

In de eerste les van Unit 3 gaan we de ingebouwde lichtsensor van de Ti-Innovator HUB gebruiken en introduceren we de het **Output**(commando uit Ti-Basic om getallen weer te geven.

Objectives:

- Uitlezen van de BRIGHTNESS sensor
- Introductie van de **While** loop
- Gebruik van het **Output**(commando
- Introductie van het commando **toString**(en tekstsamenvoegingen.

In de vorige lessen hebben we alleen opdrachten naar de TI-Innovator™ Hub gestuurd die de ingebouwde apparaten zoals LIGHT, SOUND en COLOR aanstuurden.

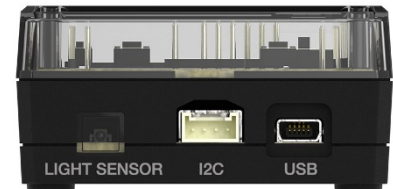
In deze unit gaan we werken met de ingebouwde lichtsensor, en gebruiken we de waarde in het programma om een lichtmeter te bouwen. De lichtsensor geeft een waarde tussen 0 en 100 (met decimalen).

Om de waarde van de lichtsensor te krijgen, hebben we twee commando's nodig:

- **Send ("READ BRIGHTNESS")**
- **Get(B)**

Het programma:

1. Maak een nieuw programma en noem het BRIGHT1.
2. Voeg de commando's **WisHome** en **Disp** toe met de titel van het programma zoals in het voorbeeld.
3. Press **[prgm]** en ga met de pijltjes naar het **HUB** menu.
4. Selecteer **2: Send("READ...** en dan **1: BRIGHTNESS.**
5. Druk weer op **[prgm]** en dan het **HUB** menu.
6. Kies **5:Get(**



```
NORM DRIJF AUTO REEEL RAD MN
PROGRAM: BRIGHT1
:WisHome
:Disp "LIGHT SENSOR"
:
:Send("READ BRIGHTNESS ")
:Get(B)
:
:
:
```

Hoe werkt het:

- **READ BRIGHTNESS** geeft de opdracht aan de TI-Innovator Hub om de helderheid of wel de lichtsterkte (brightness) te meten. Deze waarde wordt in de innovator in een buffer opgeslagen.
- **Get(B)** zorgt er voor dat de waarde die in de buffer is opgeslagen wordt doorgegeven naar de TI-84 Plus CE en in de variabele **B** wordt gestopt. B kan worden vervangen door alle numerieke variabelen van de rekenmachine: A...Z en θ (theta).

Tip: Een 'buffer' is een geheugenplaats op de TI-Innovator Hub waar tijdelijk waarden worden opgeslagen. Deze geheugenplaats wordt bij elk READ commando aangepast. Daarom moet na elk READ commando een GET commando komen om de waarde door te geven aan de rekenmachine. Het is mogelijk om een reeks metingen uit te voeren met TI-Innovator Hub en deze in een lijst op te slaan en deze voor latere analyse te gebruiken. Dat gaat echter te ver voor deze introductie.

While Loop:

De **While...End** loop (**[prgm]** CTL menu) wordt gebruikt om een hoeveelheid code te herhalen zolang (while !) de conditie waar is. Een *conditie* is een logische expressie die de waarde *waar* (true) of *niet waar* (false) geeft. De relationele operatoren en logische operatoren kun je op de rekenmachine vinden in het **test** menu van de calculator (**2nd math**).

De relationele operatoren zijn =, ≠, <, >, ≤, en ≥.

De logische operatoren zijn **and**, **or**, **not**, en **xor**.

Deze operatoren kunnen gezamenlijk gebruikt worden om condities te maken zoals : **x>0** and **y>0**.

We gaan een eenvoudige **While** loop gebruiken die stopt als de lichtsterkte (BRIGHTNESS) lager is dan 1. Om het programma te beëindigen moet je dan alleen de lichtsensor op de TI-Innovator Hub te bedekken met je hand.

Een andere manier om een programma te onderbreken is drukken op de **ON** knop. Er komt dan een foutmelding: **FOUT: AFBREKEN** aan de bovenkant van het scherm. Je krijgt de keus "1: Afsluiten" naar het basisscherm of "2: Ga Naar" naar de programma-editor op de plaats waar het programma was toen je op **ON** drukte. Dit is een gebruikelijke manier om weer verder te gaan met bewerken van je programma.



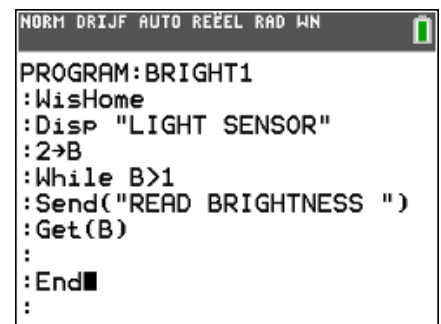
Een While Loop toevoegen:

7. Voeg voor het **Send** commando in je programma de volgende regels toe:

- **2→B**
- **While B>1** (gebruik het [test] menu om > te selecteren)

Deze regels starten de loop. Zolang de conditie (voorwaarde) **B>1** waar is, blijft de loop de lichtsensor uitlezen. Zodra de voorwaarde *onwaar* is (b.v. als er geen licht meer op de sensor valt omdat je hand het bedekt) wordt de loop beëindigd.

8. Het commando **End** van de **While** loop moet ook ingevoerd worden. Voeg na het **Get** commando **End** toe voor het einde van de **While** loop.



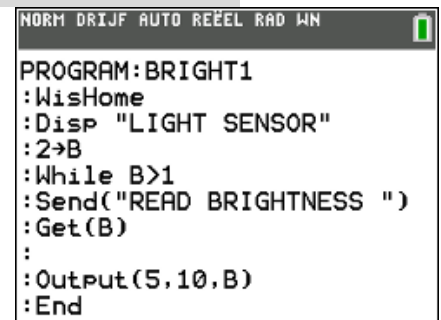
Tip: Het is aan te raden dat een programma altijd een mogelijkheid heeft om uit een loop te komen. Bedenk dat elke 'control structure' zijn eigen **End** heeft. In grotere programma's zullen veel verschillende **End** commando's zijn. De processor weet welk **End** hoort bij welke structuur/loop. De programmeur moet de juiste code ontwerpen.

9. Voeg het **Output** commando toe na het **Get** commando maar voor het **End** van de loop (zie voorbeeld). Het **Output** commando vind je via [prgm](#) en dan met de pijltjes naar **I/O**, en selecteer **6: Output**.

- Het **Output** commando geeft je veel controle over de plaats op het basisscherm waar iets moet worden getoond. De structuur van het commando luidt: **Output(<line#>, <column#>, <string of variabele>)**

Voorbeeld:

- **Output(3,7,"HELLO")** plaatst de letter "H" op lijn 3, kolom 7 van het basisscherm. De rest van het woord volgt de letter "H".
- **Output(5,10,B)** plaatst de waarde van de variabele B op lijn 5 en kolom 10 van het basisscherm. De rest van de getallen volgt.



10 Minutes of Code

TI-84 PLUS CE MET DE TI-INNOVATOR™ HUB

10. Verlaat de editor en voer het programma uit terwijl de TI-Innovator Hub aangesloten is.
- Je zult de titel zien verschijnen aan de bovenkant van het scherm en in de waarde in het midden van het scherm. Deze waarde zal variëren afhankelijk van de lichtsterkte die de sensor meet. Helaas is het mogelijk dat de getallen die je ziet niet correct zijn.

Het **Output**(commando haalt geen decimalen weg als er een korter nummer wordt getoond nadat er een langer nummer wordt getoond. Bijvoorbeeld als eerst de waarde 1.23456 getoond wordt en daarna 55, dan zie je de waarde 5523456. De laatste aanpassing aan het programma is een truc om dit probleem op te lossen.

Om dit probleem van overblijvende digits op te lossen moeten we de waarde van variabele B converteren naar tekst die we aanvullen met spaties aan het einde om de vorige waarde totaal te overschrijven.

Het commando **naarString**(kun je vinden in het **programma I/O** menu.

Het uiteindelijke commando wordt:

```
Output(5,10,naarString(B)+"    ")
```

Er zijn ongeveer 10 spaties tussen de aanhalingstekens.

Je zult nu zien dat sommige waarden korter zijn dan anderen.

Tekstamenvoegingen:

Het "+" teken dat wordt gebruikt in het **Output**(commando wordt *niet* gebruikt om op te tellen: het wordt gebruikt om twee stukken tekst aan elkaar te plakken. De spaties tussen de aanhalingstekens worden aan het einde van tekstweergave van variabele B geplakt.

Tip: Tekstmanipulatie is niet noodzakelijk voor deze lessen, maar het zal duidelijk zijn uit het voorbeeld dat het grote mogelijkheden schept voor programmeren.

UNIT 3: OEFENBLAD 1

DOCENTENHANDLEIDING

