robot. Benoem enkele sensoren, processoren en actuatoren en geef aan op welke manier terugkoppeling (feedback) in dit systeem noodzakelijk is

Voorbeelden van actuatoren die in digitale systemen bruikbaar zijn

Fysieke verandering	Actuator	Toepassing
Draaibeweging	Stappenmotor	
Lengteverplaatsing	Linaire motor	
Licht geven	Diode	
Beeldscherm	LCD, OLED: beeldpunt laat licht door of produceert licht	
Klep bedienen	Lineaire motor of Piezo (zet uit als er elektrische spanning op komt)	
Geluid laten horen	Luidspreker: spoel en magneet Piezo-element	

Licht geven of selectief doorlaten

Met actuatoren die licht geven (LED) of blokkeren (vloeibare kristallen) als beeldelement (pixel) of kleurstof die zich verplaatst bouwen we de enorme variatie aan beeldschermen die we nu kennen. Als digitale data gebruikt wordt om beelden te maken wordt iedere pixel apart aangestuurd. Kleuren worden gemaakt door aparte pixels die rood, groen of blauw licht uitzenden. Er is digitale data nodig voor kleur en helderheid van de pixel én om te vertellen welke pixel die waarde moet krijgen (adres).

6.2 Opdracht

Kijk eens met een sterk vergrootglas naar het beeldscherm van je telefoon of computer. Hoe ziet dat er uit op een plek die 'wit' is? Wat zie je bij een plek die 'geel' is?

Een beeldscherm kan een matrix zijn van LED's (die zelf licht uitzenden), zoals in een OLED (Organische LED) scherm. Vloeibare kristallen (die licht doorlaten als ze spanning krijgen) vormen samen met een witte achtergrondverlichting een LCD (Liquid Cristal Display). TFT (Dunne Film Transistor) is een variant van LCD die transistoren heeft om de individuele pixels te kunnen schakelen. De verfijnde mogelijkheden om deze beeldschermen dun en flexibel te maken zorgt voor de ongekend grote beeldschermen met miljoenen pixels en kleuren en hoge snelheid.



*Figuur 6.1
Uitvoeren: de
actuator*

Een digitale projector (beamer) heeft kleine, doorschijnende LCD beeldschermpjes voor de drie primaire kleuren rood, groen en blauw, met een heldere lamp erachter. Dit is steeds vaker een heldere LED, vandaar LED beamer. Ook de bioscoop gebruikt tegenwoordig digitale projectoren.

6.3 Opdracht

Zoek eens uit hoe een digitale projector in elkaar zit.

Digitale data wordt gebruikt om LCD pixels te schakelen. Voor zowel LCD als LED pixels wordt de helderheid bepaald door de tijdsduur die de pixel doorschijnend is of aan staat. Dit gebeurt door Pulse Width Modulation. Als een lichtbron vaker dan 24 keer per seconde aan-uit gaat nemen onze ogen geen flikkering meer waar. Vandaar dat beeldschermen een verversingsfrequentie van 50 Hz of meer hebben. Toch waarderen onze ogen die flikkering niet altijd, vooral bij het lezen van beeldschermen. Onze ogen bewegen snel en trillen zelf ook. Het flikkeren van het beeldscherm kan ervoor zorgen dat de ogen precies in beweging zijn als het scherm wordt verversd en daardoor even het focuspunt kwijt zijn. Dat kan verklaren waardoor lezen van beeldschermen minder prettig is. Bovendien stralen beeldschermen (veel) licht uit, wat ook vermoeit.

Er zijn schermen die niet constant verversd worden, namelijk E-ink of elektronisch papier. De zwarte korrels in de inkt verplaatsen zich als er spanning aan één kant wordt gezet. Als de spanning weg valt blijft de inkt op z'n plek, tot de spanning aan de andere kant van de pixel staat. Met de grootte en tijdsduur van de spanning kan de zwarteheid van de pixel worden bepaald. Hiermee ontstaat een stabiele matrix van grijswaarden. Er is geen spanning nodig om het beeld te bewaren. Zo'n scherm verbruikt nauwelijks stroom, maar is ook heel traag en dus niet zo geschikt voor een computer, maar prima voor een E-reader. Met omgevingslicht is het scherm te lezen als papier, wat onze ogen zeer waarderen. Met een beetje achtergrondlicht zijn zulke schermen ook bruikbaar in het donker.

De digitale data moet vertellen of een pixel aan staat, in welke kleur en welke helderheid. Om dit mogelijk te maken moet ook bekend zijn waar de pixel zit. Goede beeldschermen hebben veel pixels, veel kleuren, snel schakelende pixels en een groot verschil tussen donker en licht (contrast). Daarvoor is veel data nodig. Vooral bewegend beeld vraagt veel bandbreedte, om schermen te

kunnen aansturen. Door de snellere processoren, glasvezel en betere datacompressie lukt het beeld van hoge kwaliteit te transporteren én te tonen.

Verplaatsen: motoren



Figuur 6.2 Digitaal - Analooq conversie

Een elektromotor zet stroom om in een draaibeweging. Het is prettig als je die draaiing onder controle kunt houden. Daar zijn grofweg twee manieren voor. De ene manier is dat er aan de motor een sensor komt te zitten, die vertelt in welke stand de motor staat of hoeveel omwentelingen de motor heeft gedaan.

De andere manier is het gebruik van een stappenmotor. Een stroompuls zorgt dat de motor één stapje verplaatst, bijvoorbeeld 2 graden (van de 360). Met het aantal stroompulsjes is te bepalen hoever de motor draait. Het digitale signaal wordt zo rechtstreeks vertaald in een kleine verplaatsing. Met tandwielletjes en een aandrijfriem of een schroefdraad kan een draaiende beweging ook omgezet worden in een lineaire beweging. Je vindt dit bijvoorbeeld in printers (zowel 3D als inktjet). Ook de motortjes in een harde schijf zijn actuatoren die aangestuurd moet worden.

Geluid maken

Een korte spanning kan ook weer gebruikt worden om een elektromagneet te activeren. Als je dit snel doet, kan daarmee een snelle trilling (en dus geluid) ontstaan. Een trage trilling kan gebruikt worden als trilfunctie in je mobiel. Voor kwalitatief goed geluid moet het digitale signaal netjes worden omgerekend naar een continu variërende spanning met veel meer vermogen. Daarvoor dient de digitaal-analoog convertor in een (digitale) audioversterker of de aansturing van een lijnuitgang of hoofdtelefoon-uitgang.

Electromagnetische straling maken

De digitale pulsen kunnen ook gebruikt worden in een radiozender. Een spoel zendt radiogolven uit als er elektrische pulsen binnenkomen. De radiogolven van precies vastgelegde golflengte kunnen als Wifi of Bluetooth signaal opgevangen worden door antennes (ook sensoren).

Printen

Inkt of poeder (in de basiskleuren CMYK: cyaan (blauwig), magenta (rozerood), yellow (geel) en black (zwart)) vormen de basis van de vierkleurendruk, inkjetprinters en laserprinters. Motoren verplaatsen het papier en de inktkop. Laserbundels zorgen voor een elektrische lading op een metalen rol, waardoor kleurstofpoeder (toner) vastgehouden en daarna op papier gedrukt kan worden. Na verhitten zit de toner vast op het papier. Inktjetprinters werken met vloeibare inkt die uit microdunne kanaaltjes gespoten wordt. Bij een inktjetprinter bepaalt Piezo-materiaal (dat uitzet als er spanning op staat) of de inkt uit het kanaaltje kan. Een digitaal bestand vertelt wanneer, waar en hoeveel inktdruppeltjes van elke kleur boven het papier afgeschoten moet worden.

6.4 Opdracht

Thermisch papier wordt zwart als het warm wordt en is daardoor heel handig om simpel te kunnen afdrukken. Veel kassa's hebben een thermische printer. Op welke manier zou zo'n printer leesbare tekst kunnen afdrukken?

6.5 Opdracht

Onderzoek welke data een inkjet- of laserprinter nodig heeft om een afdruk te kunnen maken en hoe een printer dat voor elkaar krijgt. Welke actuatoren zitten er in de printer en hoe worden die aangestuurd? Heeft een printer ook sensoren, processoren, dataopslag en transport? Gebruik het analyseschema (leeg schema achterin de lesmodule of verkrijgbaar als los werkblad) om je bevindingen overzichtelijk te maken. Welke verschillen kun je noemen tussen de inkjetprinter, laserprinter en thermische printer?

6.6 Opdracht: Andere digitale maakapparaten

Een 3D printer, plotter, CNC frees, digitale snijmachine of een lasersnijder werken technisch niet heel veel anders dan een inkjetprinter. Het belangrijkste verschil is het werktuig: een spuitmond voor vloeibaar materiaal (3D printer), een pen, een ronddraaiende beitel, een mesje of een krachtige laserbundel die papier, metaal, hout of kunststof kan bewerken. Vind een aantal voorbeelden op het internet. Voor welke toepassingen wordt het apparaat gebruikt? Maak hiervan een overzichtelijke tabel.

Kies één van deze apparaten uit en zoek verder uit hoe deze (technisch) werkt. Onderzoek ook op welke manier zulke apparaten aangestuurd worden (hoe de data er uit ziet als ze hun werk doen). Kijk of er één of meer van deze apparaten op school aanwezig zijn en vraag om uitleg daarover. Een *Fablab* of *Makerspace* heeft zulk digitaal maakgereedschap zeker in huis.

Andere computers als actuator: datasystemen

Wanneer digitale apparaten data aan elkaar doorgeven, kunnen uitgebreide systemen worden gebouwd waarin gegevens van talloze sensoren gebruikt worden om data te verzamelen. Deze data worden gebruikt om op allerlei plekken informatie te laten zien. Een apparaat kan dan één stukje van het totale proces uitvoeren. Het ene apparaat doet bijvoorbeeld de omzetting van een temperatuur naar een digitaal signaal. Een ander apparaat verzamelt de gegevens van de meetapparatuur en kan de gegevens opslaan in een database. Een computer heeft dan zelf geen beeldscherm of printer nodig en kan toch invloed hebben op de buitenwereld.

Computers kunnen de gegevens uit de database lezen, en via programma's (algoritmen) nuttige informatie uit de database berekenen. Een klimaatwetenschapper kan dit gebruiken om temperatuurstijging uit lange termijn weergegevens te berekenen. De uitvoer van de gegevens gebeurt dan pas als de data in een lijst of als grafiek op een beeldscherm getoond worden. De bits in de data zijn dan al talloze keren tussen processor en opslag uitgewisseld.

6.7 Opdracht: verdieping

Op welke manier worden digitale motoren aangestuurd? 3D-Printbestanden, snijfiles, robotprogramma's, stuurprogramma's voor printer. Basisstructuur van een file, wat voor soort codering?

Voor de makers

Ook voor actuatoren moet code gebruikt worden om ze goed aan te sturen. Dit soort informatie wordt gedeeld via online platformen. Als je Arduino-actuatoren gebruikt, zijn ze te vinden via [Arduino.cc](https://arduino.cc) of Tinkercad of het platform dat bij je device hoort (micro:bit, Raspberry Pi of M-bot).

De professional kijkt op <https://stackoverflow.com> of <https://www.reddit.com>. Als je digitale ontwerpen (zoals 3D tekeningen) maakt, kun je die bijvoorbeeld op github.com delen. Er is véél meer dan we hier kunnen noemen. Via de leerlingensite is ook e.e.a. te vinden.



Deel B Context

waar wordt dit gebruikt (sporthorloge, satelliet, beeldvorming, robots, datacenter, misschien meer bij verwerken dan bij opslag)

6.2 Mechatronica in bedrijf

Mechatronica is het gebied waar mechanica (werktuigen) en electronica (digitale technologie) elkaar ontmoeten. Slimme machines, robots, scanners: overal worden actuatoren gebruikt. Mechatronica is een uitgebreid terrein, want een digitaal systeem zonder actuatoren is uiteindelijk onbruikbaar. In Nederland wordt veel onderzocht en gemaakt voor of met de digitale technologie.

Opdracht 6.8: Kennismaken met bedrijven

Over diverse bedrijven zijn lesbrieven gemaakt om je een beeld te geven wat ze (met digitale technologie) doen. Deze kun je vinden op: <http://www.brainport.nl/lesmateriaal>.

Kies één van de hieronder genoemde vier bedrijven en werk een gedeelte van de bijbehorende lesbrief uit (afhankelijk van je beginkennis). Je kunt uiteraard méér of andere kiezen. Overleg de keuzes met je docent.

1. ASML: hun chipmachines zijn de werkpaarden waarmee fabrikanten alle chips maken die in alle digitale apparatuur wordt toegepast. Het EUV licht wordt gemaakt met tindruppeltjes en een zware laser. Hoe gaat dat in zijn werk?
2. Vanderlande: wereldwijd worden in luchthavens en sorteercentra de lopende banden van Vanderlande gebruikt. Hoe in die systemen de digitale technologie een rol speelt ontdek je in de lesbrief.
3. DAF: DAF werkt aan zelfrijdende vrachtwagens. Kunstmatige intelligentie moet helpen om de vrachtwagen de juiste koers te laten rijden. Maar hoe gaat dat in zijn werk?
4. Philips: Bij Philips wordt medische apparatuur gemaakt, waaronder MRI scanners. Het herkennen van beelden uit deze scanners is een vak apart waar kunstmatige intelligentie welkom is. Hoe gaat dat in zijn werk?

6.9 Opdracht: Actuatoren in context?

Er zijn talloze andere, aansprekende voorbeelden van de rol van actuatoren bij digitale technologie. Ze zijn de moeite waard om verder uit te zoeken. Kies één van onderstaande onderwerpen om je verder in te verdiepen. Maak over het thema een stuk van maximaal twee pagina's (inclusief illustraties), dat je zou kunnen toevoegen aan deze module.

- Robotica (Boston Dynamics, zelfrijdende auto, drone, ...)
- Robotarm (voor operaties, inbrengen hersenimplantaat, lassen of montage)
- Het knuffelrobotje voor dementerenden
- Beeldschermen die zich aanpassen aan de lichtomstandigheden en de gebruiker
- Robotbeestje om gedrag op video vast te leggen (Pinguinrobot met camera) Planet-earth.
- Rat met actuator in brein – aardbevingsgebieden doorzoeken
- Microscopen en 3D beeldvorming



Deel C Zelf doen

Maakopdrachten op verschillende niveau's. Met een serie maakopdrachten kan telkens een nieuw aspect van digitaal maken ontdekt worden. Zoals: ledje laten branden (1. sensoren), code aanpassen (2. data verwerken), draadloos aansturen (3. Datatransport), actuator laten werken (4. actuatoren). In hoofdstuk 7 zit een verdiepende opdracht met deze kennis.

6.3 Maakopdracht met Micro:bit, Arduino of Raspberry Pi.

6.10 Opdracht

In bijlage 3 in de moduledatabase wordt de NeoPixel beschreven. Ga daarmee aan de slag, en experimenteer met de mogelijkheden die zulke LEDs bieden.



Programmeer vervolgens de Arduino (of een ander device), zodat een serie NeoPixels gebruikt kunnen worden als mooie richtingaanwijzer voor een moderne auto (ledjes die van links naar rechts oranje oplichten). De richtingaanwijzers aan de achterzijde zijn daarna uit, aan de voorzijde geven ze na gebruik continu wit licht geven als dagrijverlichting. Maak voor beide situaties een programma.

6.11 Keuzeopdracht

Er zijn talloze actuatoren en meer dan genoeg interessante systemen om te onderzoeken of te bouwen. Hieronder een aantal voorbeelden. Op de leerlingensite staan wat tips om mee te beginnen. Vraag je docent / TOA welke materialen er op school beschikbaar zijn:

Beeldscherm	Zelfrijdende auto
CNC machine	Robotarm
Lasersnijder	Kleppen
Plotter / snijplotter	Schakelaars (bijvoorbeeld een schakelaar om voor 220 V apparaten met Arduino te bedienen)
3D printer	Beweging en licht
Voedselprinter	Temperatuur verandering (verwarmer)
Inkjet	Menger
Poederstraal printer	Drukvat, compressor (aansturen zware machines)
Thermische printer	Stembediening voor je lampen, gordijnen, koffiemachine.
Holografische projector	
Beamer	
Robotproject Leaphy	
Karretje	
Motoren	

7 Eindopdracht: Digitaal device ontwerpen

Leerdoelen

- 7.1 Je leert de ontwerpcyclus beheersen en toepassen om een werkend prototype te kunnen bouwen dat een probleem oplost.*
- 7.2 Je krijgt inzicht hoe innovaties in de tijd veranderen*
- 7.3 Je maakt kennis met opdrachtgevers en realistische opdrachten uit het bedrijfsleven*

Projecten

7.1 Digitale technologie in huis: weerstation voor het binnenklimaat.

In huis wil je een veilig en gezond binnenklimaat hebben. Fijne temperatuur, geen geurtjes, tocht of vocht, en de waarden voor CO₂ en CO binnen de normen. Deze parameters zijn goed meetbaar met digitale sensoren. Met je kennis van digitale microcontrollers kun je een gecombineerd device bouwen en programmeren, dat deze waarden voor je in de gaten houdt. Maak gebruik van de standaardsensoren voor vocht, licht en temperatuur. Fijnstof (rook), CO en CO₂ vragen extra sensoren. Zoek op internet wat er verkrijgbaar is of vraag je docent/TOA naar de beschikbaarheid op school.

Opdracht:

Maak met digitale elektronica een weerstation voor in huis.

Meet een combinatie van licht, temperatuur, vocht, fijnstof en tocht (tenminste drie van deze waarden). Programmeer het apparaat zo dat de afwijking ten opzichte van de comfortwaarde (optimale waarde) wordt bepaald.

- a. Eerste stap: als er iets niet deugt, kan het device via LEDs signalen afgeven. Groen betekent: alle meetwaarden vallen binnen de grenzen (beneden-bovengrens). Geel betekent: één of meer waarden wijken gering af. Rood betekent: één of meer waarden wijken sterk af. Op een display moet blijken welke waarde afwijkt / afwijken en hoe sterk.
- b. Een stapje verder: het kan zijn dat het apparaat jou informeert via een schermpje, geluiden of een bericht op je mobiel. Zorg dat je apparaat één van deze extra opties kan gebruiken.
- c. Nog een stap verder: je kunt het device ook signalen laten versturen naar regelapparaten. Zorg dat je apparaat verwarming, verlichting of ventilatie kan in- of uitschakelen (alsof je een lamp aan- of uitschakelt). Laat je device ook een alarm geven wanneer er schadelijke concentraties stoffen in de lucht zijn.

Aanwijzingen

1. **Vochtmeting**

Voor Arduino zijn diverse vochtsensoren beschikbaar. Een standaardsensor is bijvoorbeeld geschikt om in een bloempot te zetten en te meten of de plant water moet hebben. Met een kleine aanpassing kan hier een apparaat van gemaakt worden dat vocht in muren meet.

2. **Fijnstofmeting**

Bij de module Fijnstof zit een pakketje met fijnstofsensoren (lasergestuurd) die gekoppeld is aan fijnstofmeting in de lucht. Met kleine aanpassingen zou dit geschikt zijn om fijnstof in

huis te meten.

3. Lichtmeting

Arduino heeft diverse lichtsensoren. Hiermee is redelijk eenvoudig de lichtsterkte te bepalen. Met geschikte sensor ook in Lux.

4. Temperatuurmeting (ook temperatuur/vochtigheid)

Arduino heeft sensoren voor temperatuur. Prima geschikt als thermometer voor een ruimte.

5. Tocht / convectie

Voor tocht zou een schoepenrad gebruikt kunnen worden (op een rotatiesensor). Rook is wat ingewikkelder.

6. Gehalte CO2 in de lucht /Gehalte schadelijke gassen in de lucht (gecombineerde sensor)

Voor deze metingen zijn speciale sensoren beschikbaar.

7.2 Wetropolis: sensoren en actuatoren voor waterbeweging

Nederland heeft een ervaring van eeuwen met het leven met water. Toch is het nodig dat we onderzoek blijven doen hoe we moeten omgaan met het veranderende klimaat. Meer droogte, hevige regenval in korte tijd, stijging van de zeespiegel: het vraagt aanpassingen. Wat we het beste kunnen doen, wordt onderzocht met modellen. In het Verenigd Koninkrijk zijn vaak problemen met wateroverlast. In Leeds zorgde de rivier Aire ervoor dat de stad in korte tijd blank stond. Wat er moest gebeuren om dat te voorkomen wist niemand. Onderzoekers hebben toen een model gebouwd van de rivier en de stad, om te onderzoeken wat de beste aanpak was. Dat model werd Wetropolis genoemd. In Nederland zijn we dit model verder gaan ontwikkelen om ook in het onderwijs te gebruiken. In het model stroomt water, kunnen overstromingen ontstaan en wordt gekeken hoe regenval en waterafvoer het waterpeil in het landschap verandert. Om het model realistisch te laten werken is het nodig om regenval te simuleren (dus: water op het model laten regenen), het waterpeil in een rivier, sloot of bodem te meten, stroomsnelheid te bepalen.

Ontwikkel op basis van je kennis over digitale sensoren en meetprincipes een systeem waarmee:

- 1 Het waterpeil in het model nauwkeurig en automatisch kan worden gemeten
- 2 De stroomsnelheid van water automatisch kan worden gemeten
- 3 De afvoersnelheid en afgevoerde hoeveelheid water door buisje (regenpijp, riool) kan worden gemeten in de tijd
- 4 De aangevoerde hoeveelheid rivierwater kan worden geregeld en gevarieerd
- 5 Regenval nagebootst kan worden.

Alle sensoren en actuatoren moeten uitgelezen worden en data moet op één plaats verzameld worden. Kijk voor informatie over technieken van sensoren en modelbouw op wetropolis.nl.



Figuur 7.1 Originele Wetropolis model voor rivierwater in de stad

7.3 Digitale devices die een oplossing zijn bij opwarming in de stad

In de grote steden zal bij klimaatverandering de temperatuur harder toenemen dan in het omliggende landelijk gebied. De NLT module “summer in the city” gaat hier helemaal over. Voor sommige van die gevolgen zou je een digitaal device kunnen maken dat de gevolgen hiervan vermindert.

Opdracht:

Ontwerp een apparaat dat ons leven fijner maakt als de steden flink gaan opwarmen.

Voorbeelden zijn:

- Vogelhuisjes met airco
- Drinkbakjes voor dieren die door een sensor water geven als het bakje leeg is en er een dier wil drinken
- Drinkbekers die meten hoeveel er gedronken wordt en een signaal geven als je te weinig drinkt
- Zweetdetectiesysteem in je kleding om je te waarschuwen als je teveel hebt gezweet en mogelijk gaat stinken
- Een systeem dat meet welke vuilnisbakken in het park te vol zitten en dus mogelijk erg gaan stinken. Vervolgens geeft het systeem een waarschuwing af zodat er gericht geleegd kan worden
- Een systeem dat de troebelheid in vijvers meet (met mogelijk in de zomer blauwalgrisco) en waarschuwt als het te troebel wordt (zie figuur 7.2).

Er zijn nog veel meer dingen te bedenken. Bovenstaande apparaten zijn ook al echt door leerlingen gebouwd, dus het kan echt.



Figuur 7.2 Een device dat leerlingen gemaakt hebben om troebelheid in vijvers te meten

Stappenplan

1. Bedenk een concreet probleem dat met technologie opgelost kan worden.
2. Stel de eisen op die je aan het apparaat stelt.
3. Breek het ontwerp op in stukken:
Welke zaken wil je meten met sensoren?
Wat voor effect moet je apparaat hebben?
Hoe moet de data opgeslagen worden?
Moet de data verzonden worden?
Door eerst elk probleem apart op te lossen kom je uiteindelijk sneller bij een totaaloplossing.
4. Test het ontwerp en pas het aan tot je tevreden bent

Wellicht kunnen jullie in de klas een wedstrijd doen wie het beste ontwerp bouwt voor een probleem. Op die manier wordt het ook belangrijk dat je je eigen ideeën geheim houdt.

7.4 De Q-strip: meten van transpiratievocht

7.4.1 Introductie

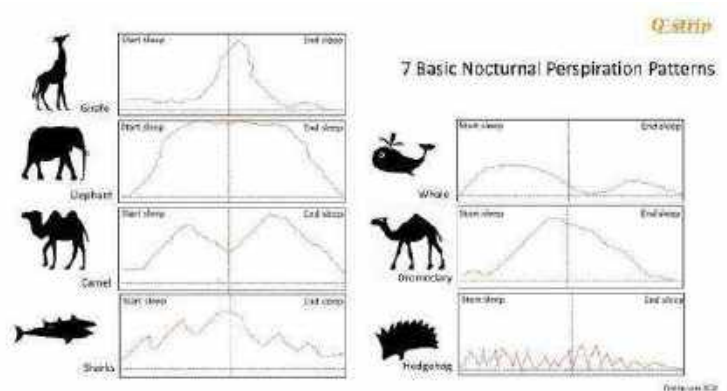
Metten aan zweten

Het meten van vocht heeft veel zinvolle toepassingen. Eén daarvan is het meten van het vocht dat we zelf afgeven via onze huid: transpiratievocht. Nuttig om te kunnen afkoelen als we het warm hebben, lastig als we fietsen in een waterdicht regenpak of 's nachts in ons bed badend in het zweet wakker schrikken.

Het kan bijvoorbeeld voorkomen dat je ziek wordt, bijvoorbeeld door griep. Als je lichaamstemperatuur omhoog moet, probeer je zoveel mogelijk warmte vast te houden. Je gaat rillen en krijgt kippenvel. Je kruipt in bed. Als je het te warm hebt gaat je lichaam flink transpireren (zweeten). Dat gebeurt ook als de koorts weer zakt of als je een paracetamol of aspirine hebt geslikt.

Tijdens de nacht produceren we vocht door transpiratie, soms tot wel een liter per nacht. Dat gaat niet heel gelijkmatig en is ook niet bij iedereen hetzelfde. Als het gelijkmatig en niet hinderlijk is, spreken we over nachtelijke transpiratie. Daarin zijn bij mensen zo'n zeven patronen te herkennen, waarbij de vorm naar een diersoort genoemd is die er op lijkt.

Als je er last van hebt en er bijvoorbeeld wakker van wordt, praten we over nachtzweeten. Ook bij veel diersoorten blijkt nachtelijke transpiratie in patronen voor te komen.



Figuur 7.3 Patronen voor nachtelijk zweten, gemeten met Q-strip

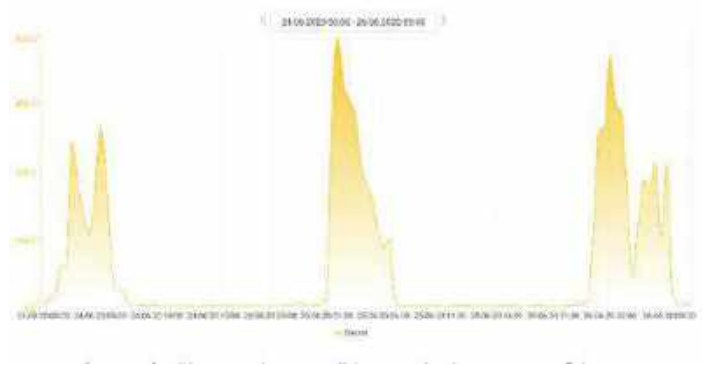
Nachtzweeten is ook een symptoom van verschillende ziekten. Het komt voor bij sommige vormen van kanker (zoals leukemie). Overmatig zweten komt ook voor in de dagen voorafgaand aan een hartinfarct. Zweet is daarom een steeds populairdere informatiebron voor de medische wetenschap. Andere informatiebronnen zijn bijvoorbeeld bloed, urine, speeksel en tranen. Naar de laatste twee vormen wordt nog veel onderzoek gedaan.

Momenteel worden wereldwijd diverse instrumenten ontwikkeld om de samenstelling van zweet te kunnen meten. Inmiddels is bekend dat je diverse stoffen in het zweet terug kunt vinden: ionen als natrium, kalium, chloride en ammonium, alcohol, melkzuur, peptiden en eiwitten (Bron: https://en.wikipedia.org/wiki/Sweat_diagnostics). Deze stoffen noemen we *biomarkers* en vertellen ons veel over het functioneren van het lichaam. Als een waarde afwijkt van wat normaal is, kan dat betekenen dat iemand ziek is. Biomarkers zijn dus heel belangrijk bij het ontdekken en bestrijden van ziekten. In Nederland wordt veel onderzoek gedaan naar biomarkers.

Naast de samenstelling van het zweetvocht is ook van belang op welke momenten dat zweet wordt geproduceerd. Het gaat dan niet om nachtzweeten, waarvan je wakker wordt, maar de normale onbewuste zweetpatronen. Die kunnen gemeten worden met de Q-strip. Dat is een eenvoudige sensor van papier en geleidend materiaal, die je op het matras kunt leggen. Een kleine computer meet de geleiding en daarmee de hoeveelheid en moment van zweetproductie.

Onderzoek doen

Dat zweetproductie een informatiebron is weten we, maar wat die zweetproductie ons kan vertellen is nog niet duidelijk. Er is nog nauwelijks wetenschappelijk onderzoek naar gedaan, voornamelijk omdat meten aan zweten heel complex is. In het verleden werd dat gedaan door proefpersonen te laten slapen in kamers met een temperatuur van 30 graden Celcius (pfff). Met de Q-strip wordt dat een heel stuk eenvoudiger, omdat die gevoelig is voor de normale zweetproductie bij de normale slaapt temperatuur.



Figuur 7.4 Meetresultaten Q-strip

De Q-strip is een eenvoudige sensor die door iedereen zelf te maken is, en geschikt is voor het zelf meten en data verwerken. Ook de meetcomputer is zelf in elkaar te zetten en te programmeren, en via het internet kunnen de data verzameld en geanalyseerd worden. Daarmee is de Q-strip een ideaal digitale technologie-project. Het onderzoek dat je hiermee doet kan een belangrijke bijdrage leveren aan het voorkomen en genezen van talrijke nare ziekten. Doe je mee?

In de opdracht is een aantal onderdelen te herkennen:

- Standaardopdracht: maken en meten
- Uitzoeken hoe de sensor optimaal gemaakt kan worden
- Meerdere sensoren gebruiken
- Wearables voor dataverzameling overdag
- Verzamelen en transport van data
- Analyse van meetdata om te ontdekken wat de transpiratiepatronen betekenen
- Privacy en beveiliging van data



Figuur 7.5 Plaatsing van een Q-strip sensor op matras

Zelf maken van een Q-strip sensor

A Opdracht:

Maak en test een kleine Q-strip-sensor

Benodigdheden:

- Keukenrol (geribbeld, 1300 x 40 ribbon)
- Koolstofvezel 4K (200-400 TEX)
- Uierzalf (gebruik géén vaseline)
- Wateroplosbare (knutsel)lijm
- Schaar (scherp)
- Strijkijzer
- Papier

Smeer de koolstofdraden dun in met uierzalf. Dit zorgt ervoor dat de 7 micrometer dunne koolstofvezeltjes aan elkaar blijven kleven. Ook kun je zo een bundel koolstofvezel splitsen in dunnere bundels (800 TEX kan gesplitst worden in 4 draden van 200 TEX).



Figuur 7.6 Een Q-strip sensor en meetdevice (Arduino en vochtsensor ingebouwd)

[NB: Een droge bundel koolstofvezel laat bijna onzichtbare dunne draadjes los. Ze kunnen kortsluiting geven in elektrische apparaten, maar ook in je Q-strip, waardoor die niet meer werkt. Het dragen van adembescherming wordt aangeraden. Als je met een botte schaar knipt ontstaan er ook losse vezeltjes].

Draai bundels in elkaar om een gelijkmatige draad te krijgen. De uierzalf helpt ook om goede geleiding tussen papier en koolstof te krijgen voor een goed werkende Q-strip.

De DIY Q-strip is een sensor die je maakt van een strook papieren tissue met 2 dunne koolstofdraden over de volledige lengte van de strip. De koolstofvezels moeten een afstand hebben tussen 22 en 24 mm. Het is de kunst om de koolstofdraden te stabiliseren, alsof ze deel uitmaken van het tissuepapier.

Het maken van de Q-strip sensor vraagt enige oefening. Begin met één vel keukenpapier en knip daarvan een strook van 9 cm breed. Vouw de randen van de strook van 9 cm naar binnen, zodat een 3 cm breed lintje ontstaat.

Leg de koolstofdraden in de strip van keukenpapier (22 - 24 mm afstand). Zorg dat de draden van koolstofvezel aan een zijde enkele centimeters uit het papier steken. Breng lijm aan tussen de koolstofdraden. Vouw de strip dicht en druk stevig aan (de lijm trekt in het gevouwen papier). Leg een vel papier op de strip en strijk papier, lijm en vezel tot één geheel met een droog strijkijzer (warmste stand). Leg voor het strijken een vel papier op de strip en zorg voor een warmtebestendige ondergrond!.

[Een andere manier om de lijm nauwkeuriger aan te brengen is als volgt: Knip ook een strookje tissue van 2 cm breed. Breng daarop ruim lijm aan en verdeel de lijm over het hele oppervlak van het strookje. Leg vervolgens het strookje keukenpapier met lijm tussen de koolstofvezels, vouw de strip dicht en strijk vast.]

Gebruik de Q-strip testunit, of een vochtigheidssensor, met een display om te testen. Druk (of plak met een plakbandje) de uiteinden van de koolstofdraden op de metalen strips van de meetunit en lees de geleiding af. [De geleidingswaarden zitten in deze range: droog 0, vochtig 50-350 en nat 600-800].

Als de draden goed geïsoleerd zijn, kan er alleen stroom van de ene naar de andere draad komen als er vocht in de tissue terecht komt. De Q-strip-transpiratiesensor is dan klaar voor gebruik.

De koolstofvezel kan met geïsoleerd metaaldraad verlengd worden (solderen is onmogelijk). Vlecht de draden koolstof en metaal in elkaar en zet ze vast met lijm of tape, een krimpkous of elektrisch geleidende verf. Bevestig de metaaldraad aan de vochtsensor van de Arduino.

[Tip: Je kunt ook werken met diverse andere materialen om (kleine) sensoren te maken, in combinatie met koolstofvezel. Denk aan filterpapier, gewoon plakband, voorgelijmd papierplakband, tissue, stofvrije tissue, papierpulp (te maken van tissue of toiletpapier).]

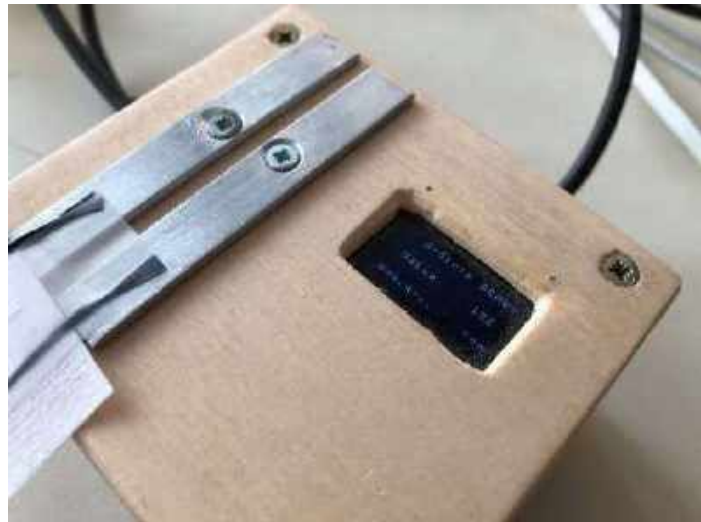
B Opdracht:

Maak een lange Q-strip

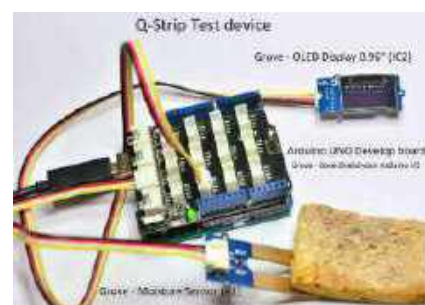
Gebruik dezelfde werkwijze, maar plak nu drie tot vier stroken (9 cm breed) aan elkaar tot een lint van 1 meter lengte. Maak koolstofvezels van 90 cm en verbind ze met geïsoleerde metaaldraad. De overgang koper-koolstof zit ongeveer op 10 cm van het einde van de strip. Het sensorgedeelte van de strip is dan zo breed als een matras.

Voor langdurig gebruik kan de Q-strip in een drager geschoven worden die voldoende stevigheid geeft om de Q-strip onder een laken te kunnen leggen en gedurende de nacht te gaan meten. Wij gebruiken de strip nu zonder extra drager.

De Q-strip zal niet super precies zijn, maar je kunt hele behoorlijke resultaten behalen als je een strip gedurende langere tijd (zeg 14 dagen) gebruikt en kun je iets zeggen over het nachtelijke ritme van de transpiratie van je lichaam.



Figuur 7.7 Een testdevice voor vochtsensor: de metalen strips maken contact met de Arduino-vochtsensor in het kastje en het display laat de geleidbaarheid zien. Het binnenwerk van deze device kun je zelf bouwen met de instructies.



Figuur 7.8 Arduino met Groove shield, vochtsensor en display. Dit zit ingebouwd in de testdevice van figuur 7.3.5, en kun je zelf bouwen met de instructies.

7.4.2 Het Q-strip-meetsysteem bouwen

Benodigheden testdevice

1. Arduino UNO
2. Groves Base shield voor Arduino (v2) HAT
3. Grove OLED display 0.96 (IC2), 128 x 64 display
4. Groove Moisture Sensor (A1)
5. Voeding (5V) USB via lader of computer
6. Code: Q-strip.ino

LoRaWAN gekoppeld meetdevice

1. Arduino TTN Uno (standaard Arduino met LoRaWAN Chipset)
2. Groves Base shield voor Arduino (v2) HAT
3. Groove Moisture Sensor (A1)

Methode

1. Maak verbinding tussen Q-strip en pootjes van de vochtsensor (bv: klem, drukknoop, vleugelmoer en oogje)
2. Leg verbinding aan met het LORA netwerk (Internet of Things) en The Things Network.
3. Stel gateway in voor de Q-strip-database.

C Opdracht

Gebruik het Q-strip-device

De sensor voor het Q-strip-device zijn de lange koolstof-electroden. De weerstand tussen de elektroden hangt af van de isolatie tussen de elektroden. Als het tissuepapier tussen de elektroden droog is, dan is de weerstand hoog. Hoe vochtiger, hoe beter de geleiding, dus hoe lager de weerstand. De electrodestrip wordt aangesloten op een vochtsensor van de Arduino. Normaal gesproken bestaat die sensor uit metalen elektroden die, bijvoorbeeld, in de grond worden geprikt. Nu zijn de metalen elektroden vervangen door de koolstofdraden in de sensorstrip. De verwerking van de electrode data met de Arduino gebeurt op dezelfde manier.

Om een relatie te kunnen leggen tussen gemeten waarde en de hoeveelheid opgenomen vocht, is het nodig de Q-strip te kalibreren. Breng een hoeveelheid vocht (bv 1 mL) op de strip aan, wacht tot het vocht verspreid is en noteer de geleiding. Breng telkens meer vocht aan tot de strip verzadigd is. Maak hiervan een kalibratiegrafiek.

Om metingen uit te voeren gedurende de nacht, installeer je de Q-strip onder het laken van je bed, ter hoogte van je borstkas. Bevestig het meetkastje en zorg dat deze elke 10 minuten een meting doet. Vaker of minder vaak kan uiteraard ook. De gegevens kun je opslaan op een geheugenkaartje (SD) of laat je doorsturen via het LoRa netwerk (LoRaWAN) naar The Things Network.

D Opdracht: Verwerken van meetdata

1. Ophalen van data uit de database met metingen
2. Opslaan van lokale data op een SD kaartje
3. Analyse van meetgegevens

7.4.3 Onderzoek en toepassing

E Verdiepingsopdracht

Gebruik het Q-strip-systeem om onderzoek te doen naar transpiratie in de nacht. Denk daarbij aan de volgende zaken:

- Proefpersonen (hoeveel, welke leeftijd, verschil en overeenkomst)
- Data verzamelen en verwerken
- Bescherming van privacy
- Wat betekent een transpiratiepatroon?
- Hoe kan het Q-strip systeem handig ingezet worden?

F Verdiepingsonderzoek: De vorm en techniek van de vochtsensor

- Is het mogelijk de sensor in een continuproces te maken (vanaf de rol)
- Welke papiersoort is het meest geschikt om de sensor te produceren?
- Werken kleinere /anders gevormde vochtsensoren ook?
- Kan de sensor van stof (bijvoorbeeld katoen) gemaakt worden?
- Welke soort lijm is het meest geschikt voor het plakken van de sensor?
- Is de vochtsensor draagbaar te maken, om bijvoorbeeld in te bouwen in een shirt, onder de armen, buik, rug, voorhoofd?

G Verdiepingsopdracht: Meten met de Q-strip

- Kan het meetsysteem in bed gebruikt worden om meer gegevens te verzamelen, bijvoorbeeld draaien, lichaamstemperatuur, luchtvochtigheid, omgevingstemperatuur, vochtdoorlatendheid van matras.
- Kan de meetdata van de sensor gekoppeld worden aan andere draagbare sensoren (bijv een meethorloge, meetring of -armband?)
- Hoe vaak moet de meetdata verzameld worden voor een goed beeld?
- Hoe snel reageert de sensor op de transpiratie / temperatuurverandering?
- Kan met de Q-strip méér gemeten worden dan alleen vochtigheid?

Informatiebron:

<https://washingtoncitypaper.com/article/221338/straight-dope-do-you-really-sweat-one-liter-each-night/>

7.5 Andere thema's voor de eindopdracht

De thema's waar je aan kunt werken zijn eindeloos. Enkele suggesties hiervoor zijn:

- Een digitaal gestuurde spuitenpomp (bij de module Lab on a chip)
- Meten met licht in een Lab on a chip (met een led, fotodiode en Arduino)
- Stuur een 3D-printer aan met Arduino
- Maak een zelfrijdend wagentje of robot, op basis van een microcontroller als Arduino, Micro:bit, Raspberry Pi of ander platform.
- Wellicht heb je zelf een idee waar je aan wilt werken. Vraag je docent wat er mogelijk is.
- Je kunt bij verschillende modules die je gedaan hebt (of misschien nog gaat doen) een ontwerp maken waar je digitale technologie voor gebruikt.

Niet alle opdrachten kunnen we in de module plaatsen. Daarom worden er ook na verschijnen van de module losse eindopdrachten beschikbaar gesteld.

8 Verder met digitale technologie

Sommige van onderstaande lesmodules nlt gaan in zijn geheel over een onderdeel van digitale technologie. Bij anderen wordt digitale technologie ingezet in een context. Ook kun je bij een groot deel van de nlt modules iets maken of onderzoeken waar digitale technologie bij gebruikt wordt. Vraag er gerust naar bij je docent of TOA.

Fijnstof

Nederlanders overlijden gemiddeld 9 maanden vroeger door fijnstof. Hoe zit het met de luchtkwaliteit rondom je school? Is dit aanleiding tot een vervolgplan van deze burgerwetenschapsaanpak? Samen op zoek naar oplossingen voor een betere luchtkwaliteit en een steentje bijdragen aan het Schone Lucht Akkoord. GLOBE Nederland en het RIVM ontwikkelden deze lesmodule om op middelbare scholen de fijnstofconcentratie te meten in hun omgeving. Doe mee met de fijnstofcampagne!



Modelleren



Modellen helpen de mens om voorspellingen te doen en bedreigingen beheersbaar te maken. In deze module ontdek je hoe situaties en processen met behulp van dynamische modellen beschreven kunnen worden. De contexten waar je aan werkt zijn de verspreiding van corona onder de bevolking, de waterhuishouding in Nederland, de groei van de wereldbevolking en/of het saneren van een vervuilde bodem.

Bio-informatica

Het concept genexpressie (transcriptie, splicing, translatie) wordt vanuit programmeren in Python benaderd. Hierbij wordt het biologisch concept vertaald naar praktische programmeer opdrachten. Via bestaande tools worden beginselen van het lezen van DNA geoefend. In een individuele eindopdracht wordt naar een onbekend gen(product) en de genetische oorzaak van een afwijkend genproduct gezocht.



De toekomst van de landbouw



Nederland staat aan de vooravond van een grootschalige herinrichting van de land- en tuinbouw. Het moet slimmer, efficiënter, schoner, met minder energie, minder landoppervlak en vooral ook meer ecologisch. In deze lesmodule verdiep je je in technologische innovaties, die in de landbouw ingezet worden. Robots die planten automatisch herkennen en verwijderen, spuiten tegen insecten, bemesten en watergeven gebeurt op basis van Big-Data van sensoren van o.a. drones. Je bouwt een eigen visie op over het complexe systeem van schakels binnen de voedselproductie en -distributie. Vanuit die visie werk je toe naar een toekomstplan, waarin al deze aspecten een rol spelen. Zo draag je positief bij aan de toekomst.

Cybersecurity

Cybersecurity gaat over het beveiligen van digitale informatie en digitale diensten die een bepaalde waarde hebben. Die waarde zorgt ervoor dat je de informatie of dienst wilt afschermen. Uiteindelijk gaan diensten over het lezen of wijzigen van digitale gegevens: de gegevens over je bankrekening, al je e-mails of de inhoud van je Facebook pagina. 'Cyber' in cybersecurity benadrukt extra dat het gaat om security in cyberspace, de virtuele wereld die bestaat uit computers en de netwerken tussen computers.



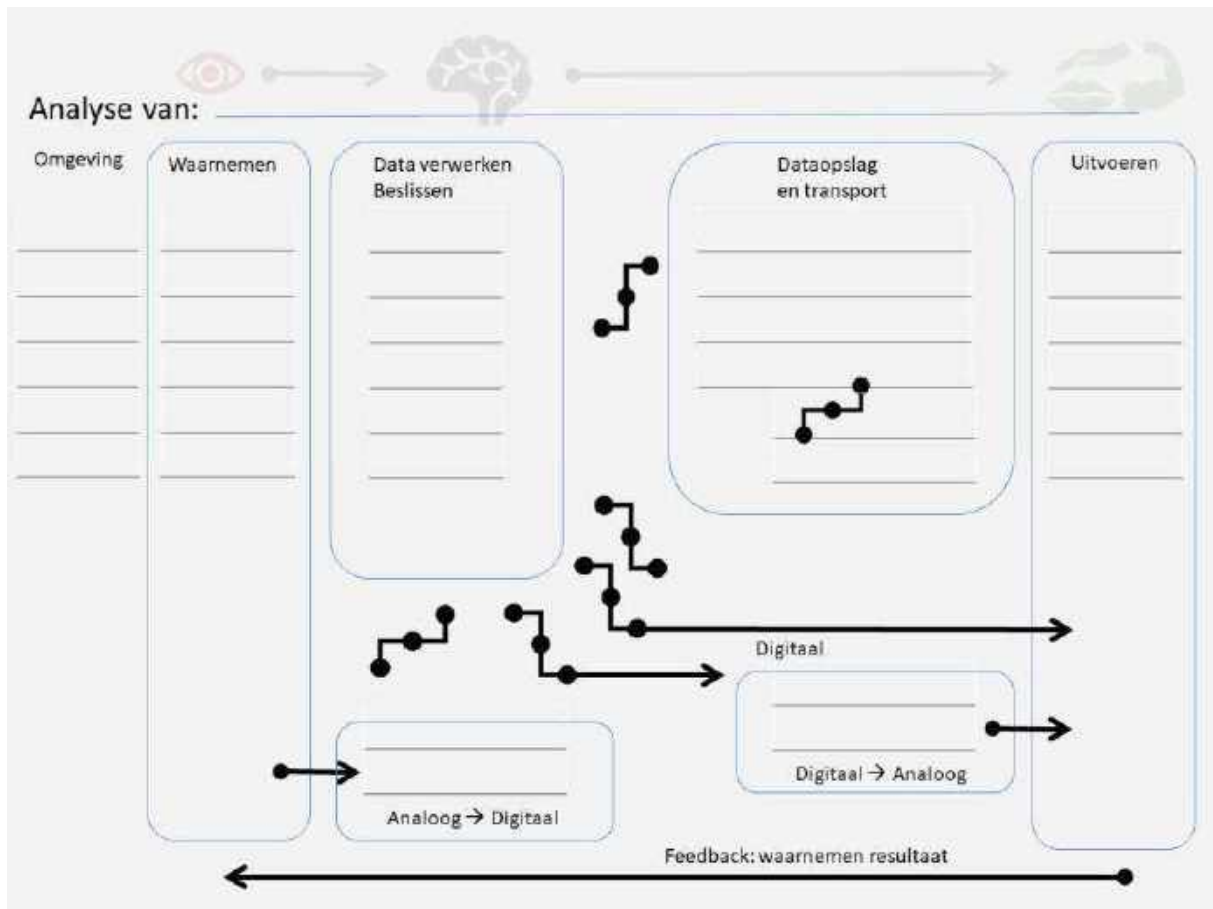
Artificiële Intelligentie (AI)

Een lesmodule waarin de leerlingen een positief project doen: ze zetten AI in om de wereld iets beter te maken. Na afloop van de module heeft de leerling geleerd dat AI een breed vakgebied is, dat gaat van het programmeren van algoritmes tot het doen van medisch onderzoek. Het is een samenspel van verschillende disciplines: spraakherkenning, machine learning, beeldherkenning en natural language processing. Kunstmatige intelligentie wordt gebruikt als je een verzekering wilt afsluiten, als je zit te 'netflixen', als je via social media contact houdt met je vrienden en op nog veel meer manieren.



Leeg analyseschema

Ook als los werkblad beschikbaar in de moduledatabase (<https://module-database.betavak-nlt.nl/>).



Bijlagen verkrijgbaar in de moduledatabase

- Bijlage 1: Uitgebreide uitleg bij onderdeel C van H4
- Bijlage 2: Uitgebreide uitleg bij onderdeel C van H5
- Bijlage 3: Uitgebreide weergave van achtergronden en verdieping zoals
Fabricage van computerchips, printplaten
Hoe transistoren werken
Computergeheugen
Logische schakelingen
Sensoren, actuatoren
Neopixel
Verschillende microcontrollers
- Bijlage 4: Uitgebreide begrippenlijst
- Bijlage 5: URL lijst

Er is voor de leerlingen ook veel informatie te vinden op de leerlingenwebsite:

https://maken.wikiwijs.nl/171772/Schakelmodule_Digitale_Technologie_online_materiaal