

Team Smart Lights

Michiel Vandewiele, Axel Vroman, Sofie Verscheuren, Lisa De Vuyst

MELICTes - april 2022

1 Opdrachtomschrijving

In het thema "sustainable campus" werken we voor dit labo een project uit om energieverpilling te voorkomen door lampen die onnodig aan staan in lokalen. We meten aan de hand van een lichtsensor of het licht aan of uit staat in het lokaal. Ook wordt er gemeten of er mensen aanwezig zijn in het lokaal of niet aan de hand van een bewegingssensor. Als laatste wordt er aan de hand van een lichtsensor gekeken hoeveel licht er van buitenaf komt. Met de informatie van de 3 sensoren zullen we een alarm kunnen laten afgaan in het geval dat er geen mensen aanwezig zijn in het lokaal maar het licht wel nog aanstaat, of wanneer er voldoende zonlicht van buitenaf schijnt waardoor het overbodig is om binnenverlichting aan te steken. Als alarmsignaal gebruiken we een led die centraal in het lokaal zichtbaar zal zijn. Ook zal dit alarmsignaal doorgestuurd worden naar het centraal systeem met de volledige klas.

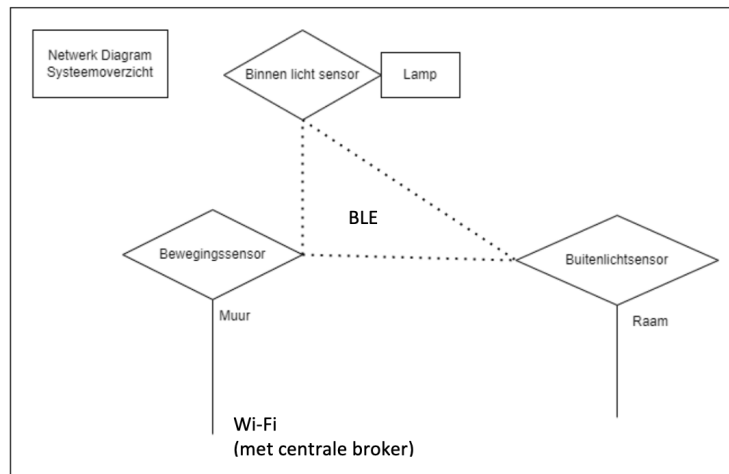


Figure 1: Systeemoverzicht.

2 Onderdelen

2.1 Microcontroller

Als microcontroller zullen we gebruik maken van ESP-32's, meer bepaald ESP32-WROOM-32E.

2.2 Binnenlicht

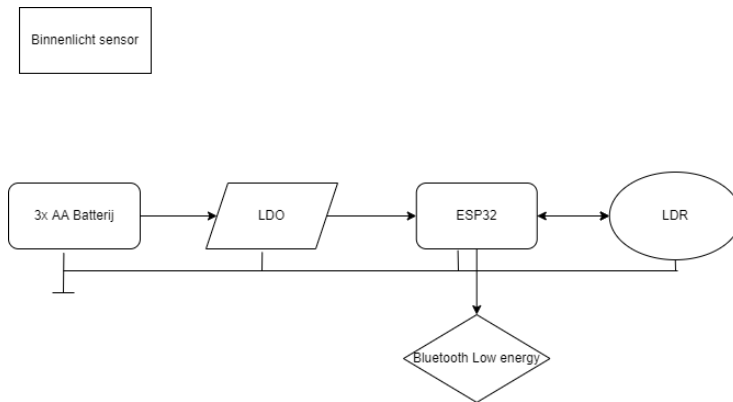


Figure 2: Blokschema van de binnenlichtsensor.

Het binnenlicht gaan we meten aan de hand van een light dependent resistor (LDR). Deze zal binnen zo dicht mogelijk bij een lamp geplaatst worden, bij voorkeur een lamp zo ver mogelijk van een raam om geen invloed te hebben van het buitenlicht.

- **Wat wordt er gemeten?** De intensiteit van het licht vinden we niet belangrijk, we willen enkel weten of dat de binnenverlichting aan of uit staat.
- **Hoe vaak wordt er gemeten?** De metingen gebeuren om de 5 minuten. Vaker meten is overkill - we gaan er namelijk van uit dat in een praktisch scenario het licht in een klaslokaal amper aan- of uit gezet wordt: aan het begin of einde van een les bijvoorbeeld.
- **Hoe wordt de data doorgestuurd?** Voor deze sensor is het niet nodig getalwaarden door te sturen buiten een 0 of een 1. Dit omdat we, zoals eerder vermeld, enkel geïnteresseerd zijn in de toestand van het licht: staan de lampen aan of uit? De data wordt enkel doorgestuurd naar de centrale ESP-32 (dit is die met de bewegingssensor) wanneer er een verandering is van toestand: 0 wordt 1 of 1 wordt 0. Het licht is dus aan- of uitgezet.

2.3 Buitenlicht

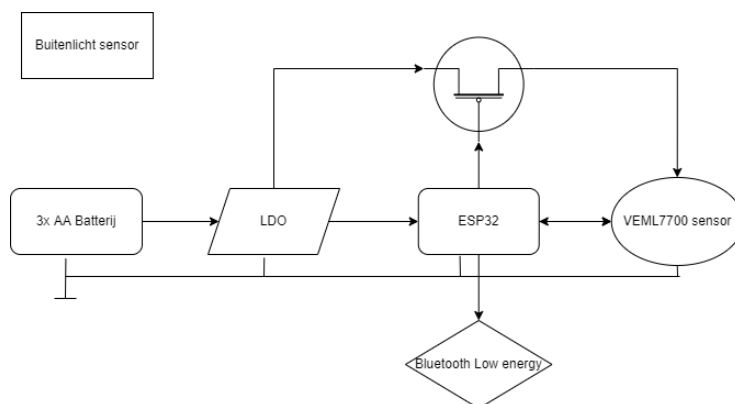


Figure 3: Blokschema van de buitenlichtsensor.

Het buitenlicht wordt gemeten aan de hand van een nauwkeurigere sensor dan de LDR, namelijk de VEML7700. Deze wordt geplaatst bovenaan aan de binnenkant van het raam en wijst naar buiten. We stellen zelf randvoorwaarden op: we gaan uit van een normaal, niet te groot klaslokaal. Er mag dus bijvoorbeeld geen extra scheiding mag zijn tussen het raam en het lokaal, zoals het geval is in lokaal B230. Als optie kunnen we een vergelijkende studie doen tussen de LDR en de VEML7700 om te kijken of de LDR ook voldoende zou kunnen zijn om het buitenlicht op te meten.

- **Wat wordt er gemeten?** We meten de intensiteit van het buitenlicht.
- **Hoe vaak wordt er gemeten?** Analooog aan het binnenlicht wordt er hier ook elke 5 minuten gemeten. Vaker meten is hier ook overkill. Het heeft ook geen zin om vaker een verschil te detecteren: lichten binnen enkele minuten/seconden aan- en terug uitschakelen omdat er een wolk passeert heeft praktisch geen enkel nut.
- **Hoe wordt de data doorgestuurd?** Er wordt een drempelwaarde gekozen, wat deze drempelwaarde is wordt experimenteel bepaald. Het is niet nodig de lichtintensiteit zelf door te sturen, enkel of de lichtintensiteit voldoende is buiten. Analooog aan de binnenlichtsensor wordt hier ook enkel een 0 of 1 doorgestuurd, wanneer er een verandering is van toestand.

2.4 Bewegingssensor

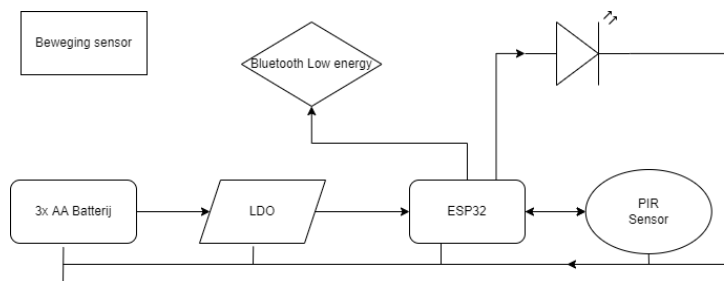


Figure 4: Blokschema van de bewegingssensor.

De bewegingssensor wordt centraal in het lokaal aan het plafond gemonteerd. Om te detecteren of er mensen aanwezig zijn in het lokaal gebruiken we een passieve infrarode sensor (PIR).

- **Wat wordt er gemeten?** We meten of er beweging is - of er dus personen aanwezig zijn in de ruimte.
- **Hoe vaak wordt er gemeten?** De sensor staat standaard in sleep en zal zichzelf in stand-by zetten wanneer er een high status bereikt wordt.
- **Hoe wordt de data doorgestuurd?** Ook voor deze sensor is het niet nodig getalwaarden door te sturen, we moeten enkel doorsturen of er wel of niet beweging gedetecteerd is aan de hand van een 1 of 0 signaal. Omdat de ESP-32 met deze sensor gebruikt wordt als centrale ESP, en dus de data van de andere twee sensoren naar hier wordt gestuurd, moet eigenlijk niets doorgestuurd worden. Er wordt wel nog een alarmsignaal doorgestuurd wanneer het licht uitgezet moet worden, wat besproken wordt in het volgende puntje.

2.5 Alarm

Zoals eerder vermeld gebruiken we als alarm een led aanwezig op de ESP-32 met de bewegingssensor, net als een signaal naar het klassikale systeem. Er is gekozen om de led te plaatsen op de pcb van de bewegingssensor zodat deze centraal staat - in het midden van het lokaal aan het plafond - en zo dus het best zichtbaar is.

Het alarm kan afgaan in 2 situaties. Wanneer het licht brandt en er geen mensen aanwezig zijn of wanneer het licht brandt maar er voldoende zonlicht door de ramen schijnt. In het tweede geval zal het alarm eerder een indicatie zijn zodat de persoon aanwezig in het lokaal zelf het licht kan uitdoen en kan ondervinden of er effectief genoeg licht is, of dat hij liever het licht aan laat staan.

Indien het ledje begint te branden zal dus een persoon manueel het licht moeten

uit doen. Dit heeft uiteraard enkel zin in het geval waar er voldoende buitenlicht is om het licht uit te doen wanneer er mensen aanwezig zijn in het lokaal. Wanneer het licht uit moet omdat er niemand aanwezig is, is er dus ook niemand om op de lichtknop te drukken - hiervoor dient het centrale systeem dus.

3 Communicatiemiddel

3.1 Technische overweging

Er zijn heel wat manieren om wireless te communiceren. In de afbeelding hieronder staan verschillende methodes uitgebeeld aan de hand van hun data rate en afstand.¹

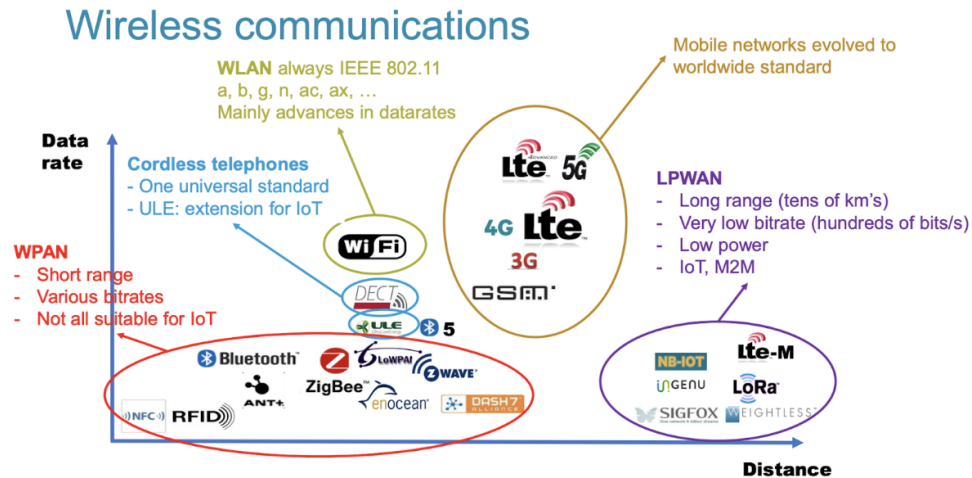


Figure 5: Mogelijke manieren om draadloos te communiceren.

De cordless telephones en mobile networks gaan we buiten beschouwing laten. We bespreken dus WLAN (Wireless Local Area Network), WPAN (Wireless Personal Area Network) en LPWAN (Low Power Wide Area Network).

¹Bron afbeelding: slides les "Short range wireless technologies for IoT" (prof. Lieven De Strycker)

WLAN vs WPAN vs LPWAN

Op het eerste zicht lijkt WPAN de geschikte oplossing: we hebben maar een range nodig van enkele meters aangezien de sensoren zich in eenzelfde klaslokaal bevinden. De data rate hoeft ook niet hoog te zijn, we willen ook enkel 0 of 1 kunnen doorsturen bij een verandering.

3.1.1 WPAN methodes

Hieronder worden verschillende short-range draadloze technologieën vergeleken.²

Comparison with other short-range wireless technologies

Feature	NFC	Bluetooth (2.1 + EDR)	ZigBee	WLAN (802.11n)	RFID	QR code
range	10 cm	100 m (class 3)	+ 10 m (indoor)	100 m	> 1 m	depends on camera and size
frequency	13,56 MHz	2,4 GHz	2,4 GHz	2,4 GHz 5 GHz	13,56 MHz but also LF and UHF	N.A.
data rate	848 kbit/s	3 Mbit/s	250 kbit/s	600 Mbit/s	848 kbit/s	N.A.
setup time	< 10 ms	seconds	seconds	seconds	< 10 ms	N.A.
passive ?	yes	no	no	no	yes	yes
applications	smart posters, mobile payment, ticketing	audio streaming, file exchange	building automation, distributed sensing	wireless LAN	product tracing, identification	smart posters

Figure 6: Comparison with other short-range wireless technologies.

NFC en RFID kunnen we meteen uitsluiten aangezien de range té kort is hiervoor. Bluetooth, Zigbee en WLAN hebben wel een voldoende grote range. Zigbee heeft de laagste data rate - maar nog voldoende voor deze applicatie. WLAN heeft de hoogste data rate wat in een hoger energieverbruik resulteert. Aan de hand van deze tabel lijkt Zigbee de ideale oplossing. Typische toepassingen met Zigbee zijn namelijk ook systemen gelijkaardig aan deze.

Zigbee

Zigbee is gebaseerd IEEE's 802.15.4 personal-area network standaard en bestaat al meer dan een decennia. Het is een goed alternatief voor Wi-Fi en Bluetooth voor toepassingen die low-powered apparaten gebruiken dat geen hoge bandbreedte nodig hebben - ideaal dus voor smart home sensoren. Philips Hue gebruikt bijvoorbeeld dit protocol om connectie te maken tussen hun lampen.

²Bron afbeelding: slides les "Short range wireless technologies for IoT" (prof. Lieven De Strycker)

Bluetooth

Vanaf Bluetooth versie 4.0 is er een Bluetooth Low Energy (BLE) protocol aanwezig. Deze heeft kortere pakketjes wat zorgt voor een kortere TX/RX tijd. Er zijn ook minder RF kanalen aanwezig wat zorgt voor kortere discover en connection tijden.

ESP-NOW

ESP-NOW is een protocol door Espressif - dezelfde ontwikkelaars als de ESP-32. Dit protocol is, analoog aan Zigbee, ontwikkeld om te communiceren onderling zonder gebruik te maken van wi-fi. Zoals Bluetooth moet er eerst gepaired worden voordat er gecommuniceerd kan worden. Na het paren is de connectie peer-to-peer en veilig - er is geen handshake nodig.

3.1.2 WLAN methodes

Wi-Fi

Wi-fi kan gebruik maken van radiofrequenties in de 2,4GHz- en/of 5,0GHz-band. van wikipedia: De bandbreedte en het bereik van wifi zijn groter dan die van bluetooth. Om deze redenen is wifi een van de belangrijkste toegangsmethoden voor een alom aanwezig draadloos internet. Een keerzijde van wifi, met name vergeleken met bluetooth, is het relatief hoge energieverbruik. Dit is bij kleine apparaten met een beperkte batterijcapaciteit, zoals pda's, een probleem.

3.1.3 LPWAN methodes

LPWAN staat voor low power wide area network. Het wordt typisch gebruikt voor IoT devices die op grote afstand data moeten kunnen doorsturen. Er is een verschil tussen licensed en unlicensed LPWAN. Bij de licensed versie wordt er telkens een vaste frequentie gebruikt, bij unlicensed is dit niet het geval. Licensed LPWAN kan gebruikt worden voor connecties met hoge snelheid waarbij er grote publieke netwerken kunnen gemaakt worden over de hele wereld. Unlicensed daarentegen wordt gebruikt voor kleine privé netwerken. We kunnen dus concluderen dat we geen nood hebben aan licensed LPWAN aangezien we een privé netwerk willen en het niet nodig is om hoge snelheden te bereiken. LTE-M1, EC-GSM en NB-IoT kunnen we dus laten vallen in de vergelijking. Ook Ingenu is niet ideaal voor deze toepassing aangezien het eerder bedoeld is voor machine-to-machine communicatie. Wanneer we de keuze tussen LoRa en Sigfox zouden moeten maken lijkt LoRa ons de betere optie. De transmission power is ongeveer gelijk, LoRa gaat langer mee. Ook is een data rate van 50 kbps perfect. Sigfox en LoRa kunnen beide over een lang genoeg afstand connectie maken.

Sigfox

Sigfox gebruikt differential binary phase-shift keying (DBPSK) en Gaussian fre-

quency shift keying (GFSK) waardoor een communicatie via een radio band (868MHz) kan opgezet worden.

LoRa

LoRa staat voor Long Range, het is een communicatiemiddel met een laag verbruik en een lage data rate. Met een LoRa verbinding kan er tot 10km vergecommunicerd worden. LoRa gebruikt license-free radio frequentie banden, in België is dit 863-870 MHz. Elke end node/client heeft zijn eigen ID nummer ze zullen hun data sturen naar de gateway die de data online plaatst. LoRa bevat geen network protocols om het verkeer van data, data opslag te beheren. LoRaWAN is een protocol dat bovenop LoRa een network layer voorziet.

3.2 Praktische keuze

Nu we deze mogelijke communicatiemiddelen bekeken hebben, wat is nu voor deze situatie het beste?

Uit de datasheet³ halen we dat BLE en Wi-Fi in de ESP-32 zit - hiervoor is dus geen aparte module nodig wat zorgt voor een lager kostplaatje maar ook gemakkelijker is om mee te werken.

Zigbee is op deze campus niet mogelijk en LoRa duurt te lang waardoor we deze twee manieren uitsluiten.

We hebben volgende keuzes gemaakt: De klassikale communicatie zal via een Wi-Fi verbinding gebeuren. Onze interne communicatie zal via BLE verlopen.

4 Energieconsumptie

Voor de energieconsumptie van de sensoren wordt een ruwe schatting gemaakt van het verbruik van elk van de modules. Om een zo conservatief mogelijk resultaat te bekomen wordt in elk geval van een worst case scenario uitgegaan. Om de modules te voeden op 3,3 Volt wordt gebruik gemaakt van drie AA batterijen die in serie zullen worden geschakeld. Deze leveren een nominale spanning hoger dan 3.3 volt die doorheen de levensduur van de batterij langzaam zakt tot 3.3 Volt na ongeveer 80% van het batterijleven. Om de esp32 en sensoren niet onder een te hoge spanning te plaatsen wordt gebruik gemaakt van een LDO na de batterij. Een set van AA batterijen heeft een 2500 mAh als energie reserve, aan een vaste stroom van 100 mA gaan de batterijen tot 25 uur mee.⁴ Echter is de spanning van de batterijen niet meer voldoende na 80%, dus komt dit neer op 2000 mAh bruikbare energie. De drie modules buitenlichtsensor, binnenlichtsensor en bewegingssensor verbruiken elk een verschillende hoeveelheid

³Referenced from: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e-esp32-wroom-32ue_datasheet_en.pdf (01/04/2022)

⁴Referenced from: <https://www.batterijenhuis.nl/image/data/datasheets/duracell/Datasheet-Duracell-Ultra-Power-AA.pdf> (02/04/2022)

aan energie. De meest complexe van deze modules is de bewegingssensor omdat deze ook als hub wordt gebruikt. Deze zal ook het meest verbruiken van elk van de modules en zal dus zo ook de levensduur bepalen van de volledige opstelling voor er gewisseld moet worden van batterijen.

4.1 Bewegingssensor

Het gebruik van de bewegingssensor loopt als volgt. De module blijft in deep-sleepmode van de esp32 voor 5 minuten, hierbij is de stroom naar de PIR-sensor niet aanwezig. De deep sleep van de esp32 verbruikt 0,0001 mA.⁵ De esp wordt na deze 5 minuten voor een ogenblik wakker om de GPIO pin die de PIR-sensor aanstuurt te veranderen van staat en gaat hierna terug in deep sleep terwijl de PIR-sensor door de kalibratie loopt. Tijdens deze kalibratie gebruikt de EKMB PIR-sensor 0.006 mA, de kalibratie duurt 25 seconden.⁶ Hierna wordt de esp weer wakker om de sensor uit te lezen. Dit uitlezen duurt 10 seconden. In het geval dat een persoon stil zou zitten wordt verwacht dat deze om de 10 seconden detecteerbaar beweegt, echter is hier nog een praktische test voor nodig. Hiervoor kan de esp in modem sleep actief staan, dit verbruikt 20 tot 31 mA.⁷ De actieve detectie van beweging vraagt maximaal 0.106 mA wanneer beweging gedetecteerd wordt. Omdat er van het worst case scenario wordt uitgegaan wordt deze als de stroom gerekend tijdens het detecteren. Verder moet deze module ook de andere modules aanhoren in Bluetooth low energy. Dit wordt op een vastgezet punt op de RTC na system setup gedaan. De tijd nodig om de bluetooth connectie op te zetten en de data te ontvangen, wordt ingeschat op vijf seconden. Hiervan zijn vier seconden om de connectie op te zetten en te ontvangen en één marge seconde waarbij een mogelijke minimale fout op de RTC kan worden vermeden.⁸ Dit luisteren naar de andere modules moet ook telkens om de 5 minuten gebeuren en verbruikt 80 mA. Verder om het uur of wanneer er een alarm wordt getriggerd zou deze module ook een wifi MQTT bericht moeten sturen. Dit vraagt het maximale stroomverbruik van de esp 260 mA. Voor de connectie op te stellen en een bericht te sturen wordt ook 5 seconden genomen. Deze inschatting is zeer ruim genomen voor wat speling met een theoretische minimale tijd van 1-2 seconden.⁹ Na alles door te sturen of ontvangen te hebben gaat de module terug in deep sleep modus en na 5 minuten begint deze cyclus opnieuw. Uit de voorgaande beschrijving kan de mAh van de module worden berekend door het stroomverbruik (mA) te vermenigvuldigen met de duur en het aantal keer per uur dat dit gebeurt. Wanneer dit in een

⁵Referenced from: <https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/> (02/04/2022)

⁶Referenced from: https://www.mouser.be/datasheet/2/315/bltn_eng_papirs_1365490.pdf (02/04/2022)

⁷Referenced from: <https://www.mischianti.org/2021/03/10/esp32-power-saving-modem-and-light-sleep-2/> (02/04/2022)

⁸Referenced from: <https://www.novelbits.io/ble-connection-intervals/: :text=A%20BLE%20connection%20interval%20is,with%20increments%20of%201.25%20m> (02/04/2022)

⁹Referenced from: https://www.reddit.com/r/esp32/comments/gfroev/time_to_wake_up_and_connect_to_wifi_from_deep/ (02/04/2022)

tabel wordt geplaatst wordt Table 1: Verbruik PIR-sensor en hub bekomen. Dit bekamt een totaal van 2,731910889 mAh voor de module in het worst-case scenario.

4.2 Binnenlichtsensor

Een andere module is deze van de binnensensor. Hier wordt net zoals in de hierboven besproken bewegingssensor vijf minuten deep sleep van de esp32 gedaan. Hierna wordt over gegaan naar een actieve esp voor een seconde die de LDR uitleest. De seconde is ruim gerekend en kan waarschijnlijk nog geminimaliseerd worden. De LDR staat in parallel met een weerstand van 10k Ohm met een spanningsval van 3.3 volt wordt dus een maximale stroom van 0.33 mA verwacht. De waarde van de LDR wordt binnen een threshold geplaatst voor licht aan of licht uit. Indien deze status niet verandert, wordt er geen BLE bericht uitgestuurd om stroom te besparen. Voor dit project wordt eens per 2 uur een verandering van de status verwacht, dit omdat een les gemiddeld anderhalf uur duurt op campus en er dus wel of geen licht aan zal staan. Het versturen van een BLE bericht kost 120 mA. Na het al dan niet versturen van de data gaat de esp terug in deep sleep en begint de cyclus opnieuw. Wanneer de bovenstaande cyclus wordt gebruikt om het verbruik in mAh te berekenen wordt Table 2: Verbruik binnenlicht sensor bekomen met een totaal verbruik van 0,194810778.

4.3 Buitenlichtsensor

Zeer gelijkaardig aan de binnenlichtsensor wordt de esp eerst 5 minuten in deep sleep gehouden en wordt dan gekeken of de lichtsensoren veranderd is van threshold voor het buitenlicht. Omdat dit in de meeste gevallen slechts 2 maal per dag zal veranderen wordt hier 0,083333 per uur gebruikt wat neerkomt op eens per 12 uur. Eens de sensor gecontroleerd is op een verandering en in het geval van een verandering is doorgestuurd, gaat de module terug in deep sleep voor 5 minuten tot de volgende controle. Wanneer de bovenstaande cyclus wordt gebruikt om het verbruik in mAh te berekenen wordt Table 3: Verbruik buitenlicht sensor bekomen met een totaal verbruik van 0,168979296.

4.4 Levensduur op batterij

Wanneer de hierboven berekende verbruiken uitgerekend worden op de realistische energievoorraad van de batterijen zoals eerder besproken werd kan Table 4: Levensduur op batterij worden opgesteld. Hierbij kan geconcludeerd worden dat de modules van de binnen- en buitenlichtsensor lang zullen meegaan op de batterij met beide een levensduur van meer dan een jaar. Echter is dit niet het geval voor de hub door het verplicht luisteren naar de andere modules indien er een verandering is van lichtwaarden alsook het versturen van het alarmsignaal en updates over wifi. Deze module heeft dus slechts een levensduur van ongeveer een maand lang.

PIR sensor + hub				
	Stroom (mA)	duur (s)	#/h	mAh
PIR standby (startup)	0,006	25	12	0,0005
PIR active (detecting movement)	0,106	10	12	0,003533
Deep sleep	0,0001	3592	1	9,98E-05
BLE listen	80	5	12	1,333333
wifi send	260	5	1	0,361111
esp on (modem sleep)	31	10	12	1,033333

Table 1: Verbruik PIR sensor en hub

binnenlicht sensor				
	Stroom (mA)	duur (s)	#/h	mah
LDR	0,33	1	12	0,0011
BLE	130	5	0,5	0,090278
Deep sleep	0,0001	3588	1	9,97E-05
esp on (modem sleep)	31	1	12	0,103333

Table 2: Verbruik binnenlichtsensor

buitenlichtsensor				
	Stroom (mA)	duur (s)	#/h	mah
VELM	15,15	1	12	0,0505
BLE	130	5	0,083333	0,015046
Deep sleep	0,0001	3588	1	9,97E-05
esp on (modem sleep)	31	1	12	0,103333

Table 3: Verbruik buitenlichtsensor

	Batterij Realistisch (mAh)	Verbruik/u (mAh)	Levensduur (u)	Levensduur (d)	Levensduur (j)
buitenlicht	2000	0,168979296	11835,76949	493,1570622	1,351115239
binnenlicht	2000	0,194810778	10266,37244	427,7655183	1,171960324
PIR+HUB	2000	2,731910889	732,0883006	30,50367919	0,083571724

Table 4: Levensduur op batterij

5 Bespreking van de code

De code kan teruggevonden worden in onze GitHub repo:
<https://github.com/VdwMichiel/TeamSmartlights>

5.1 Buitenlichtsensor

Voor de buitenlichtsensor wordt gebruik gemaakt van de VEML7700. Hiervoor is de library *DFRobot_VEML7700* nodig. Deze wordt dus via een include toegevoegd aan het project. Om zo energiezuinig mogelijk te werken, wordt de VEML7700 gevoed via de ESP32 zelf. Hiervoor wordt gebruik gemaakt van GPIO 17. De gebruikte VEML7700 wordt aangemaakt onder de naam *als* via *DFRobot_VEML7700*. Binnen de setup wordt deze sensor gestart. Tijdens de setup wordt de initiële gemeten lichtsterkte gemeten en opgeslagen. Deze waarde wordt bepaald als het gemiddelde van 10 metingen te nemen om zo het effect van foutieve metingen tegen te gaan.

Binnen de loop wordt er eerst en vooral voor gezorgd dat de VEML7700 gevoed wordt door de bijhorende GPIO pin hoog te zetten. Wanneer dit gebeurt is wordt de ESP32 enkele seconden in deep sleep mode geplaatst tot de VEML7700 opgestart is om energiezuinig te werk te gaan. Wanneer deze opgestart is en de ESP32 uit die deep sleep is, wordt opnieuw het gemiddelde van 10 metingen genomen om de lichtsterkte te bepalen.

Om uiteindelijk te kunnen nagaan als er een alarm gestuurd moet worden, wordt gebruik gemaakt van een functie *lux_interval*. Dit is een functie die bepaalt als de lichtsterkte al dan niet voldoende is door gebruik te maken van twee intervallen en terug te geven in welk interval de opgemeten lichtsterkte zich bevindt. Binnen de loop wordt vervolgens nagegaan als de gemeten lichtsterkte in hetzelfde interval ligt als de vorige gemeten lichtsterkte. Wanneer deze verschillen van elkaar en de meest recent gemeten lichtsterkte in het tweede, dus hoogste, interval ligt, moet er een bericht verstuurd worden dat er een alarm moet afgaan zodat duidelijk wordt dat de lichten uitgeschakeld moeten worden. Uiteindelijk wordt de meest recent gemeten lichtsterkte opgeslagen om bij de volgende stap opnieuw te kunnen vergelijken.

Na al dit wordt de voeding naar de VEML7700 weer uitgeschakeld en de ESP32 voor 5 minuten in deep sleep mode geplaatst. De loop wordt dus om de 5 minuten herhaald. Dit alles gebeurt om zo energiezuinig mogelijk te werk te gaan.

5.2 Binnenlichtsensor

Om te meten hoe lang de verlichting aangeschakeld staat, wordt gebruik gemaakt van een ldr. Deze wordt via een analoge GPIO aan de ESP32, namelijk GPIO 36. Binnen de loop wordt eerst en vooral de waarde afkomstig van de ldr opgeslagen.

Wanneer deze waarde minstens 1000 hoger is dan een vorige waarde, waaruit dus blijkt dat de verlichting aangeschakeld is, wordt het tijdstip bepaald en de waarde opgeslagen onder *last_value*. Als initiële waarde van deze variabele wordt een waarde genomen waarvoor de verlichting uitgeschakeld is. Wanneer deze waarde minstens 1000 lager is dan de vorige waarde, waaruit dus blijkt dat de verlichting uitgeschakeld is, wordt opnieuw het tijdstip bepaald. Uiteindelijk wordt de tijd dat de verlichting aangeschakeld stond bepaald door het verschil tussen deze tijdstippen. Om de totale tijd te bekomen wordt deze waarde opgeteld bij de totale tijd die bij elke stap bijgehouden wordt. Deze tijd kan dan omgezet worden naar uren en minuten en op een display getoond worden of doorgestuurd worden naar een centrale eenheid.

Ook hier wordt opnieuw gebruik gemaakt van de deep sleep mode die beschikbaar is voor de ESP32 om zo energiezuinig mogelijk te werk te gaan. Op het einde van de loop wordt de ESP32 5 minuten in deep sleep geplaatst. Er zal dus om de 5 minuten nagegaan worden als de verlichting aan of uitgeschakeld is. Hierdoor wordt er een resultaat verkregen dat niet volledig accuraat is maar wel een goed inzicht geeft over het energieverbruik.