

CE

CONRAD

Alle Versuche im Überblick

Internet of Things Adventskalender 2017.....	3	12e dag.....	27
Broncodes en bijkomende informatie.....	3	Vandaag in de adventskalender.....	27
Achtergrondinformatie m.b.t. de onderdelen.....	3	Looplichtsnelheid via app instellen.....	27
LED's.....	3	De sketch.....	27
Weerstand en hun kleurcodes.....	4	De app.....	27
1e dag.....	5	13e dag.....	28
Vandaag in de adventskalender.....	5	Vandaag in de adventskalender.....	28
IoT-board configureren.....	5	RGB via slider in app aanpassen.....	28
Stap 1: Installatie van de driver voor het IoT-board.....	5	De sketch.....	28
Stap 2: Installatie van de Arduino IDE.....	5	De app.....	29
Update van de firmware op het IoT-board.....	6	14e dag.....	30
IoT-board testen.....	7	Vandaag in de adventskalender.....	30
On-Board-LED doen knipperen.....	7	Boetseerkelecontact.....	30
2e dag.....	9	Zo werken sensorcontacten.....	30
Vandaag in de adventskalender.....	9	De sketch.....	30
Analoge waarden meten.....	9	De app.....	31
Het programma.....	9	15e dag.....	32
Zo werkt het programma.....	9	Vandaag in de adventskalender.....	32
3e dag.....	10	Verschillende boetseerkelecontacten.....	32
Vandaag in de adventskalender.....	10	De sketch.....	32
Knipperlicht.....	10	De app.....	32
Het programma.....	10	16e dag.....	33
Zo werkt het programma.....	10	Vandaag in de adventskalender.....	33
4e dag.....	11	Knipper-LED via app sturen.....	33
Vandaag in de adventskalender.....	11	De sketch.....	33
Knipperlicht (wisselend).....	11	De app.....	33
Het programma.....	11	17e dag.....	34
Zo werkt het programma.....	11	Vandaag in de adventskalender.....	34
5e dag.....	12	Weergave van de weerstandswaarde.....	34
Vandaag in de adventskalender.....	12	De sketch.....	34
Verkeerslicht.....	12	De app.....	34
Snap! installeren en IoT-board voorbereiden.....	12	18e dag.....	35
Programma in Snap! omzetten.....	12	Vandaag in de adventskalender.....	35
6e dag.....	13	RGB-looplicht.....	35
Vandaag in de adventskalender.....	13	De sketch.....	35
Verbinding met het IoT-board.....	13	De app.....	35
App voor de besturing installeren.....	13	19e dag.....	36
Het programma.....	13	Vandaag in de adventskalender.....	36
7e dag.....	15	app voor selectie van hardware-apps.....	36
Vandaag in de adventskalender.....	15	De sketch.....	36
Stuurbaar looplicht.....	15	De app.....	36
Het programma.....	15	20e dag.....	37
Zo werkt het programma.....	15	Vandaag in de adventskalender.....	37
8e dag.....	16	Warmtesensor in de app.....	37
Vandaag in de adventskalender.....	16	De sketch.....	37
Tonen via app uitvoeren.....	16	De app.....	37
Ontwikkelingsomgeving voor de apps.....	16	21e dag.....	38
De eerste app met AI2.....	17	Vandaag in de adventskalender.....	38
Piezo met app besturen.....	20	Helderheid en duisternis in de app meten.....	38
Werkwijze van de app.....	20	De sketch.....	38
App testen.....	22	De app.....	38
9e dag.....	23	22e dag.....	39
Vandaag in de adventskalender.....	23	Vandaag in de adventskalender.....	39
RGB-LED's.....	23	Vochtigheidsmeter.....	39
Kleur van een RGB-LED via app wijzigen.....	23	De sketch.....	39
De sketch voor het IoT-board.....	23	De app.....	39
De app.....	24	23e dag.....	40
10e dag.....	25	Vandaag in de adventskalender.....	40
Vandaag in de adventskalender.....	25	Codebreker.....	40
Toetsdruk weergeven.....	25	De sketch.....	40
De sketch.....	25	De app.....	40
Reactie van het IoT-board weergeven.....	25	24e dag.....	42
11e dag.....	26	Vandaag in de adventskalender.....	42
Vandaag in de adventskalender.....	26	Reactiespel.....	42
LED-echo via app.....	26	De sketch.....	42
De sketch.....	26	De app.....	42
De app.....	26		

Internet of Things Adventskalender 2017

Volgens Cisco zullen er tegen 2020 meer dan 50 miljard apparaten worden gebruikt die met het internet zijn verbonden. Intel is nog optimistischer en spreekt van 200 miljard apparaten¹. Elk van deze apparaten – ook „Ding“ genaamd – beschikt over een uniek adres en communiceert via het internet of via een andere interface, bv. Bluetooth, met de buitenwereld. Van het koffiezetapparaat tot de koelkast, van de auto tot de trein, van de productiemachine tot en met een armband, alles is programmeerbaar en kan met andere dingen communiceren. Dit thema is meer dan een hype, het wordt de standaard van de techniek en daarom moeten wij er ons allemaal mee bezig houden. Gebruik de adventsperiode en stap mee in het Internet der Dingen, ook Internet of Things (IoT) genaamd. In 24 experimenten leert u meer over het thema en programmeert u uw eigen ding. Veel plezier ermee!

Broncodes en bijkomende informatie

U ontvangt in de volgende 24 uur veel nieuwe informatie over het thema IoT (Internet of Things) en zult deze in spannende projecten omzetten. Opdat u de deels echt grote programma's niet moet aftippen, geven wij u de volledige broncodes en evt. bijkomende informatie om te downloaden. Ga hiervoor naar de website <http://www.buch.cd> en voer de code **15007-3** in. Daar vindt u een archief om te downloaden. In het archief bevindt zich voor elke dag een eigen directory. Voor meer gedetailleerde informatie leest u ook het bestand **Leesmij.pdf** in het archief.

De programma's vanaf dag 8 zijn deels heel groot. Daarom worden de programma's ook niet volledig afgedrukt. Alleen de bestanddelen die u absoluut nodig hebt om het project te begrijpen en om te zetten, worden in het handboek beschreven. Bij projecten die met een eigen aangemaakte smartphone app worden gestuurd, is er eveneens een volledige download beschikbaar. Als u wijzigingen aan het programma uitvoert, dan kijkt u altijd eerst naar het volledige programma, maakt u een kopie daarvan en wijzigt u dan de kopie. Als er iets mislukt, dan vindt u op <http://www.buch.cd> de startbestanden.

Updates in het downloadbereik

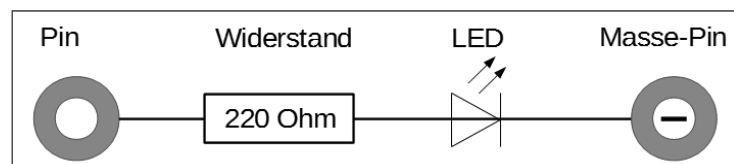
De adventskalender werd lang voor de adventsperiode opgemaakt en berust op de programmaversies die op dat moment beschikbaar waren. Als er tegen de advent grote wijzigingen optreden, vindt u een update van het betreffende project in het downloadbereik.

Achtergrondinformatie m.b.t. de onderdelen

In elk vak bevindt zich een onderdeel. Op deze plaats verneemt u de belangrijkste informatie in verband met de onderdelen.

LED's

LED's (lichtdioden) branden wanneer er stroom in de doorstroomrichting doorstroomt. LED's worden in schakelingen met een pijlvormig driehoeksymbool weergegeven dat de stroomrichting van de pluspool naar de minpool of massaleiding aangeeft. Een LED laat in de doorstroomrichting bijna een willekeurige hoeveelheid stroom door, ze heeft slechts een zeer kleine weerstand. Om de debietstroom te begrenzen en zo het doorbranden van de LED te voorkomen, moet gewoonlijk tussen de gebruikte aansluitpin en de anode van de LED of tussen de kathode en massapin een voorweerstand van 220 Ohm worden ingebouwd. Deze voorweerstand beschermt ook de uitgang van het inbegrepen IoT-board tegen te hoge stroomsterktes. De LED's in de adventskalender hebben de voorweerstand reeds ingebouwd en kunnen daarom rechtstreeks aan de pins van het IoT-board worden aangesloten.



Schakelplan van een LED met voorweerstand

¹ Bron: <https://www.fool.com/investing/general/2016/01/18/internet-of-things-in-2016-6-stats-everyone-should.aspx>

LED in welke richting aansluiten?

Beide aansluitkabels van een LED hebben een verschillende lengte. De langere is de pluspool, de anode, de kortere de kathode. Makkelijk op te merken: Het plusteken heeft een streep meer dan het minteken en dat maakt de kabel een beetje langer. Bovendien zijn de meeste LED's langs de minzijde afgevlakt, zoals een minteken. Makkelijk op te merken: Kathode = kort = kant.

Weerstanden en hun kleurcodes

Weerstanden worden bij gevoelige elektronische onderdelen en als voorweerstand voor LED's gebruikt om de stroom te begrenzen. De maateenheid voor weerstanden is Ohm. 1.000 Ohm komt overeen met een kiloohm, afgekort kOhm. 1.000 kOhm komt overeen met een megaohm, afgekort MOhm. Vaak wordt voor de eenheid Ohm het omegateken Ω gebruikt.

Kleur	Weerstandswaarde in Ohm			
	1e ring (tientallen)	2e ring (eenheden)	3e ring (multiplicator)	4e ring (tolerantie)
Zilver			$10^{-2} = 0,01$	$\pm 10 \%$
Goud			$10^{-1} = 0,1$	$\pm 5 \%$
Zwart		0	$10^0 = 1$	
Bruin	1	1	$10^1 = 10$	$\pm 1 \%$
Rood	2	2	$10^2 = 100$	$\pm 2 \%$
Oranje	3	3	$10^3 = 1.000$	
Geel	4	4	$10^4 = 10.000$	
Groen	5	5	$10^5 = 100.000$	$\pm 0,5 \%$
Blauw	6	6	$10^6 = 1.000.000$	$\pm 0,25 \%$
Paars	7	7	$10^7 = 10.000.000$	$\pm 0,1 \%$
Grijs	8	8	$10^8 = 100.000.000$	$\pm 0,05 \%$
Wit	9	9	$10^9 = 1.000.000.000$	

De gekleurde ringen op de weerstanden geven de weerstandswaarde aan. Deze zijn duidelijk makkelijker te herkennen als de ultrakleine cijfers die men alleen nog op hele oude weerstanden vindt en met een beetje oefening kunt u de waarden ook uit de kleurcodes snel „vertalen“.

De meeste weerstanden hebben vier zulke kleuringen. De eerste twee kleuringen staan voor de cijfers, de derde is een multiplicator en de vierde de tolerantie. Deze tolerantiering is meestal goud- of zilverkleurig - kleuren die op de eerste twee ringen niet voorkomen. Daardoor is de leesrichting uniek. De tolerantiewaarde zelf speelt in de digitale elektronica nauwelijks een rol. De tabel toont de betekenis van de gekleurde ringen op weerstanden.

Het is gelijk in welke richting een weerstand wordt ingebouwd. Bij LED's daarentegen speelt de inbouwrichting een belangrijke rol.

1e dag

Vandaag in de adventskalender

• 1 x IoT Bluething Board²

Vandaag leert u het board kennen waarmee u de projecten in de volgende 24 dagen zult omzetten. Als voorbereiding voor de volgende dagen zult u de driver voor de USB-verbinding installeren die Arduino IDE installeert en tot slot maakt u een eerste programma voor het board aan.

Chipset op het IoT-board

Het IoT-board komt met twee chipsets. Voor de uitvoering van de programma's bevindt zich een ATmega328P op het board. Deze microcontroller communiceert via de seriële interface met een HC-05. De HC-05 is verantwoordelijk voor de draadloze verbinding (Bluetooth). De module ondersteunt Bluetooth V2.0+EDR. Voor de app-communicatie is een smartphone met Android nodig.

IoT-board configureren

Om het IoT-board in gebruik te nemen hebt u een pc met Linux, Mac OS X of Windows en een micro-USB-kabel nodig. Deze aansluitkabel dient voor de stroomvoorzorging en de verbinding van het IoT-board met de pc om het te programmeren. U hoeft geen bijkomende dergelijke kabel te kopen aangezien u er vermoedelijk al een bezit - bijna alle moderne smartphones gebruiken dit stekkertype.

Juiste USB-poort aan de pc selecteren

Sluit de kabel indien mogelijk aan een USB-2.0-aansluiting van uw pc aan. Bij USB-3.0-aansluitingen kan dit tot verbindingsproblemen leiden. Een USB 3.0-aansluiting herkent u vaak aan een blauwe bus.

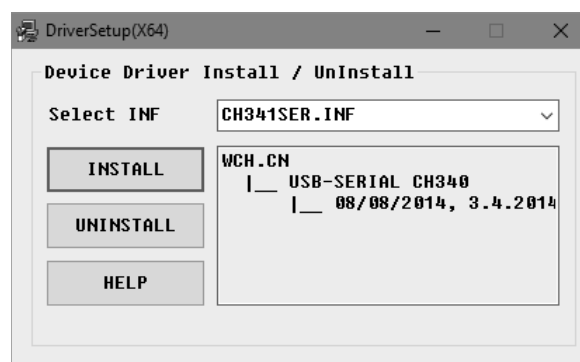
Voor u het board met de pc verbindt, zijn de volgende stappen nodig:

- Stap 1: Installatie van de driver voor het IoT-board
- Stap 2: Installatie van de Arduino IDE

Stap 1: Installatie van de driver voor het IoT-board

De USB-poort die zich op het IoT-board bevindt is met een CH340G-chipset verbonden. Om deze chipset voor de USB-verbinding te gebruiken moet u de passende driver voor uw besturingssysteem installeren. Voer hiervoor de volgende vier stappen uit:

- 1 Download de voorbeeldprogramma's en de apparaatdriver op <http://www.buch.cd>. Voer daar de code **15007-3** in en volg de instructies op het scherm.
- 2 Pak het ZIP-archief in een willekeurige map onder uw gebruikersmappen uit.
- 3 Sluit het IoT-board via de USB-kabel aan en start dan de installatie van de driver met het bestand CH341SER.EXE. Voor de installatie moet u eventueel een vraag van het Windows-gebruikersaccountbeheer bevestigen.
- 4 Klik in de installatiedialoog op **Install** en wacht tot een bevestiging verschijnt dat de driver geïnstalleerd werd.



Installatie van de driver van het apparaat

Stap 2: Installatie van de Arduino IDE

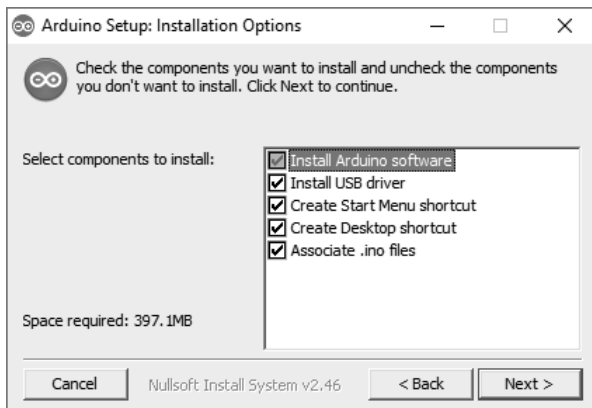
Het IoT-Board is compatibel met de Arduino Nano and kan met de Arduino IDE3 geprogrammeerd worden. In de Arduino IDE schrijft u programma's in de programmeertaal C en zet u ze dan onmiddellijk op het IoT-board over. Na de overdracht loopt het programma zonder verbinding met de pc en daarom kunt u daarna deze verbinding ook verbreken.

IoT-board uitschakelen

Het IoT-board heeft geen uit-schakelaar. U hoeft gewoon de USB-kabel van de pc of de adapter te ontkoppelen en het IoT-board schakelt zichzelf uit. Bij het volgende inschakelen start automatisch het laatst opgeslagen programma. Hetzelfde gebeurt wanneer men op de resetknop drukt.

² Hieronder verder als IoT-board genoemd.

³ Voor de projecten werd de Arduino IDE in versie 1.8.2. gebruikt. Als er bij het verschijnen van de adventskalender een nieuwere versie beschikbaar is, kunt u deze gebruiken. Als de wijzigingen markant zijn, vindt u in het downloadbereik de overeenkomstige instructies.



Installatie van de Arduino IDE

Download de Windows Installer voor de huidige versie van Arduino IDE op www.arduino.cc/en/Main/Software of gebruik gewoon het bestand `arduino-windows.exe` uit de downloads bij de adventskalender. Onder Windows 10 kunt u de Arduino IDE ook uit de Windows Store downloaden en installeren.

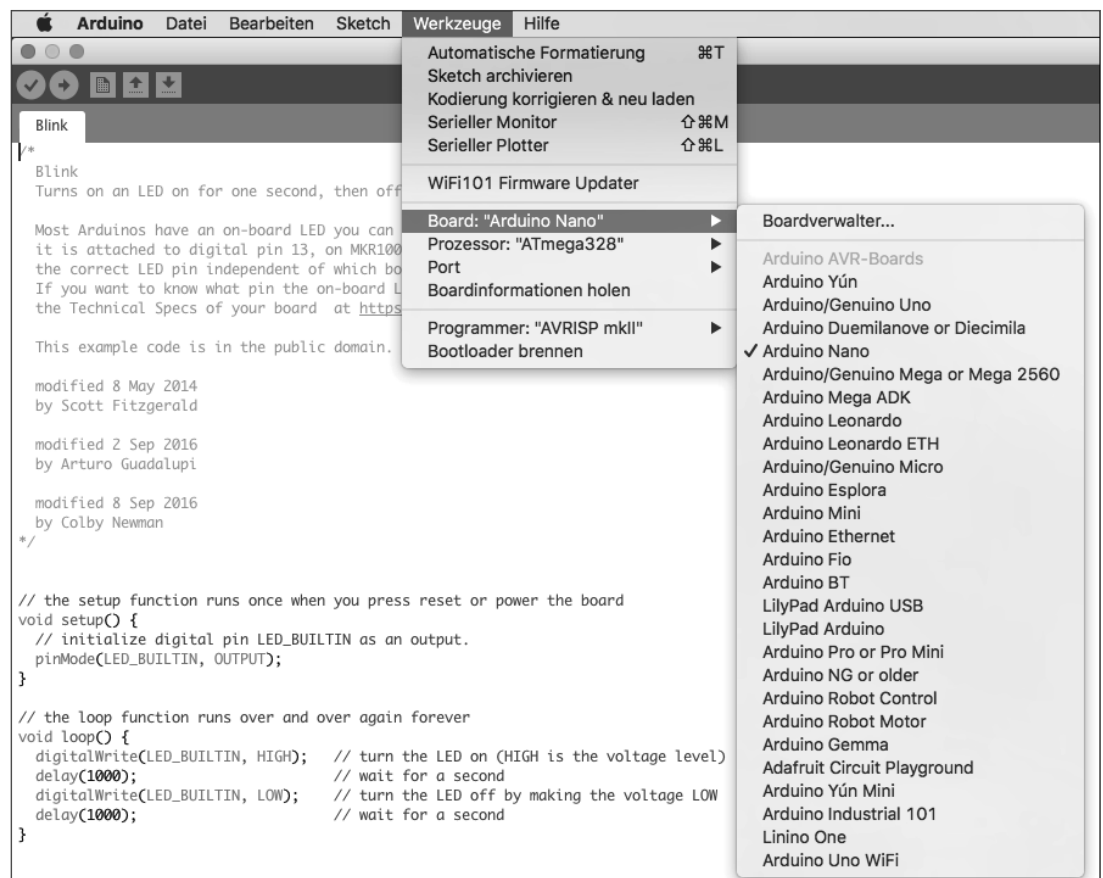
Let daarbij op dat in het dialoogveld **Installation Options** alle haakjes zijn aangevinkt. Naargelang de Windowsconfiguratie is een bevestiging van het gebruikersaccountbeheer nodig.

Update van de firmware op het IoT-board

Opdat u ook alle voorbeelden met uw IoT-board kunt uitvoeren, update u eerst de firmware van het board. Hiervoor opent u eerste de eerder geïnstalleerde Arduino IDE en kies in het menu van Arduino IDE **Tools /**

Port. Hier wordt in de meeste gevallen slechts een enkele seriële poort weergegeven. Plaats hier het haakje.

Kies vervolgens via het menupunt **Tools / Board** de **Arduino Nano** wanneer deze niet reeds automatisch werd herkend. Als processor moet **ATmega328** geselecteerd worden.



In de Arduino IDE kiest u het passende board.



Snelle start van de download van de firmware op het board

Na de start van Arduino IDE en de selectie van het board opent u nu via het menupunt **Bestand / Openen...** de firmware **Firmware_V1.2b.ino**. Deze bevindt zich in het downloadarchief in de directory **Firmware_V1.2b**. Voor u de firmware afspeelt, verzekert u dat de jumper op het IoT-board op **AT** staat. Laad via **Sketch / Uploaden** de firmware op het board op. Anders kunt u ook de icons in de Editor daarvoor gebruiken: Via de pijl naar rechts laadt u de firmware⁴ op het board.

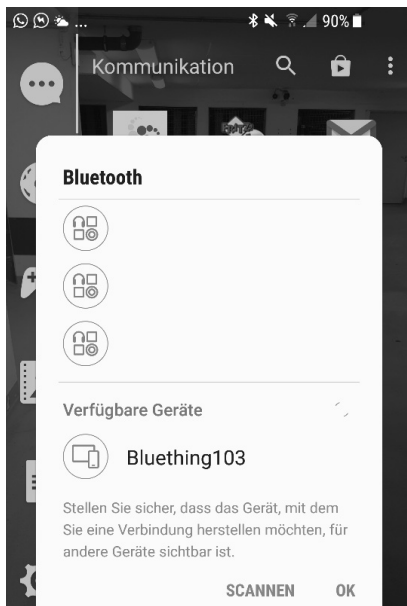
Na korte tijd moet in de statusregel van de editor het bericht **Uploaden voltooid** staan. Nu is het IoT-board geüpdatet.

⁴ De firmware is een normaal Arduino-programma. Een dergelijk Arduino-programma wordt als Sketch beschreven.

IoT-board testen

Na de uitgevoerde update van de firmware kunt u het board nu testen. Open daarvoor de seriële monitor van de Arduino IDE via **Tools / Seriële monitor**. Stel de gegevensoverdracht op **57600 Baud** in. Nu moet leesbare tekst in het uitvoervenster verschijnen. Als teken dat de configuratie voltooid is, verschijnt het bericht **Configuratie succesvol!**, gevolgd door informatie over het draadloos netwerk.

Nu kunt u met uw Android smartphone testen of u het draadloos netwerk ook ziet. Schakel hiervoor op de smartphone Bluetooth in en na korte tijd moet het draadloos netwerk verschijnen. Kies nu het draadloos netwerk (in het volgende screenshot **Bluething103**) en bevestig de selectie met **OK**.



Op uw systeem zal de naam van het IoT-board ook met Bluething beginnen maar evt. met een ander nummer eindigen. Het nummer wordt aan de hand van het MAC-adres van uw board individueel gegenereerd.

Voor de opbouw van de verbinding moet u een wachtwoord invoeren, in dit geval **1234**.

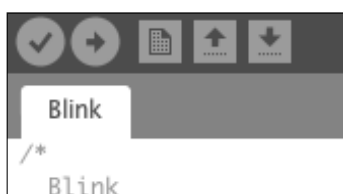
Nadat het juiste wachtwoord werd ingevoerd, verschijnt het netwerk onder **GEKOPPELDE APPARATEN**. Nu kan uw smartphone met het IoT-board communiceren.

Van zodra een apparaat met een smartphone is gekoppeld, is het mogelijk ook zonder nieuwe scan een verbinding aan te maken.

On-Board-LED doen knipperen

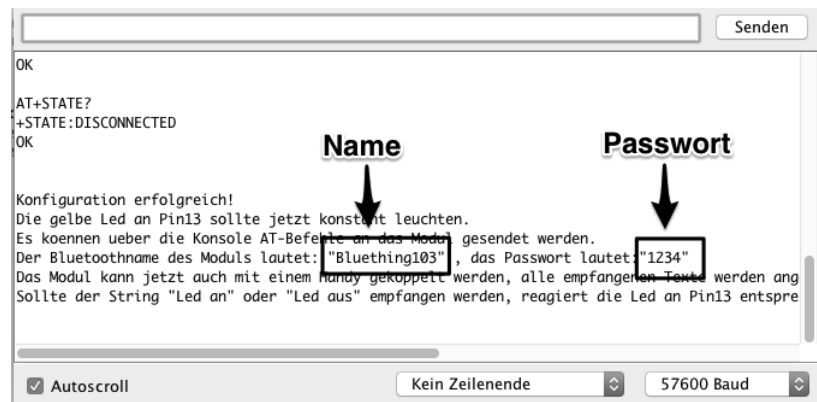
Nu moet het programmeren van het IoT-board worden getest. Als voorbeeld wordt de in de Arduino IDE geïntegreerde Blink-sketch gebruikt. Kies daarvoor in het menu **Bestand / Voorbeelden / 01.Basics / Knipperen**.

Klik nu linksboven met de pijl op het ronde symbool om het programma op het aangesloten IoT-board te uploaden, wat ook Flashen wordt genoemd.

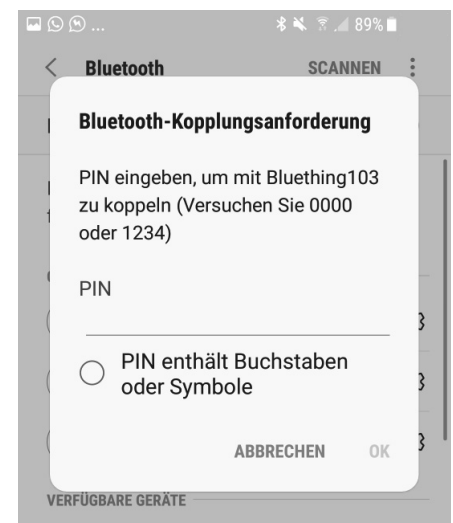


Met het haakje compileert u de bestanden, met de pijl naar rechts laadt u het programma op het IoT-board.

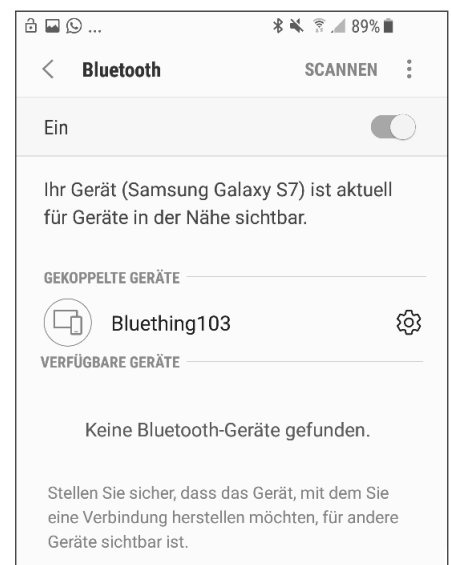
Nadat het uploaden is voltooid, knippert LED D2 (naast aansluiting D13) op het IoT-board. Nu is het board gebruiksgereed voor de volgende dag.



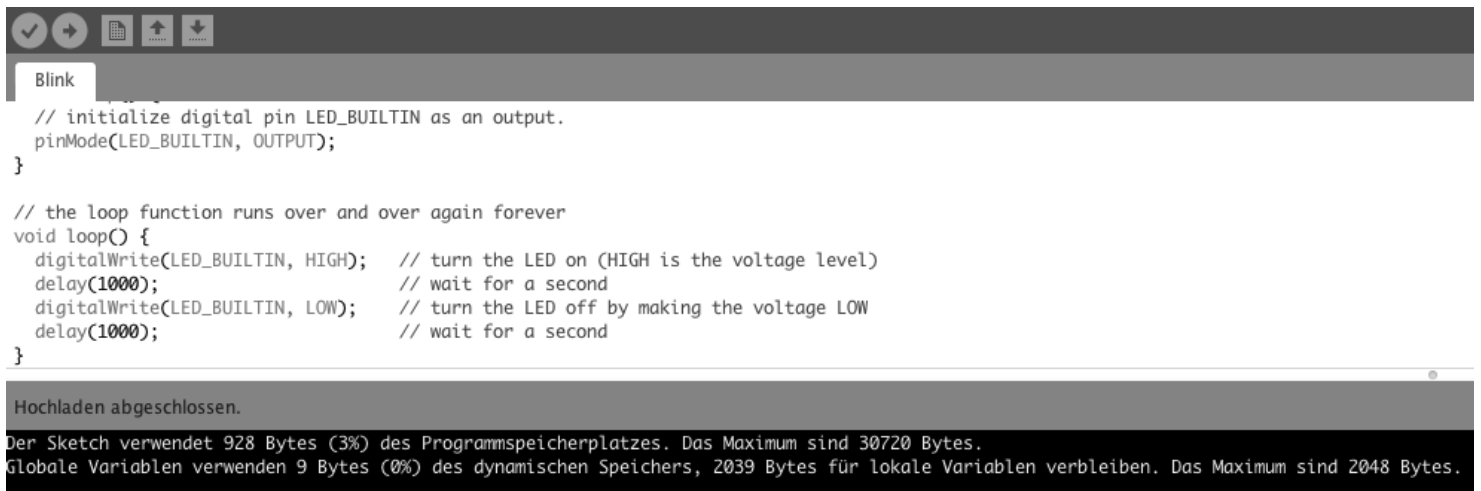
De naam van het netwerk in dit voorbeeld luidt **Bluething103** en het wachtwoord is **1234**.



Bij de verbindingsoptname met het IoT-board moet u een wachtwoord invoeren.



Op uw systeem zal de naam van het IoT-board ook met Bluething beginnen maar evt. met een ander nummer eindigen. Het nummer wordt aan de hand van het MAC-adres van uw board individueel gegenereerd.



```

Blink
// initialize digital pin LED_BUILTIN as an output.
pinMode(LED_BUILTIN, OUTPUT);
}

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

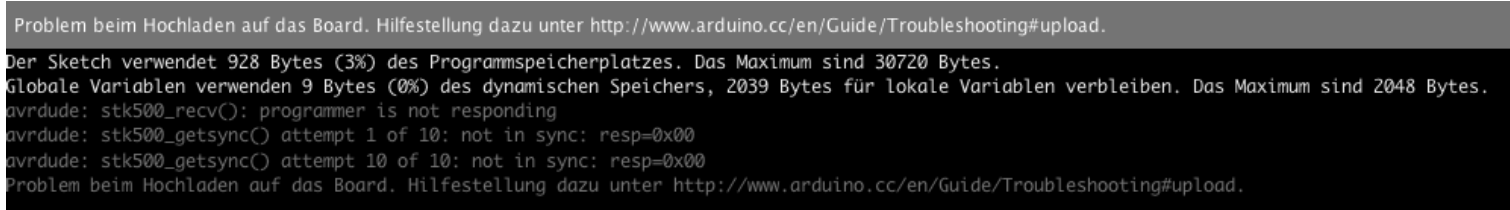
```

Hochladen abgeschlossen.

Der Sketch verwendet 928 Bytes (3%) des Programmspeicherplatzes. Das Maximum sind 30720 Bytes.
 Globale Variablen verwenden 9 Bytes (0%) des dynamischen Speichers, 2039 Bytes für lokale Variablen verbleiben. Das Maximum sind 2048 Bytes.

Onder het broncodevenster ziet u de uitgaven van de Arduino IDE bij het compileren en het uploaden.

Als de LED niet knippert, controleert u de foutmelding in de Arduino IDE.



Problem beim Hochladen auf das Board. Hilfestellung dazu unter <http://www.arduino.cc/en/Guide/Troubleshooting#upload>.

Der Sketch verwendet 928 Bytes (3%) des Programmspeicherplatzes. Das Maximum sind 30720 Bytes.
 Globale Variablen verwenden 9 Bytes (0%) des dynamischen Speichers, 2039 Bytes für lokale Variablen verbleiben. Das Maximum sind 2048 Bytes.

```

avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 1 of 10: not in sync: resp=0x00
avrdude: stk500_getsync() attempt 10 of 10: not in sync: resp=0x00

```

Problem beim Hochladen auf das Board. Hilfestellung dazu unter <http://www.arduino.cc/en/Guide/Troubleshooting#upload>.

Hier is de verbinding met het IoT-board mislukt. In dit geval werd de verkeerde poort geselecteerd, dit kan snel via het menupunt **Tools /Poort** worden verholpen.

IoT-board altijd in de AT-modus gebruiken

Op het IoT-board bevindt zich een jumper. Deze jumper moet voor alle projecten in deze kalender op **AT** staan.

2e dag

Vandaag in de adventskalender

- 1 x steekplatine (SYB 46)
- 1 x jumperkabel

Analoge waarden meten

Vandaag programmeert u een sketch in de Arduino IDE om waarden van een analoge ingang te analyseren. De waarden worden via tekstweergave en grafisch weergegeven.

Onderdelen: 1 x steekplatine, 1 x jumperkabel (mannelijk - mannelijk)

Het programma

Het programma voor vandaag draagt de naam `Tag02.ino` en bevindt zich in het downloadarchief in de directory `Dag02`.

```
int analogValue = 0;
int analogPin = A0;

void setup() {
  pinMode(analogPin, INPUT);
  Serial.begin(9600);
}
```

```
void loop() {
  analogValue = analogRead(analogPin);
  Serial.println(analogValue);
  delay(1000);
}
```

Zo werkt het programma

```
int analogPin = A0;
```

In het begin wordt het gebruikte pinnummer in de variabele `analogPin` vastgelegd. Alle programma's in de Arduino IDE bestaan tenminste uit de twee functies `setup` en `loop`:

```
void setup() {
  pinMode(analogPin, INPUT);
  Serial.begin(9600);
}
```

De functie `setup` loopt eenmaal tot aan de start en wordt meestal voor de configuratie gebruikt. De analoge pin wordt als invoer vastgelegd. Via `Serial.begin(9600)` wordt de seriële communicatie gestart om waarden op de seriële monitor en de seriële plotter weer te geven. De parameter bij deze functie is het Baudrate.

```
void loop() {
  ...
}
```

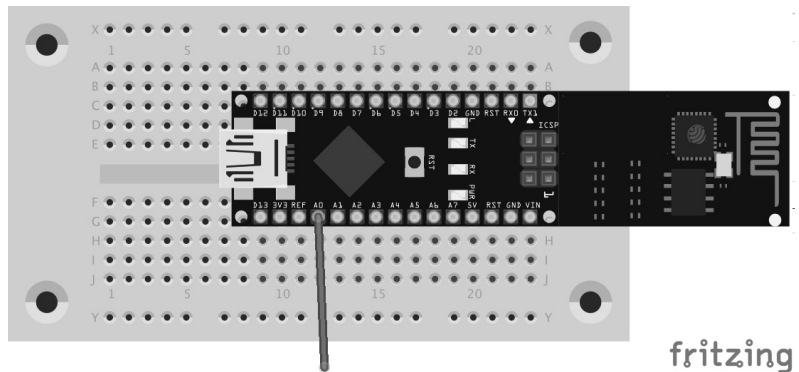
De functie `loop` wordt lang herhaald tot men de voedingsspanning ontkoppelt of op de resetknop drukt.

```
void loop() {
  analogValue = analogRead(analogPin);
  Serial.println(analogValue);
  delay(1000);
}
```

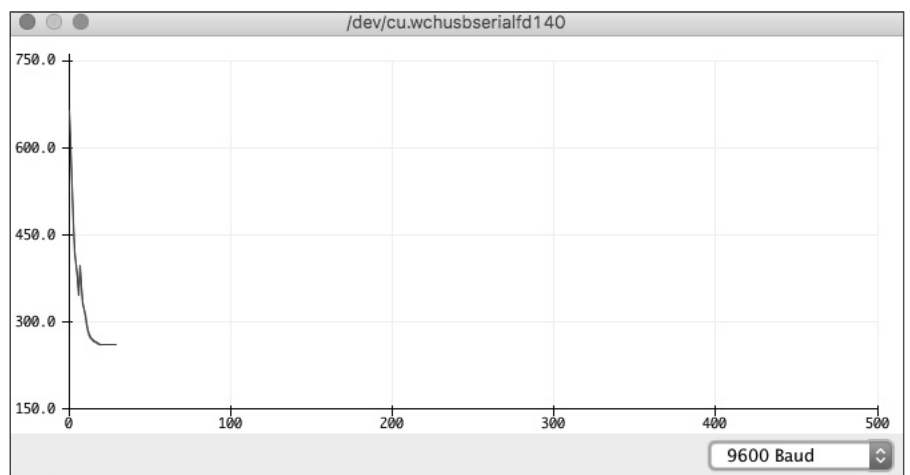
Via `analogRead` wordt de waarde van de analoge ingang in de variabele `analogValue` opgeslagen en via `Serial.println` uitgegeven. Door `delay(1000)` wacht het programma 1000 milliseconden.

Open nu de seriële monitor van de Arduino IDE via **Tools / Seriële monitor**. Zo ziet u de gemeten waarden. Die waarden kunt u ook grafisch laten weergeven. Sluit hiervoor de voorheen geopende seriële monitor en open de seriële plotter via **Tools / Seriële Plotter**.

2. dag



De jumperkabel dient als antenne. Probeer de metingen ook eens zonder deze kabel om te zien of de kabel voor een verschil in uw meetomgeving zorgt.



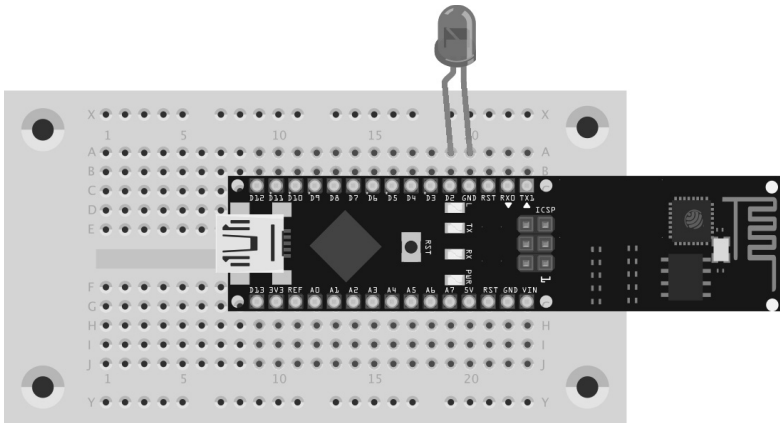
Het data-rate moet in het dropdown rechtsonder in de hoek van het venster correct zijn ingesteld, in dit geval **9600 Baud**.

3. dag

3e dag

Vandaag in de adventskalender

- 1 x LED rood met voorweerstand
- 1 x schakeldraad



fritzing

U hebt geen afzonderlijke weerstand nodig want de LED heeft reeds een geïntegreerde weerstand.

Knipperlicht

Vandaag laat u een LED met een frequentie van 2 Hz knipperen.

Onderdelen: • 1 x steekplatine, 1 x LED rood met voorweerstand

Het programma

Het programma voor vandaag draagt de naam Tag03.ino en bevindt zich in het downloadarchief in de directory Dag03.

```
const int ledPin = 2;
int ledState = LOW;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {

  if (ledState == LOW) ledState = HIGH;
  else ledState = LOW;
  digitalWrite(ledPin, ledState);
  delay(500);
}
```

Zo werkt het programma

```
if (ledState == LOW) ledState = HIGH;
else ledState = LOW;
```

In de variabele `ledState` wordt onthouden of de LED al of niet brandt. In het begin heeft de variabele de waarde `LOW`. Deze waarde wordt via `delay(500)` alle 500 ms omgeschakeld. Daardoor knippert de LED.

4e dag

Vandaag in de adventskalender

• 1 x LED geel met voorweerstand

Knipperlicht (wisselend)

Twee LED's knipperen afwisselend.

Onderdelen: 1 x steekplaatje, 1 x LED geel met voorweerstand, 1 x LED rood met voorweerstand

Het programma

Het programma voor vandaag draagt de naam `Tag04.ino` en bevindt zich in het downloadarchief in de directory `Dag04`.

```
const int ledPin1 = 10;
const int ledPin2 = 12;

int ledState = LOW;
int pin = ledPin1;

void setup() {
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  digitalWrite(ledPin1, LOW);
  digitalWrite(ledPin2, LOW);
}

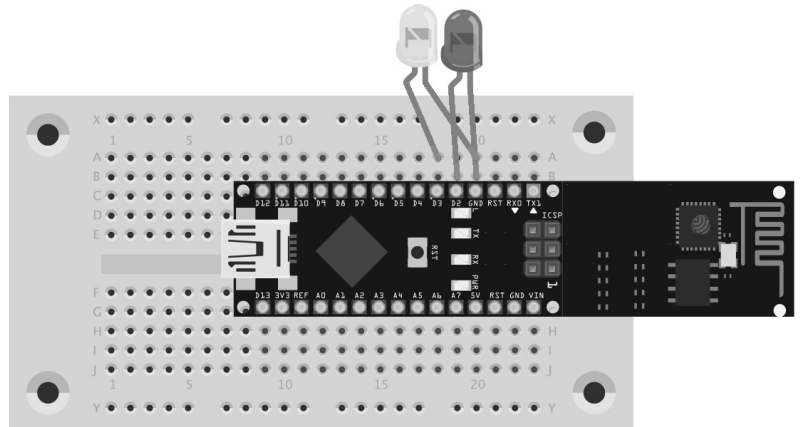
void loop() {
  if (pin == ledPin1) {
    pin = ledPin2;
    digitalWrite(ledPin1, LOW);
  } else {
    pin = ledPin1;
    digitalWrite(ledPin2, LOW);
  }
  digitalWrite(pin, HIGH);
  delay(500);
}
```

Zo werkt het programma

```
if (pin == ledPin1) {
  pin = ledPin2;
  digitalWrite(ledPin1, LOW);
} else {
  pin = ledPin1;
  digitalWrite(ledPin2, LOW);
}
```

In de tijdelijke variabele `pin` wordt opgeslagen welke LED nu brandt. Oorspronkelijk staat deze waarde op `ledPin1`. Via een if-vraag wordt de variabele op de andere pin gezet en de huidig brandende LED wordt uitgeschakeld. Het omschakelen gebeurt elke 500 ms.

4. dag



fritzing

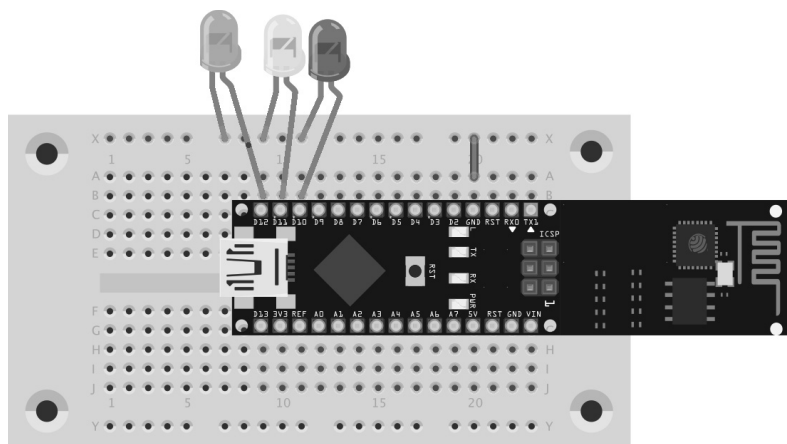
Beide kathoden (korte benen van de LED) moeten op GND worden geleid. Links is de gele LED, rechts de rode.

5. dag

5e dag

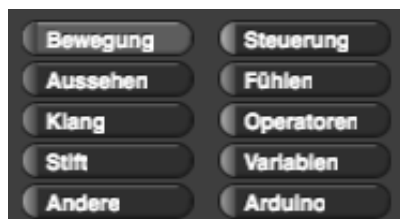
Vandaag in de adventskalender

• 1 x LED groen met voorweerstand



fritzing

Vanaf drie LED's wordt het een beetje knapper, daarom wordt GND extra aan de bovenste lijst gevoerd. LED's van links naar rechts: groen, geel en rood.



In de categorie **Arduino** vindt u de elementen voor de besturing van het IoT-board.

Om Snap4Arduino met uw bord te kunnen gebruiken, moet u een speciale Sketch op het bord afspelen, nl. Firmata. Sluit hiervoor het IoT-board aan uw pc aan en open Arduino IDE. Kies onder **Bestand / Voorbeelden / Firmata / Standaard Firmata** en laad de Sketch via een muisklik op de pijl naar rechts op het board. Nu kunt u het board met Snap4Arduino programmeren.

Arduino IDE en Snap! niet tegelijk gebruiken

Arduino IDE en Snap! kunnen niet tegelijk worden gebruikt. Sluit daarom het respectievelijk andere programma voor u met de door u gewenste omgeving aan de slag gaat.

Programma in Snap! omzetten

Een programma in Snap! wordt met grafische elementen samengesteld die in categorieën zijn ingedeeld. Het menu van de categorieën vindt u linksboven.

Trek nu uit **Besturing** het element met de groene kleur in het middelste werkbereik. Om de LED's in en uit te schakelen gebruikt u het element **Plaats digitale pin**. De waarden zet u via het omschakelbare element **waar** uit **Operatoren**. Opdat het programma ook niet stopt, hebt u het luselement uit **Besturing** nodig. Opdat de LED's ook niet onmiddellijk opnieuw omschakelen, bouwt u een pauze van een seconde in via **wacht** uit **Besturing**.

Voor u het programma kunt starten, moet u het IoT-board met Snap! verbinden. Hiervoor klikt u op het element **Met Arduino verbinden** in de categorie **Arduino**. Bij een klik op het symbool ziet u de beschikbare aansluitingen. Selecteer de eerste aansluiting. Nu klikt u op de groene pijl (rechtsboven) en het verkeerslicht begint reeds te werken.

Extern project in Snap! Openen

Het programma voor vandaag bevindt zich in de map **Dag05**. In de menulijst gaat u op het eerste symbool staan en kiest u **Importeren....** Navigeer naar de map **Dag05** en selecteer **dag05-snap.xml**. Nu is het project geopend en u kunt het gebruiken.



Het met Snap! gemaakte programma.

5 Anders kunt u ook de meest recente versie downloaden onder <http://snap4arduino.org/>. Het oppervlak kan lichtjes van de screenshots afwijken, de programma's moeten echter toch werken.

6e dag

Vandaag in de adventskalender

• 1 x LED blauw met voorweerstand

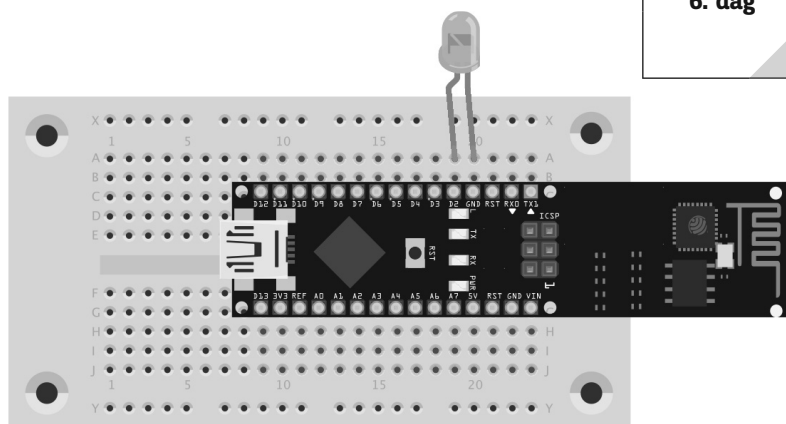
Verbinding met het IoT-board

Vandaag bouwt u met uw smartphone een verbinding met het IoT-board en schakelt u de blauwe LED via de smartphone in en opnieuw uit.

Onderdelen: • 1 x steekplatine, 1 x LED blauw met voorweerstand

App voor de besturing installeren

Vandaag gebruikt u nog geen eigen app, maar stuurt u het board via de gratis app **Serial Bluetooth Terminal** uit de app store Google Play.



De schakeling gelijkt op de schakeling van dag 3.

6. dag

fritzing

Het programma

Het programma voor vandaag bouwt op de firmware voor het IoT-board verder. De uit de firmware benodigde programma-onderdelen liggen in het bestand `Model.ino` in de directory `Model`.

Draadloze verbinding gereed

De firmware (en dus ook het model) is zo geprogrammeerd dat de oranje LED aan pin 13 permanent brandt van zodra de draadloze verbinding gereed is. Wacht daarom tot deze LED brandt voor u een verbinding met het IoT-board aanmaakt.

Kopieer het bestand `Model.ino` naar een nieuwe directory `Dag06` en geef het bestand een nieuwe naam `Dag06.ino`. Anders kunt u onmiddellijk het klare `Dag06.ino` uit de directory `Dag06` gebruiken. Nu opent u het bestand met Arduino IDE. Het model bevat reeds enkele functies. De constante voor de interne LED is reeds aanwezig; De Constante `LedPinBlue` voor de bijkomende LED definieert u rechtstreeks onder de interne LED:

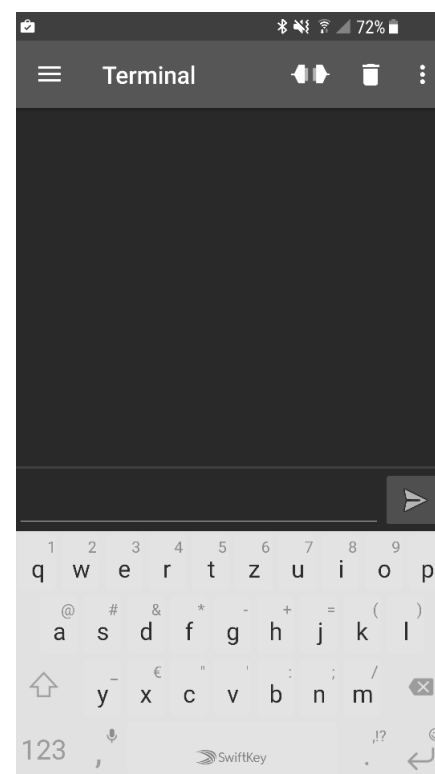
```
#define LedPin 13
#define LedPinBlue 2
```

Aan het einde van de methode `setup` wordt de LED ingeschakeld:

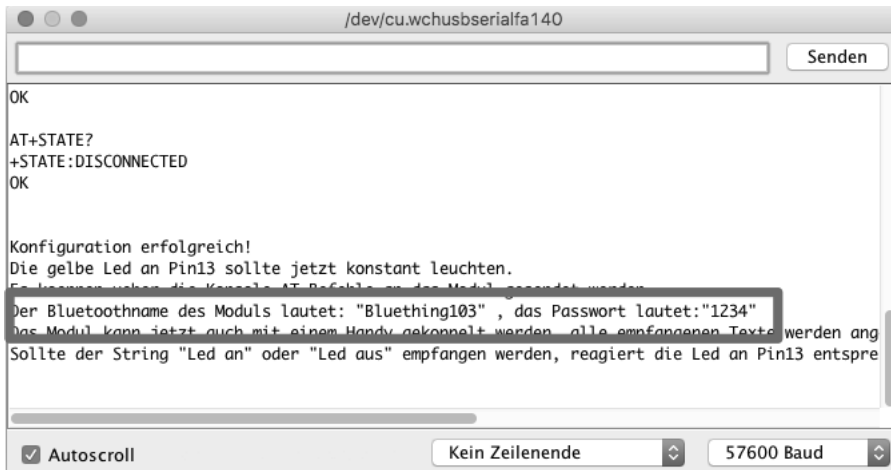
```
void setup() {
    ...
    digitalWrite(LedPinBlue, HIGH);
}
```

Voor het in- en uitschakelen van de geïntegreerde LED aan pin 13 is de code reeds aanwezig. Deze wordt nu als volgt aangevuld:

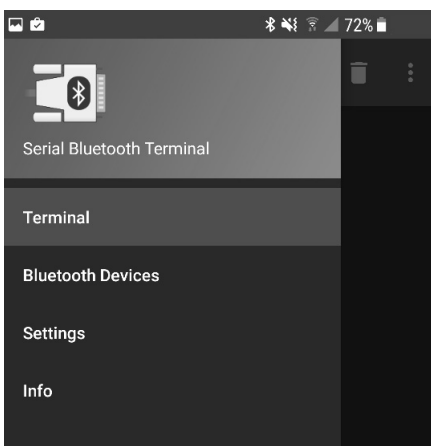
```
if (Text.startsWith("Led aan") || Text.startsWith("LED aan") || Text.startsWith("LED AAN")){
    digitalWrite(LedPin, HIGH);
    digitalWrite(LedPinBlue, HIGH);
}
if (Text.startsWith("Led uit") || Text.startsWith("LED uit") || Text.startsWith("LED UIT")){
    digitalWrite(LedPin, LOW);
    digitalWrite(LedPinBlue, LOW);
}
```



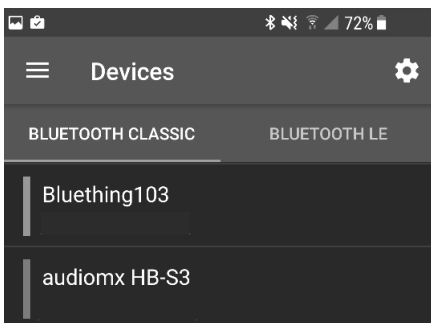
Na de eerste start verschijnt een leeg zwart venster.



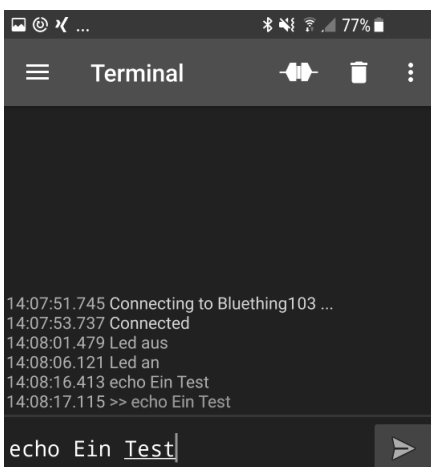
De naam van het netwerk luidt **Bluething103** en het wachtwoord is **1234**.



Via het centrale menu komt u in de instellingen van de app.



In de lijst van de Bluetooth-netwerken vindt u ook het netwerk van het IoT-board.



Via **echo** kunt u zien dat de app ook informatie van het IoT-board kan ontvangen.

Wanneer het board LED aan of Led aan of LED AAN ontvangt, worden via `digitalWrite(LedPin, HIGH)` en `digitalWrite(LedPinBlue, HIGH)` de interne LED en de blauwe LED ingeschakeld. Wanneer het IoT-board gegevens ontvangt, wordt de volgende lus uitgevoerd:

```
while(HC05.available() > 0){
  ...
  Text="";
}
```

Voeg nu voor de laatste regel een oproep van de nog te programmeren functie `hookRec(Text)` in:

```
while(HC05.available() > 0){
  ...
  hookRec(Text);
  Text="";
}
```

In deze functie geeft u nu de ontvangen tekst via de draadloze interface uit:

```
void hookRec(String text) {
  if (text.startsWith("echo") || text.startsWith("Echo")) {
    HC05.print(">> " + text);
  }
}
```

Nu kunt u het programma op het IoT-board afspelen. Schakel Bluetooth op uw smartphone in en selecteer het nieuw aangemaakte Bluetooth-netwerk. U ziet de overeenkomstige naam in de seriële monitor van de Arduino IDE. Het bijhorende venster opent u via **Tools / Seriële monitor**. Bij de verbinding moet u een wachtwoord invoeren; dit wachtwoord ziet u ook in de seriële monitor.

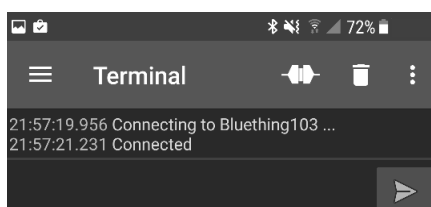
Nu start u de eerder geïnstalleerde Bluetooth-app. Ga naar start in het menu (drie boven elkaar liggende streepjes in de linker rand) van de app in selecteer het menupunt **Bluetooth Devices**.

Nu kiest u het overeenkomstige netwerk en gaat u naar de terminal door eenmaal op het menu-symbool te klikken en dan **Terminal** te selecteren.

In de terminal klikt u op het verbindingssymbool (links naast de prullenmand). Na korte tijd krijgt u de vraag om het wachtwoord in te voeren. Dit luidt **1234**. Bij de eerste keer lukt het opbouwen van de verbinding nog niet en daarom klikt u nogmaals op het symbool. Nu moet de verbinding zijn aangemaakt. In het terminalvenster verschijnt de melding **Connected**. Nu kunt u met het IoT-board communiceren.

Dubbele verbindingsofbouw nodig

Zolang u in de Bluetooth-app nog geen wachtwoord hebt ingevoerd, moet u tweemaal de verbinding opbouwen. Bij de eerste maal voert u het wachtwoord in en bij de tweede maal wordt de verbinding aangemaakt.



De terminal-app is nu gereed voor de communicatie.

Wanneer u nu **Led uit** invoert en op het pijlsymbool drukt, worden beide LED's uitgeschakeld. Door **Led aan** in te voeren, schakelt u de LED's opnieuw in. Elke invoer die met **echo** begint, wordt naar het board verzonden en dit zendt de invoer beginnend met twee pijlen terug. De tekst wordt dan door de terminal uitgegeven.

7e dag

Vandaag in de adventskalender

• 1 x potentiometer, 15 kOhm

Stuurbaar looplicht

Het project van vandaag is een looplicht waarvan de snelheid via een potentiometer kan worden gestuurd.

Onderdelen: 1 x steekplaatje, 1 x LED groen met voorweerstand, 1 x LED geel met voorweerstand, 1 x LED rood met voorweerstand, 1 x potentiometer van 15 kOhm, 4 x draadbrug (verschillende lengtes)

Het programma

Het programma voor vandaag draagt de naam Tag07.ino en bevindt zich in het downloadarchief in de directory Dag07.

```
int analogPin = A5;
int analogValue;

int led1 = 8;
int led2 = 6;
int led3 = 4;

void setup() {
  pinMode(analogPin, INPUT);

  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);
  Serial.begin(9600);
}

void loop() {
  analogValue = analogRead(analogPin);
  Serial.println(analogValue);

  delay(analogValue);
  digitalWrite(led3, LOW);
  digitalWrite(led1, HIGH);

  delay(analogValue);
  digitalWrite(led1, LOW);
  digitalWrite(led2, HIGH);

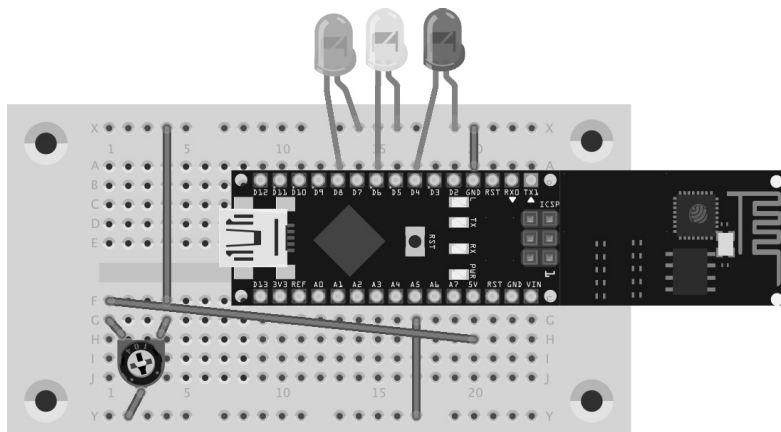
  delay(analogValue);
  digitalWrite(led2, LOW);
  digitalWrite(led3, HIGH);
}
```

Zo werkt het programma

```
analogValue = analogRead(analogPin);
delay(analogValue);
digitalWrite(led3, LOW);
digitalWrite(led1, HIGH);
```

In de tijdelijke variabele `analogPin` wordt de ingestelde waarde van de potentiometer gelezen. Via `delay` wordt gewacht en dan worden de LED's omgeschakeld.

7. dag



fritzing

De potentiometer neemt een beetje plaats in. Daarom moet het IoT-board in vergelijking met het model, naar boven worden geschoven. LED's van links naar rechts: groen, geel en rood.

8e dag

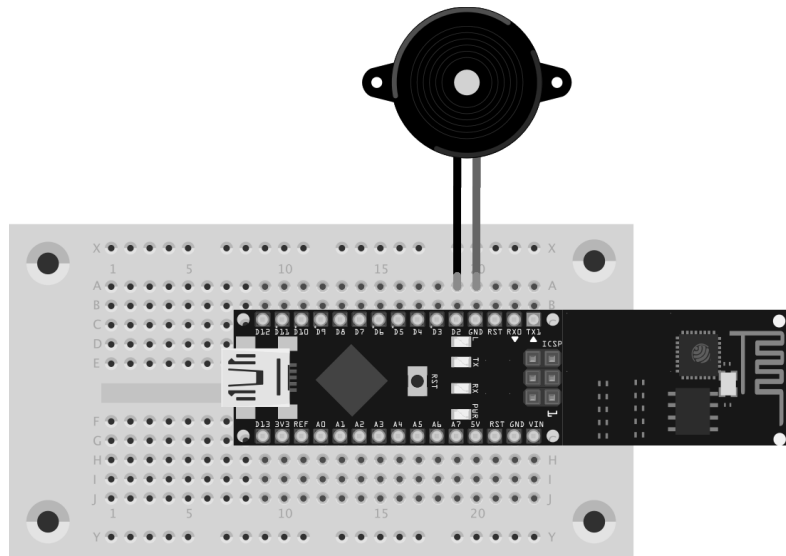
Vandaag in de adventskalender

• 1 x Piezo

Tonen via app uitvoeren

Via een app geeft u tonen op een piezo uit.

Onderdelen: 1 x steekplatine, 1 x Piezo

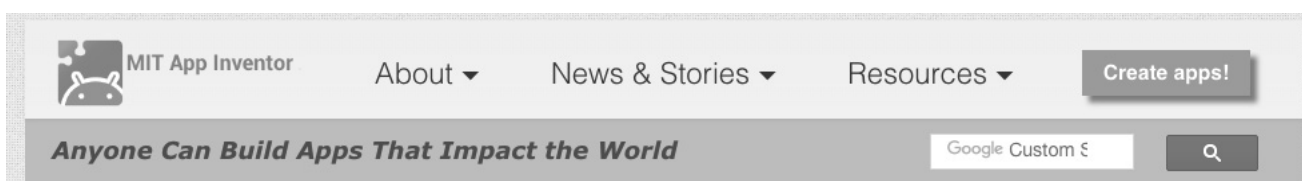


fritzing

De twee draden van de piezo worden met D2 en GND verbonden. De rest wordt via software geactiveerd.

Ontwikkelingsomgeving voor de apps

De besturing van het IoT-board gebeurt in deze adventskalender met de reeds gebruikte app **Serial Bluetooth Terminal** en met zelf-ontwikkelde smartphone apps voor het besturingssysteem Android6. De afgewerkte projectbestanden kunt u telkens van <http://www.buch.cd> downloaden. Als ontwikkelingsomgeving wordt MIT App Inventor 2 (<http://appinventor.mit.edu>) gebruikt.



De ontwikkelingsomgeving loopt in de browser. Daarom hebt u tijdens de ontwikkelingstijd ook toegang tot het internet nodig.

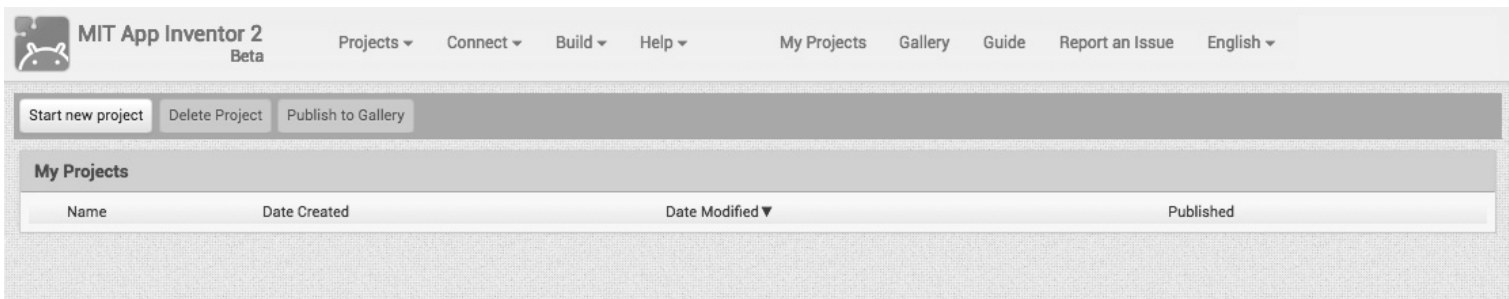
Met een klik op de knop **Create apps!** start u de ontwikkelingsomgeving. Voor het gebruik hebt u een gratis account bij Google nodig.

Account bij Google aanmaken

Als u nog geen account bij Google hebt, maakt u voor de ontwikkeling met de **MIT App Inventor 2 (AI2)** een account aan via <https://accounts.google.com>.

Nadat u zich met succes hebt aangemeld, bevindt u zich in het dashboard. Hier zit u alle totnogtoe ontwikkelde apps. Aangezien u waarschijnlijk nog geen app met AI2 hebt ontwikkeld, ziet u een leeg dashboard.

6 De op het IoT-board geïntegreerde chipset wordt alleen door Android smartphones ondersteund.

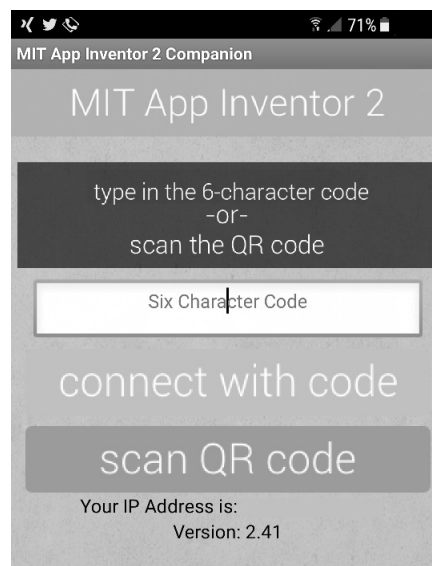


Het oppervlak is nog niet in het Nederlands beschikbaar.

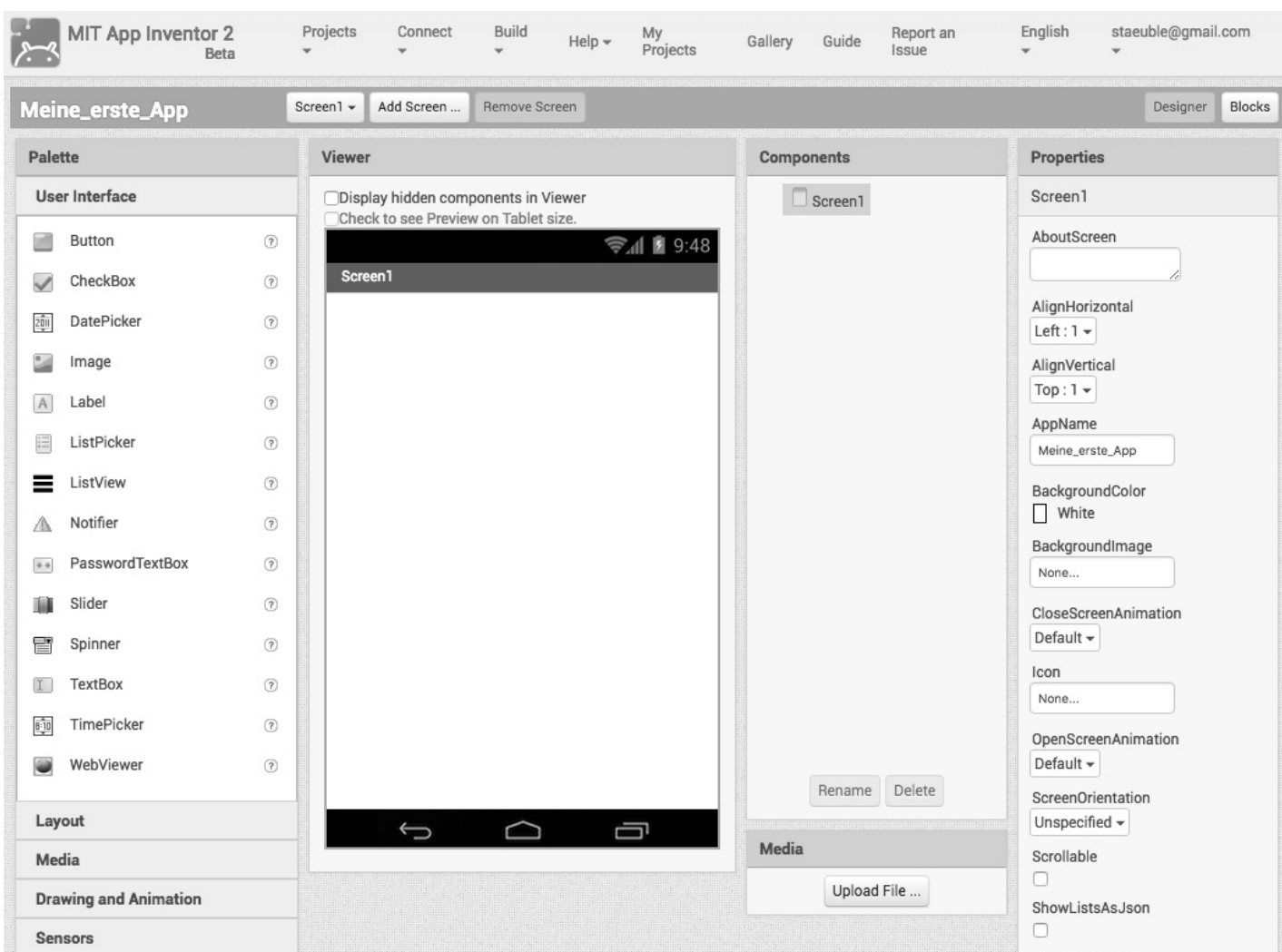
Om uw zelf-ontwikkelde app ook al tijdens de ontwikkeling snel te kunnen testen, moet u de gratis app **MIT AI2 Companion** uit de Google Play Store op uw smartphone installeren. Na de start moet u de code van uw app invoeren. Hiervoor moet u eerst een app aanmaken.

De eerste app met AI2

Kik nu op de startknop **Start new project** en voer als naam **Mijn_eerste_app** in. Let op dat de naam geen spaties mag bevatten. Nu opent de ontwikkelingsomgeving zich in uw browser.



Tijdens de ontwikkeling start u niet direct de app op, maar neemt u een omweg via MIT App Inventor 2 Companion. Daardoor bespaart u zich een manuele installatie van de app op het apparaat.



Na de start is het oppervlak van de app nog leeg.



Wijzigingen aan een onderdeel voert u in het venster **Properties** uit. Het aanzicht in de viewer verandert dan onmiddellijk.

Het venster is in meerdere delen opgedeeld. Links ziet u in het bereik **Palet** de oppervlakte-elementen die voor uw app beschikbaar zijn. Daarnaast ziet u in **Viewer** het oppervlak van de app. In **Components** ziet u de in uw app gebruikte onderdelen en in **Properties** de eigenschappen van de huidige geselecteerde onderdelen.

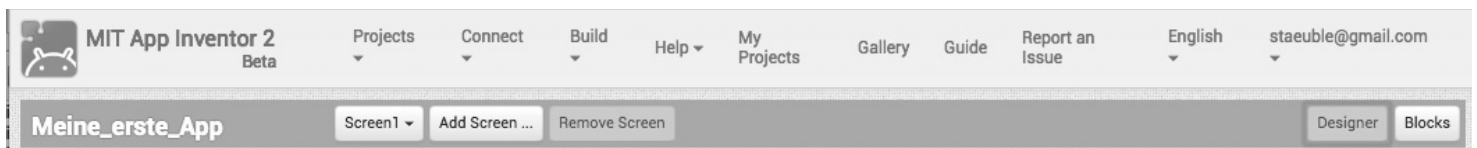
Trek nu een opschrift in de vorm van een label in het venster. Pas het opschrift aan door **Width** op **Fill parent** te zetten en **TextAlignment** op **center**. Nu is het opschrift horizontaal gecentreerd. Vervolgens verandert u de tekengrootte in **FontSize** 30 en plaatst u de tekst (Veld: **Tekst**) op **Mijn eerste app**.

Nu moet de app nog met een interactie worden uitgebreid. Hiervoor voegt u een knop uit het venster **Components** toe en wijzigt u via **Properties** het opschrift in **Touch me**. Daaronder voegt u een label met het opschrift **Status: not touched** toe.



App-oppervlak met de drie componenten

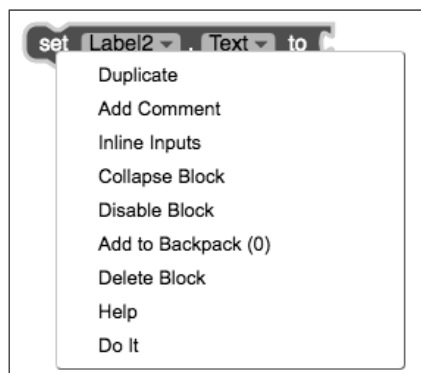
AI2 heeft twee aanzichten voor een app: het aanzicht voor de weergave van het oppervlak, de zogenaamde designer (driver **Designer**), en het aanzicht voor de programmering in de bloktaal (driver **Blocks**). Deze beide drivers ziet u bovenaan de ontwikkelingsomgeving.



Via de drivers **Designer** en **Blocks** gaat u naar beide aanzichten van de ontwikkelingsomgeving.

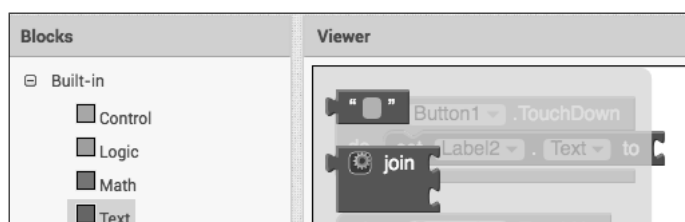
Nadat u op **Blocks** hebt geklikt, opent het aanzicht van de Block-programmeertaal. Kies nu binnen het bereik **Blocks** de onderdelen **Button1** en u ziet al de beschikbare gebeurtenissen waarop u in de app kunt reageren.

Selecteer nu de twee blokken **when Button1.TouchDown** en **when Button1.TouchUp**. In beide gevallen moet de inhoud van het label **Label2** worden aangepast. Klik hiervoor op **Label2** en selecteer het element **set Label2.Text to**. Ofwel voegt u het element tweemaal in of voegt u het element eenmaal in en kopieert u het element via het contextmenu (rechter muisknop) van de onderdelen.



Onderdelen hebben een contextmenu. Via **Duplicate** kopieert u de geselecteerde onderdelen.

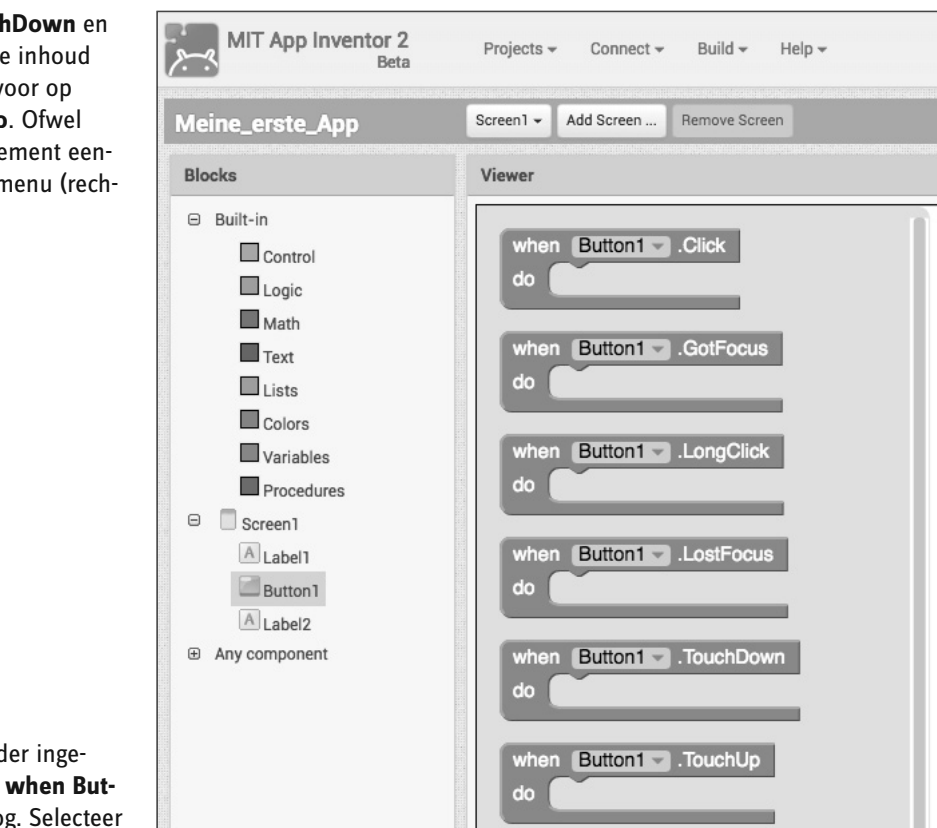
Beide onderdelen voegt u nu binnen de beide eerder ingevoegde elementen (**when Button1.TouchDown** en **when Button1.TouchUp**) in. Nu ontbreekt de tekstinhoud nog. Selecteer hiervoor uit de categorie **Text** het lege element (bovenste element).



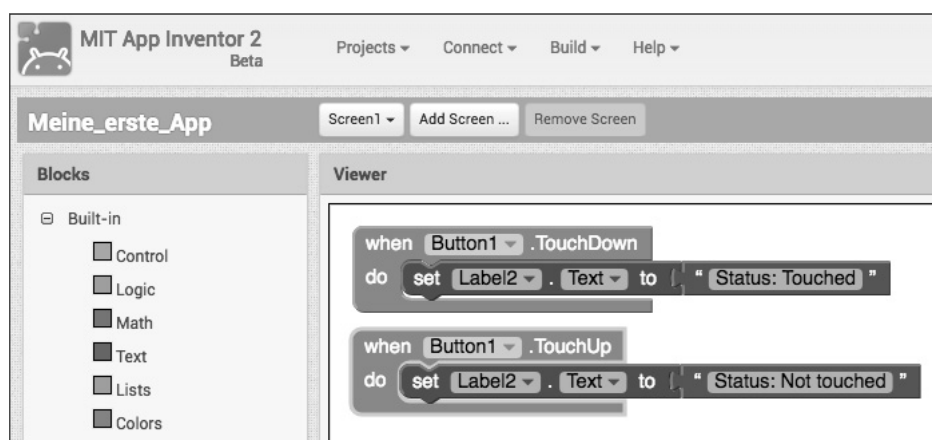
Het element met de twee aanhalingstekens wordt gebruikt om het opschrift te wijzigen.

Ingeval van **when Button1.TouchDown** wordt de tekst op **Status: Touched** ingesteld. Bij **when Button1.TouchUp** wordt de tekst op **Status: Not touched** geplaatst.

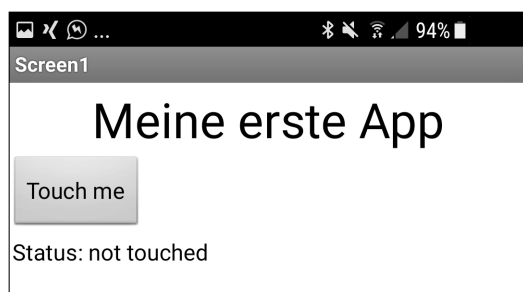
Nu kunt u de app op uw smartphone testen. Selecteer daarvoor in het menu **Connect / AI Companion**. Nu verschijnt er een tekst- en een QR-code. Voer nu hetzij de code in **MIT AI2 Companion** in of scan de QR-code via **scan QR code**. De app starten u met **connect with Code**. Nu opent de app zich op uw smartphone. Wanneer u nu de knop aanraakt, verandert de titel.



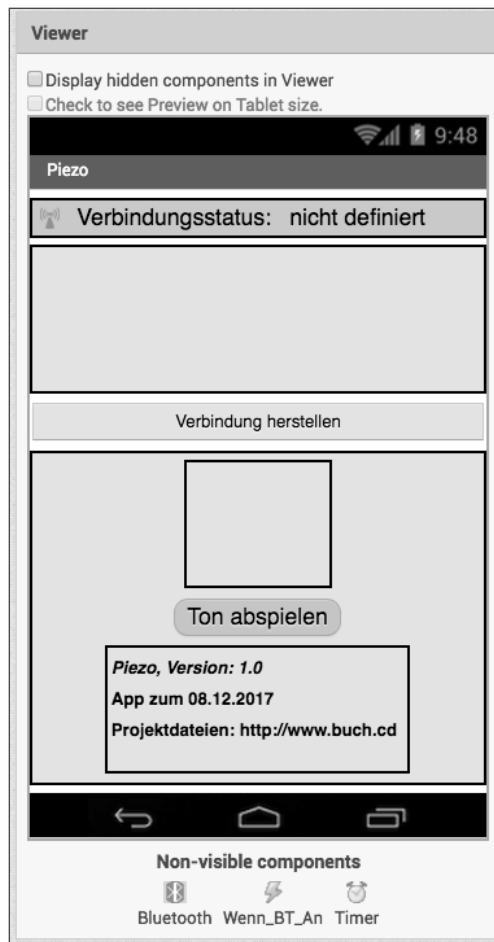
Voor de programmering van de logica, kunt u voor de keuze via categorieën (**Built-in**) of via de onderdelen (**Screen1**) gaan.



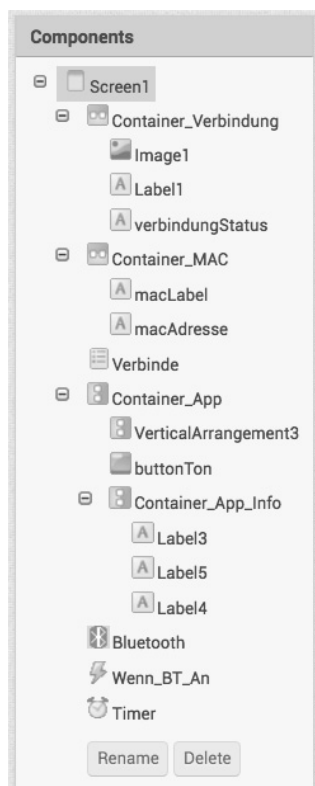
De tekst van het label wordt bij het aanraken van de knop aangepast.



Zo eenvoudig is de app-ontwikkeling met AI2.



Bovenaan de app wordt aangeduid of de Bluetooth-verbinding al of niet is aangemaakt.



Alle onderdelen in de app in een hiërarchisch overzicht. De belangrijke onderdelen werden van sprekende namen voorzien.

Piezo met app besturen

Om het IoT-board via een app te sturen, hebt u een sketch op het board nodig die op de commando's van de app reageert en een app. Vandaag moet de meegeleverde piezo via een app worden gestuurd. De sketch die daarvoor nodig is, is `Dag08.ino` in de directory `Dag08`. Laad deze sketch op het IoT-board op.

De sketch baseert zich op het reeds gebruikte bestand `Model.ino`. Het principe van alle sketches in deze adventskalender die op een app reageren, is als volgt: Er wordt een tekst via de draadloze interface verzonden en deze wordt in de sketch geanalyseerd en dan wordt een actie uitgevoerd. In de andere richting gaat het net zo: De app ontvangt via de draadloze interface een tekst van het IoT-board, analyseert de tekst en voert dan een overeenkomstige actie uit. Het commando om een melodie af te spelen, luidt `melodie`:

```
if (Text.startsWith("Ton") || Text.startsWith("TON") || Text.startsWith("ton")){
    playMelody();
}
```

De daarbijhorende app bevindt zich in het bestand `Piezo.aia`. Importeer het bestand in AI2.

Projecten in AI2 importeren

Alle apps liggen voorgemaakt als aia-bestanden in het downloadarchief gereed. Om een bestand in AI2 te openen, moet u het projectbestand via het menupunt **Projects / Import project (.aia) from my computer ...** importeren.

Werkwijze van de app

De volgende apps die met AI2 opgemaakt werden, zijn aan de hand van een model opgebouwd en daarom wordt deze app nu beschreven en in het verdere verloop wordt daarvoor verwezen. De app bestaat uit in totaal vier bereiken:

- Verbindingsbereik
- Werkbereik
- App-infobereik
- Intern, niet-zichtbaar bereik

In de linker afbeelding ziet u de afzonderlijke bereiken van de app. Bij het verbindingsbereik horen **Container_Verbinding** en **Container_MAC**. De eigenlijke app die ook van project tot project verandert, bevindt zich in **Container_App**. Deze container bevat het functiebereik (**buttonTon**) en het app-infobereik (**Container_App_Info**).

Wat is een container?

Om de elementen te plaatsen zijn er in de categorie **Layout** meerdere layoutmanagers (bv. **HorizontalArrangement**) beschikbaar; deze helpen u bij de horizontale en verticale plaatsing. De elementen worden binnen een dergelijke layoutmanager geplaatst. Aangezien deze layoutmanagers meestal meerdere elementen bevatten, worden deze ook containers (layoutcontainers) genoemd.

Onderdelennamen wijzigen

Wanneer u onderdelen tot een oppervlak in de AI2 toevoegt, hebben de onderdelen een automatisch gegenereerde naam. In tegenstelling tot andere eigenschappen van een onderdeel kunt u de naam niet via het venster **Properties** wijzigen, maar in het bereik, **Components**. Hier selecteert u de overeenkomstige onderdelen en klikt u op de knop **Rename**.

Het oppervlak bevat in total drie niet-zichtbare onderdelen. Voor het gebruik van Bluetooth hebt u de onderdelen **BluetoothClient** uit de categorie **Connectivity** nodig. De onderdelen worden daarbij slechts in het oppervlaktebereik getrokken en moeten niet verder geconfigureerd worden. Voor de keuze van het overeenkomstige Bluetooth-apparaat hebt u het onderdeel **ActivityStarter** (in de bovenste afbeelding **Wanneer_BT_Aan**) uit **Connectivity** nodig. Aan de ActivityStarter wordt in het venster **Properties** de waarde `android.bluetooth.adapter.action.REQUEST_ENABLE` toegewezen. Voor de constante controle van

de draadloze verbinding wordt het element **Clock** (in de bovenste afbeelding **Timer**) uit **Sensors** gebruikt. Het **TimeInterval** wordt in het venster **Properties** op **200** gezet.

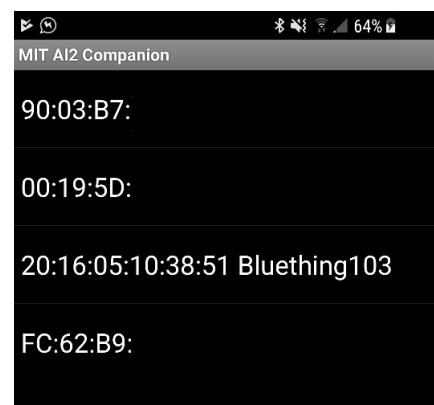
De programmering gebeurt in de driver **Blocks**. Via de **Timer** wordt de verbindingstatus gezet. Via de eigenschap **isConnected** kunt u controleren of de app met een Bluetooth-apparaat is verbonden.



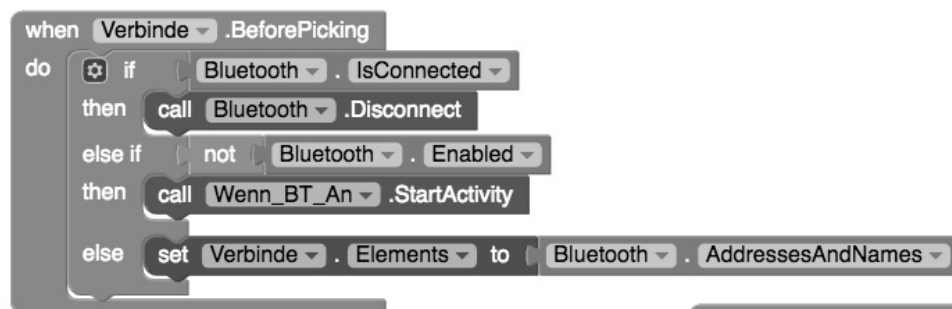
Regelmatig wordt gecontroleerd of de draadloze verbinding al of niet bestaat.

Bij een klik op de knop **Verbinding maken** wordt de lijst van de beschikbare Bluetooth-apparaten weergegeven. Als de lijst leeg is, moet u in de app terugkeren en dan wordt automatisch na de toegangsautorisering op de Bluetooth-interface gevraagd. Als wachtwoord voert u **1234** in. Wanneer men opnieuw op de knop klikt, moet de lijst worden weergegeven.

Hier moet ook een invoer met de naam **Bluething** opduiken. Na een klik op de invoer moet u nog het wachtwoord **1234** invoeren en de app is met het board verbonden. De weergave van deze lijst wordt via het blok **when Verbind.BeforePicking** gerealiseerd. Zoals u in het volgende blok ziet, heeft de knop twee functies: Verbinding maken en verbinding ontkoppelen.



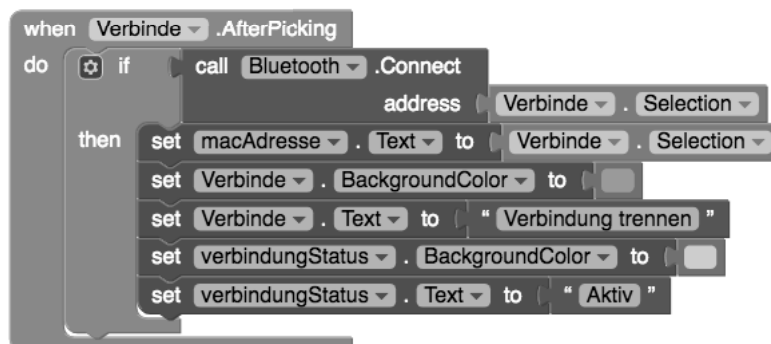
De naam en het MAC-adres van de beschikbare apparaten worden weergegeven.



Via de knop kan de Bluetooth-verbinding tussen het board en de app geïnitieerd worden.

De verbinding zelf wordt in het blok **when Verbinde.AfterPicking** gemaakt.

De bovenste drie blokken zullen in alle apps in deze adventskalender voorkomen. De eigenlijke functionaliteit in de huidige app is in het blok **when buttonTon.Click** gerealiseerd. In deze waarde wordt slechts de waarde **Melodie** bepaald, afgesloten door een **\n**. Het verzenden van een tekst via de draadloze interface gebeurt via het blok **call Bluetooth.SendText**. Als parameters voert u de tekst in. Vergeet niet een eindcriterium te definiëren. In dit voorbeeld is het eindcriterium **\n**. De sketch reageert op dit teken.



De verbinding met het geselecteerde apparaat wordt gemaakt.



Strings kunnen via de onderdelen **join** met elkaar worden verbonden.

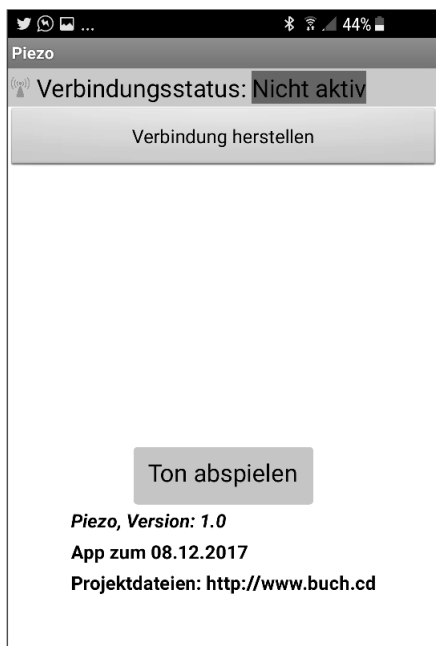
Blokken naar andere projecten kopiëren

In het bereik **Blocks** zit u een rugzak, het zogenaamde backpack. Hiermee kunt u blokken uit de blokeditor van een project naar een ander project kopiëren. Trek gewoon de overeenkomstige blokken in het backpack en u kunt de blokken in een ander project opnieuw gebruiken. Om het backpack opnieuw leeg te maken, klikt u met de rechter muisknop op het werkvlak en kiest u **Empty Backpack**.

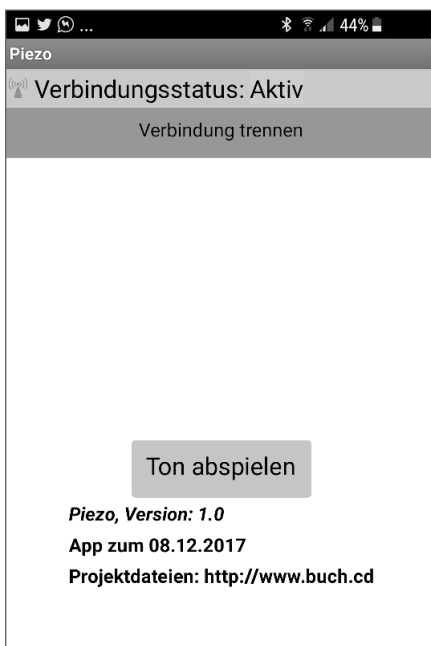
App testen

Start de app nu via **Connect / AI Companion** en roep de app **MIT AI2 Companion** op uw Android-smartphone op. Scan de QR-code of voer de code in en start de app via **connect with code**. Nu start de app.

Klik nu op **Verbinding maken**. Nu kunt u zich met het IoT-board verbinden.



In het begin is de verbinding nog niet actief.



De app is nu met het IoT-board verbonden.

Wanneer u nu **Melodie afspelen** aanraakt, wordt op het IoT-board een melodie via de piezo afgespeeld.

9e dag

Vandaag in de adventskalender

• 1 x RGB-LED met voorweerstand

RGB-LED's

Een normale LED licht altijd slechts in een kleur op. De in de adventskalender gebruikte RGB-LED's kunnen naar keuze in meerdere kleuren oplichten. Hier zijn in principe drie LED's met verschillende kleuren in een doorzichtige behuizing ingebouwd. Elk van deze drie LED's heeft een eigen anode via dewelke ze met een GPIO-pin wordt verbonden. De kathode die met de massaleiding wordt verbonden, is slechts eenmaal aanwezig. Daarom heeft een RGB-LED vier aansluitdraden.

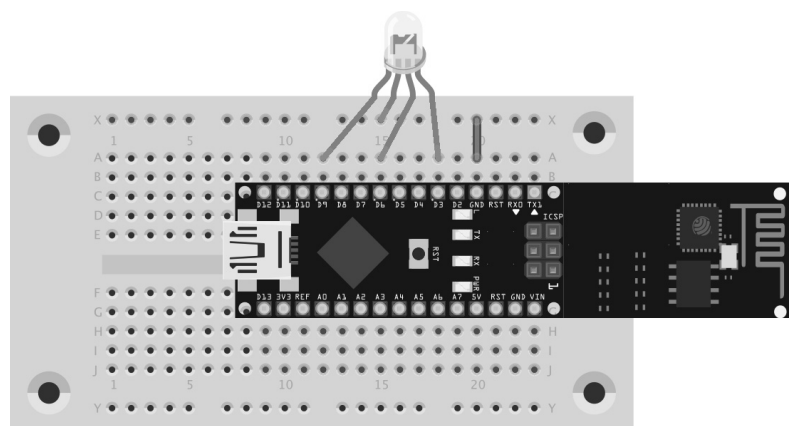
De aansluitdraden van de RGB-LED's hebben een verschillende lengte om ze uniek kenbaar te maken. In tegenstelling tot normale LED's is de kathode hier de langste draad.

RGB-LED's werken zoals drie afzonderlijke LED's en hebben daarom ook drie voorweerstanden van 220 Ohm (rood-rood-bruin) nodig. In de RGB-LED's in deze adventskalender zijn deze reeds ingebouwd.

Kleur van een RGB-LED via app wijzigen

Vandaag programmeert u een app om een kleur van de RGB-LED te veranderen.

Onderdelen: 1 x insteekprintplaat, 1 x RGB-LED met voorweerstand, 1 x draadbrug



fritzing

Ook de RGB-LED heeft reeds de voorweerstanden geïntegreerd. Het tweede been (kort been) is de kathode en moet met de massa worden verbonden.

De sketch voor het IoT-board

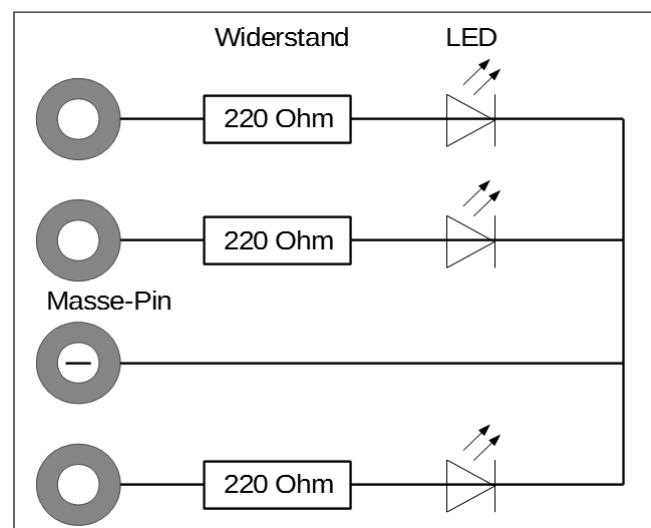
Het programma voor vandaag draagt de naam Tag09.ino en bevindt zich in het downloadarchief in de directory Dag09. De sketch reageert op vier toetsen:

```
if (Text.startsWith("Rot") || Text.startsWith("ROT") || Text.startsWith("rot")){
  rood();
} else if (Text.startsWith("Blauw") || Text.startsWith("BLAUW") || Text.startsWith("blauw")){
  blauw();
} else if (Text.startsWith("Groen") || Text.startsWith("GROEN") || Text.startsWith("groen")){
  groen();
} else if (Text.startsWith("Uit") || Text.startsWith("UIT") || Text.startsWith("uit")){
  uit();
}
```

9. dag



Aansluitpins van een RGB-LED



Schakelplan voor een RGB-LED met 3 voorweerstanden.



De LED's worden via vier knoppen geschakeld.

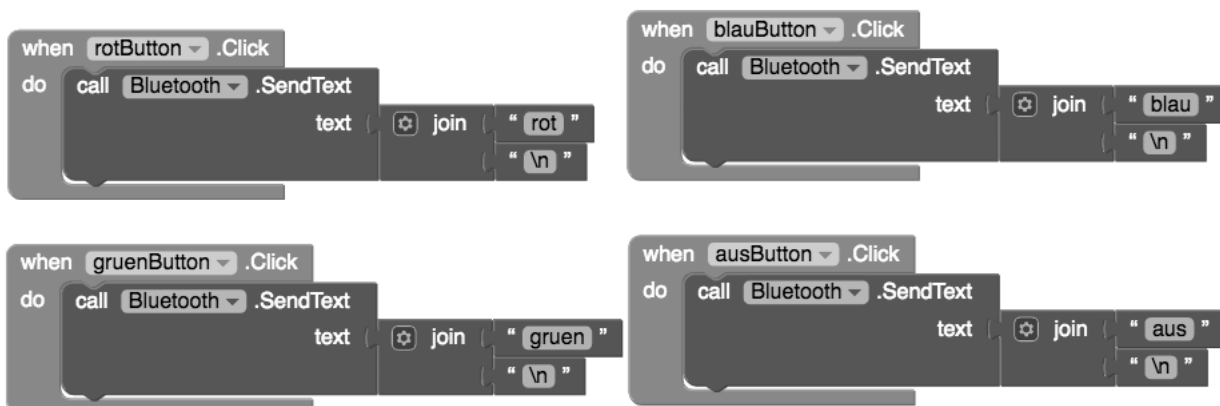
Elke kleur heeft een eigen functie binnen de sketch. Daarna wordt de functie `rood()` voorgesteld:

```
void rood() {
  Serial.println("Rot");
  analogWrite(redPin, HIGH);
  analogWrite(greenPin, LOW);
  analogWrite(bluePin, LOW);
}
```

De app

De daarbij horende app bevindt zich in het bestand `RGB.aia`. Importeer het bestand in AI2.

Afhankelijk van de ingedrukte knop wordt een overeenkomstige tekst via de Bluetooth-interface verzonden: Voor rood wordt `rood\n` verzonden, voor blauw wordt `blauw\n` verzonden, voor groen wordt `groen\n` verzonden, en voor het uitschakelen `uit\n` verzonden.



De blokken in de AI2

10e dag

Vandaag in de adventskalender

• 1 x knop

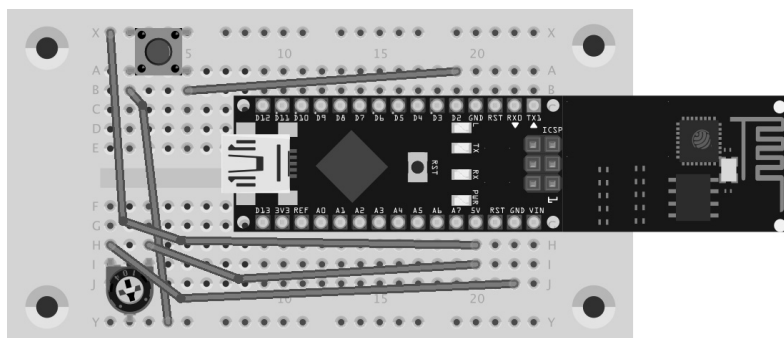
10. dag

Toetsdruk weergeven

Vandaag reageert uw IoT-board op een mechanische druk op de knop en geeft een melding aan de draadloze interface door.

Onderdelen: 1 x steekplatine, 1 x knop, 1 x potentiometer, 5 x draadbruggen (verschillende lengtes)

Digitale pins kunnen niet alleen gegevens uitgeven, bijvoorbeeld via LED's, maar ook voor gegevensinvoer worden gebruikt. Voor de invoer gebruiken wij in het huidige project een knop die rechtstreeks op de steekplatine wordt gestoken. De knop heeft vier aansluitpins, waarbij telkens twee tegenoverliggende (grote afstand) met elkaar verbonden zijn. Zolang de knop ingedrukt is, zijn alle vier de aansluitingen met elkaar verbonden. In tegenstelling tot een schakelaar klikt een knop niet in. De verbinding wordt bij het loslaten onmiddellijk opnieuw verbroken.



fritzing

Het IoT-board is zeer lang en daarom moet de toets dwars over het steekplatine worden gestoken en deels moeten de draadbruggen onder de aangesloten USB-kabel worden gelegd.

Als er met een digitale ingang een +5-V-sigitaal verbonden is, wordt deze als logisch **waar** geanalyseerd.

Bij een open knop heeft de ingang geen uniek gedefinieerde toestand. Wanneer een programma deze pin opvraagt, kan het tot toevallige resultaten komen. Om dat te voorkomen, sluit met ter vergelijking een zeer hoge weerstand tegen massa. Deze zogenaamde Pull down-weerstand trekt de status van de ingangspin bij geopende knop opnieuw naar beneden tot 0 V. Aangezien de weerstand zeer hoog is, bestaat er ook geen kortsluitingsgevaar zolang de knop is ingedrukt. In de ingedrukte toestand zijn +5 V en de massaleiding rechtstreeks via deze weerstand verbonden.

De sketch

Het programma voor vandaag draagt de naam `Tag10.ino` en bevindt zich in het downloadarchief in de directory `Dag10`. Door het oproepen van **digitalRead** wordt geanalyseerd of de knop ingedrukt werd:

```
void loop() {
  int reading = digitalRead(buttonPin);

  if (reading != lastButtonState) {
    lastDebounceTime = millis();
  }

  if ((millis() - lastDebounceTime) > debounceDelay) {
    if (reading != buttonState) {
      buttonState = reading;

      if (buttonState == HIGH) {
        ledState = !ledState;
      }
    }
  }

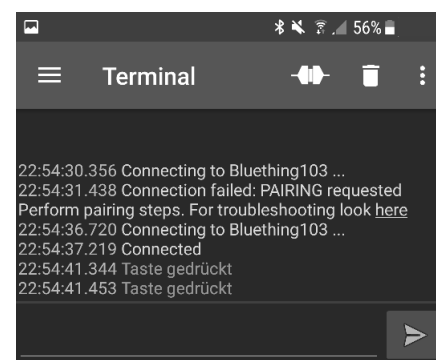
  digitalWrite(LedPin, ledState);
}
```

Als de knop werd ingedrukt, wordt een bericht via de draadloze interface verzonden:

```
if (reading != lastButtonState) {
  HC05.print("Knop ingedrukt\n");
}
```

Reactie van het IoT-board weergeven

Voor de weergave van de druk op de knop op de smartphone, wordt de reeds gebruikte app **Serial Bluetooth Terminal** gebruikt. Na de verbinding met het IoT-board wordt bij een druk op de knop de melding **Knop ingedrukt** weergegeven.



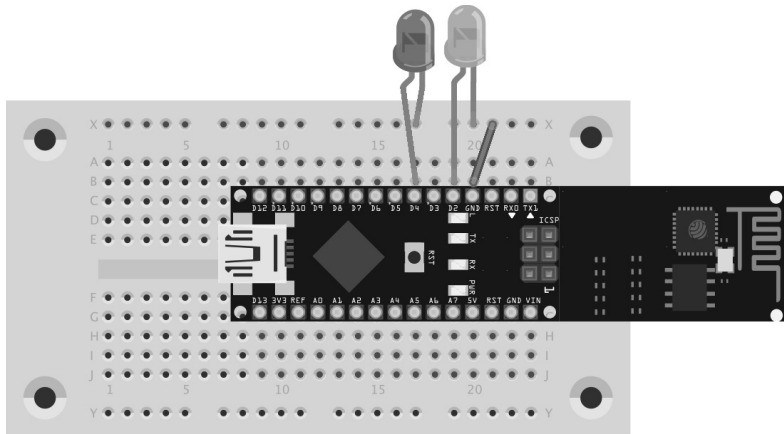
Ontvangen meldingen worden in het groen in het terminalvenster geschreven.

11. dag

11e dag

Vandaag in de adventskalender

• 1 x schakeldraad



fritzing

Door de geïntegreerde voorweerstanden is de schakeling zeer compact. LED's van links naar rechts: rood en groen.

LED-echo via app

Het huidige project is een LED-echo. Ze stellen een sequentie via twee knoppen in de app in en de LED's op het steekplaatje knipperen in deze sequentie.

Onderdelen: 1 x steekplaatje, 1 x LED rood met voorweerstand, 1 x LED groen met voorweerstand, 1x draadbrug

De sketch

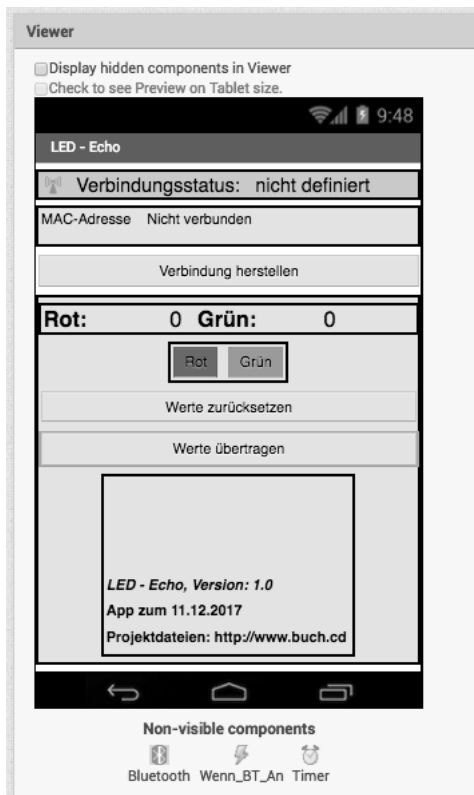
Het programma voor vandaag is Tag11.ino en bevindt zich in de directory Dag11. De echosequentie wordt door de app bepaald. Daarbij komt de tekst in de volgende vorm: RNNNGNN. R staat voor rood en G voor groen, bv. R5G3 - hier zou de roe LED vijfmaal knipperen en daarna de groene LED driemaal:

```
if (Text.indexOf("R") != -1 && Text.indexOf("G") != -1) {
  zetKleur(Text);
} else if (Text.startsWith("UIT") || Text.startsWith("UIT")) {
  uit();
}
```

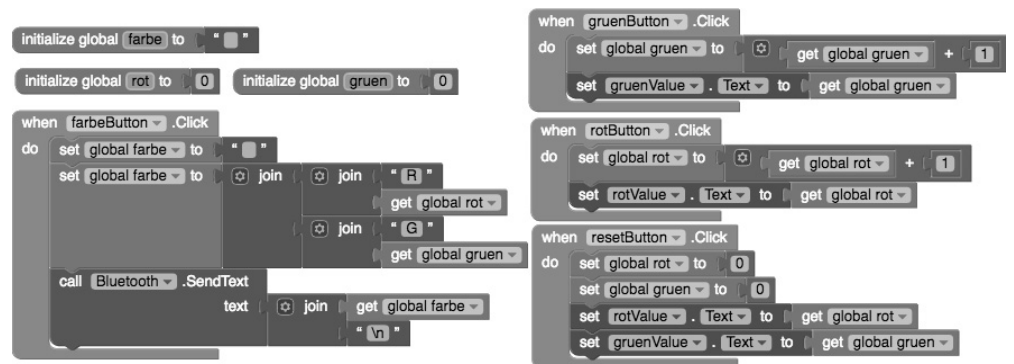
De app

De app heeft naast de reeds bekende **Verbinding maken** vier knoppen: **Rood**, **Groen**, **Waarden overdragen** en **Waarden terugzetten**.

Bij het aanraken van **Rood** of **Groen** wordt het overeenkomstige label met een verhoogd. Met **Waarden terugzetten** zet u beide waarden terug op 0. Met **Waarden overdragen** bepaalt u de waarden aan het IoT-board. De te bepalen waarde wordt daarbij in een globale variabele tussenopgeslagen. De inhoud van de variabelen wordt bij het aanraken via Bluetooth naar **Waarden overdragen** overgedragen.



De app gebruikt de reeds gekende app van dag 8.



Voor elke knop is er een aparte when.Click-vraag.

12e dag

Vandaag in de adventskalender

• 1 x LED oranje met voorweerstand

Looplichtsnelheid via app instellen

In het huidige project stuurt u de snelheid van een looplicht via Bluetooth.

Onderdelen: 1 x steekplatine, 1 x LED rood met voorweerstand, 1 x LED oranje met voorweerstand, 1 x LED groen met voorweerstand, 1 x draadbrug

De sketch

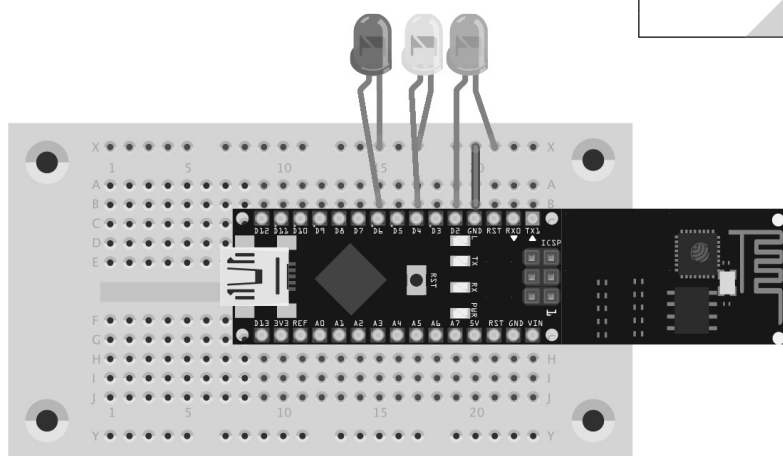
Het programma voor vandaag is `Tag12.ino` en bevindt zich in de directory `Dag12`. Voor de sturing van het looplicht zijn twee parameters nodig: de wachttijd terwijl een andere LED oplicht en het aantal doorlopen. Deze waarden worden door de app in een string van de vorm `wTTTdNN` verzonden. Bijvoorbeeld `w500d4` dat elke LED 500 ms brandt en voor het inschakelen van een andere LED wordt 500 ms gewacht. In totaal zou het looplicht viermaal lopen:

```
if (Text.indexOf("w") != -1 && Text.indexOf("d") != -1) {
  String temp = Text.substring(Text.indexOf("w")+1,Text.indexOf("d"));
  int wachttijd = temp.toInt();
  temp = Text.substring(Text.indexOf("d")+1);
  int doorlopen = temp.toInt();
  looplicht(wachttijd, doorlopen);
}
```

De app

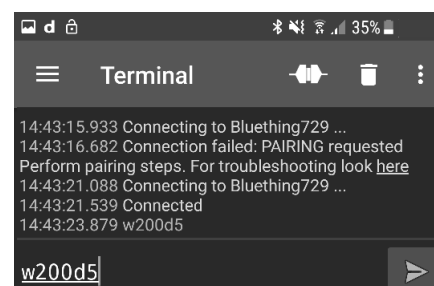
Om de waarden te bepalen wordt de app **Serial Bluetooth Terminal** gebruikt. Na het aanmaken van de verbinding kunt u rechtstreeks het commando in het beschreven formaat naar het IoT-board verzenden.

12. dag



Looplicht met drie LED's: rood, oranje en groen.

fritzing



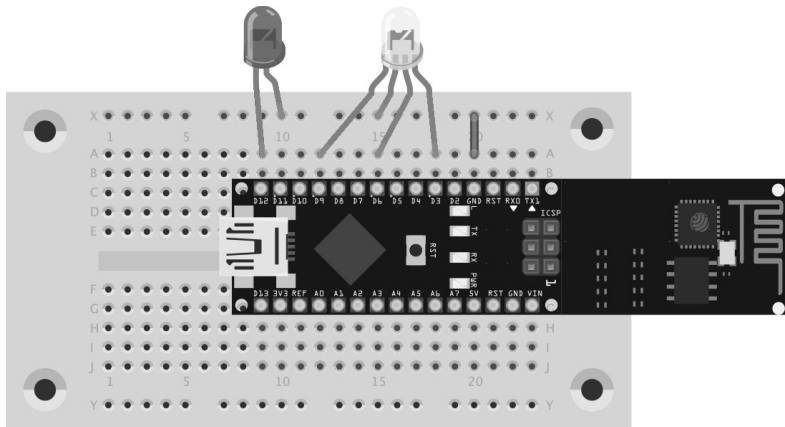
In de uitgabe ziet men dat twee verbindingspogingen nodig zijn voor een succesvolle verbinding met het IoT-board. Als de verbinding niet slaagt, controleert u via het menupunt **Bluetooth Devices**, of u het juiste toestel hebt gekozen.

13. dag

13e dag

Vandaag in de adventskalender

• 1 x LED roze met voorweerstand



fritzing

De roze LED wordt gebruikt om de opgebouwde verbinding te signaleren;

RGB via slider in app aanpassen

Een RGB-LED kan niet alleen een van de drie grondkleuren weergeven, maar ook overgangen. Met behulp van een app kunt u de kleur van de RGB-LED precies instellen.

Onderdelen: 1 x steekplaatje, 1 x RGB-LED met voorweerstand, 1 x LED roze met voorweerstand, 1x draadbrug

De sketch

Het programma voor vandaag is Tag13.ino en bevindt zich in de directory Dag13. De sturing van de RGB-LED gebeurt via een string in de vorm RNNNGNNNBNNN. Slechts wanneer een string in deze vorm door het IoT-board wordt ontvangen, wordt de kleur via de methode zetKleur gezet. Om de LED uit te schakelen, moet de string uit verzonden worden:

```
while(HC05.available() > 0){
  Tekst = HC05.read();
  Text.concat(Tekst);
  if (Tekst == '\n') {
    if (Text.indexOf("R") != -1 && Text.indexOf("G") && Text.indexOf("B") ) {
      zetKleur(Text);
    } else if (Text.startsWith("Uit") || Text.startsWith("UIT") || Text.startsWith("uit"))
    {
      uit();
    }
    Text="";
  }
}
```

In de functie zetKleur wordt de overgegeven tekst met de methode substring in afzonderlijke delen opgedeeld en de waarden vervolgens via analogWrite geschreven:

```
void zetKleur(String text) {
  if (text.indexOf("R") != -1 && text.indexOf("G") && text.indexOf("B") ) {
    String temp = text.substring(text.indexOf("R")+1,text.indexOf("G"));
    if (temp.indexOf(".") != -1) {
      temp = temp.substring(0, temp.indexOf("."));
    }
    int rot = temp.toInt();

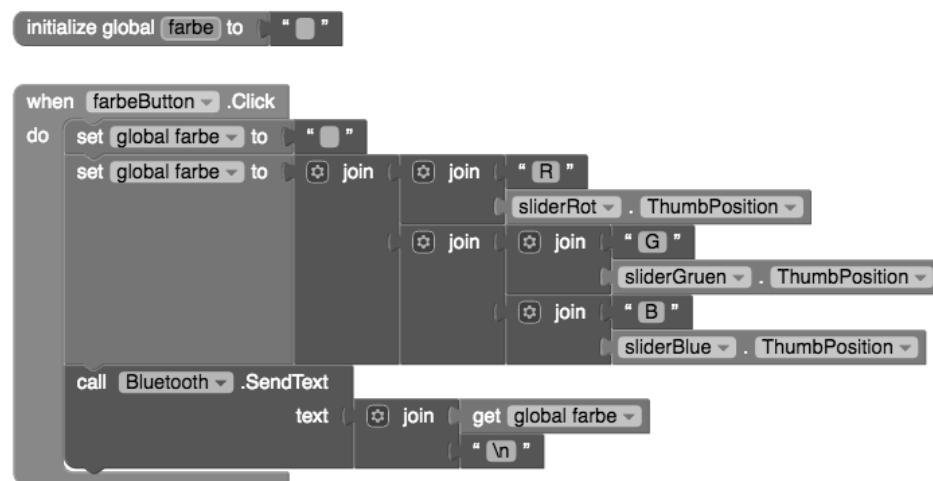
    temp = text.substring(text.indexOf("G")+1,text.indexOf("B"));
    if (temp.indexOf(".") != -1) {
      temp = temp.substring(0, temp.indexOf("."));
    }
    int groen = temp.toInt();

    temp = text.substring(text.indexOf("B")+1);
    if (temp.indexOf(".") != -1) {
      temp = temp.substring(0, temp.indexOf("."));
    }
    int blauw = temp.toInt();
    analogWrite(redPin,rot);
    analogWrite(greenPin,groen);
    analogWrite(bluePin, blauw);
  }
}
```

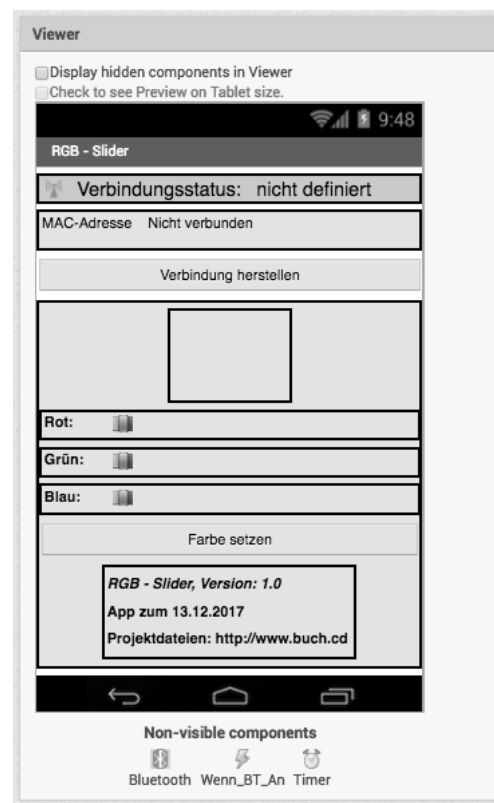
De app

AI2 dient als ontwikkelingsomgeving voor de huidige app. Importeer daarvoor het bestand `RGB_Slider.aia`. Via drie sliders wordt de waarde van het respectievelijke kleur ingesteld. Door het indrukken van **Kleur zetten** wordt de waarde van de drie sliders naar het IoT-board verzonden.

Bij het aanraken van **Kleur zetten** wordt de te bepalen tekst in de variabele `kleur` samengezet. Via `Bluetooth.SendText` wordt de tekst via de draadloze interface verzonden.



De samenstelling van de string gebeurt via **join**, dat men ook kan dozen kan.



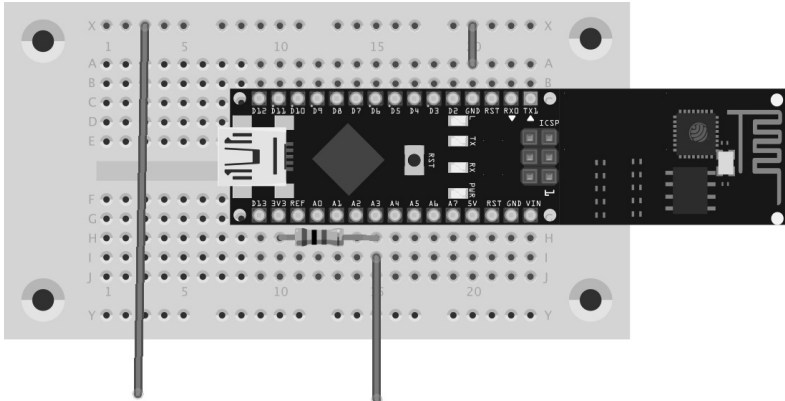
Elke slider is samen met een label in een **HorizontalArrangement**. Opdat de sliders allemaal onder elkaar zijn, werd de breedte van de drie labels op 15% ingesteld.

14. dag

14e dag

Vandaag in de adventskalender

- 1 x boetseerlei rood
- 1 x weerstand 20 MOhm



Boetseerlecontact

Vandaag gebruikt u de inbegrepen boetseerlei als invoer en wijzigt u bij het aanraken van de boetseerlei de achtergrondkleur van een oppervlak in de app.

Onderdelen: 1 x steekplatine, 2 x stuk boetseerlei, 1 x weerstand 20 MOhm, 3 x draadbruggen (verschillende lengtes)

Zo werken sensorcontacten

De als ingang geschakelde pin is via een extreem hoog-ohmige weerstand (20 MOhm) met +3,3 V verbonden zodat een zwakke, maar duidelijk als High gedefinieerd signaal aanwezig is. Een mens die niet helemaal vrij in de lucht zweeft, is altijd geaard en levert via de elektrisch geleidende huid een Low-niveau. Als deze mens nu een sensorcontact aanraakt, wordt het zwakke High-Signal met het duidelijk sterkere Low-niveau van de vingertop gemengd en trekt de pin naar Low-niveau.

Hoe hoog de weerstand tussen hand en massa werkelijk is, hangt van veel dingen af, onder meer van schoenen en vloer. Blootsvoets in het natte gras is de verbinding met de massa van de aarde het best, maar ook op een stenen vloer werkt het meestal goed. Houten vloeren isoleren beter, kunststof vloerbekledingen zijn vaak zelfs positief geladen. Opdat de schakeling altijd werkt is, zoals bij sensortoetsen aan liften en deuren bij elke schakeling een bijkomend massacontact voorzien. Als men dit aanraakt en tegelijk de eigenlijke sensor, is de massaverbinding in elk geval aangemaakt.

Boetseerlei leidt de stroom ongeveer zo goed als menselijke huid.

Ze laat zich in elke willekeurige vorm brengen en een boetseerlecontact kan veel makkelijker worden vastgenomen dan een eenvoudig stuk draad. Het oppervlak waarmee de hand het contact raakt, is duidelijk groter. Zo komt het niet zo snel tot een "loszittend contact". Snijd een lang stuk van ongeveer 10 cm van de schakeldraad af, verwijder aan beide uiteinden over ongeveer 1 cm lengte de isolatie en steek het ene uiteinde in een stuk boetseerlei. Steek het andere uiteinde in de steekplatine, zoals op de afbeelding.

Het IoT-board beschikt over analoge ingangen die zeer goed geschikt zijn voor sensorcontacten. Analoge ingangen leveren waarden tussen 0 (low niveau) en 1023 (high niveau): waarden tussen 100 en 200 zijn goede grenswaarden om een onderscheid te maken tussen aangeraakt en niet-aangeraakt sensorcontact.

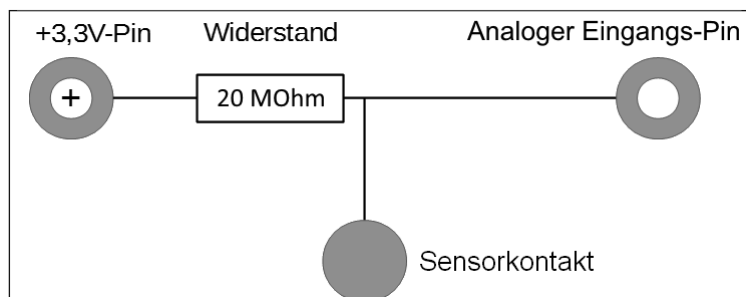
De sketch

Het programma voor vandaag is Tag14.ino en bevindt zich in de directory Dag14. De eigenlijke functie in deze sketch kan heel snel worden geprogrammeerd. Met `analogRead` wordt de waarde ingelezen en wanneer deze onder de drempelwaarde ligt, dan wordt via de draadloze interface een tekst met `HC05.print` verzonden:

```
void loop() {
  sensorValue = analogRead(sensorPin);

  if (sensorValue < 256) {
    HC05.print("LOW\n");
    delay(1000);
  }
}
```

Het uiteinde van beide uitstekende draadbruggen gaat telkens in een stuk boetseerlei. U vormt het best telkens een kleine ronde bol. Het boetseerlecontact aan A3 wordt als ingang gebruikt.



Schakelschema voor sensorcontacten aan het IoT-board

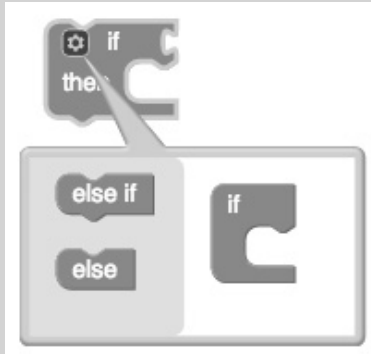
De app

De huidige app `Boetseerklei_schakelaar.aia` bevat een label dat bij het begin oranje als achtergrondkleur heeft.

Om te controleren of een tekst via de Bluetooth-interface werd verzonden, moet regelmatig worden gecontroleerd of er een tekst aanwezig is. Dit wordt in de aanwezige timer gedaan.

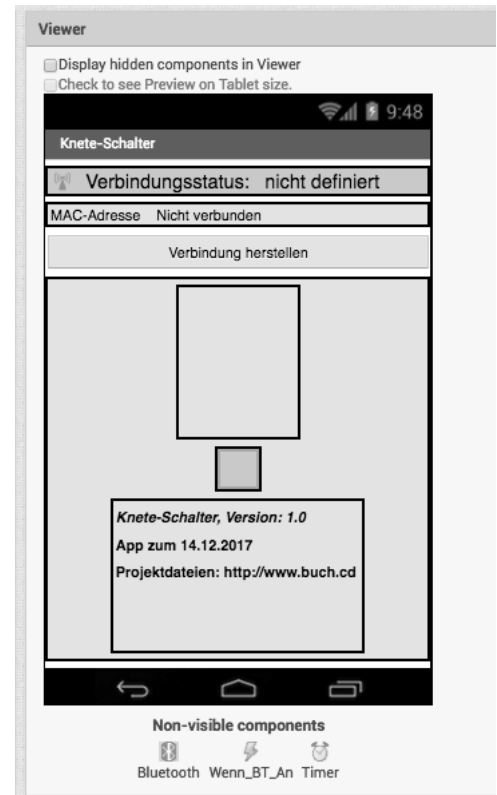
Blokken configureren

De elementen in het bereik **Blocks** in de blokeditor kunnen deels geconfigureerd worden. De configuratie gebeurt via het blauwe tandwiel aan het onderdeel.

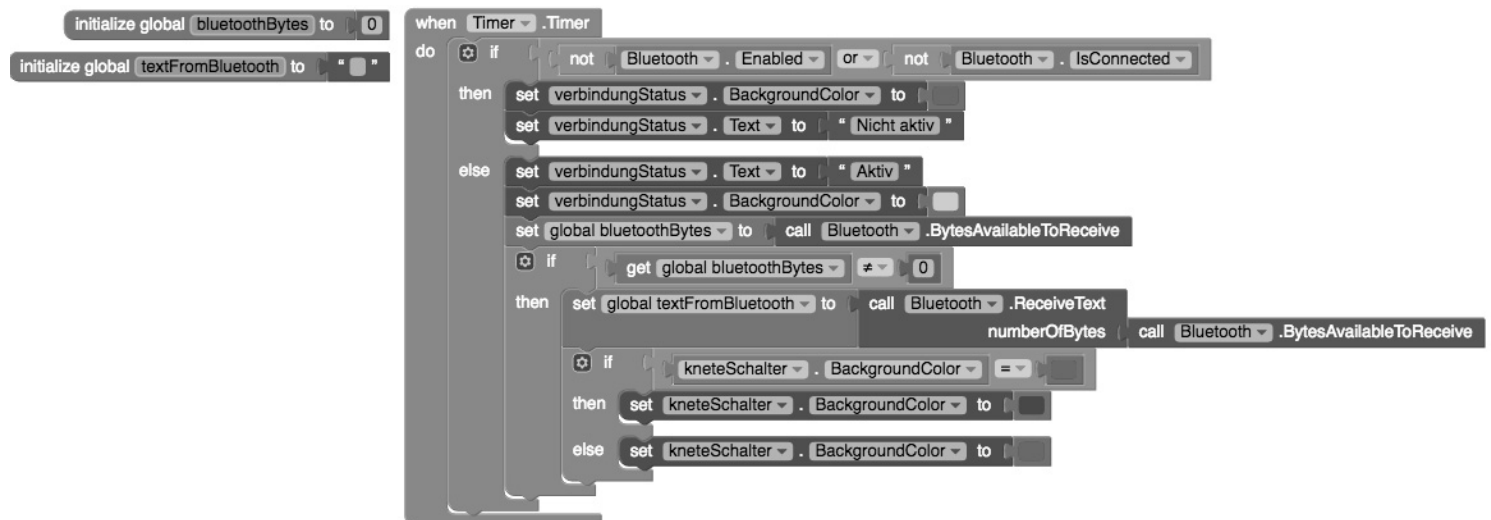


De if-blok ondersteunt ook else en else-if.

Klik dan op het element dat u wilt gebruiken en het onderdeel verandert al.



Opdat het gekleurde label goed zichtbaar is, werden breedte en hoogte op 30 pixels ingesteld.



Totnogtoe werd de timer alleen gebruikt om de status van de draadloze verbinding weer te geven.

Via `Bluetooth.ReceiveText` wordt de tekst via de draadloze interface gelezen.



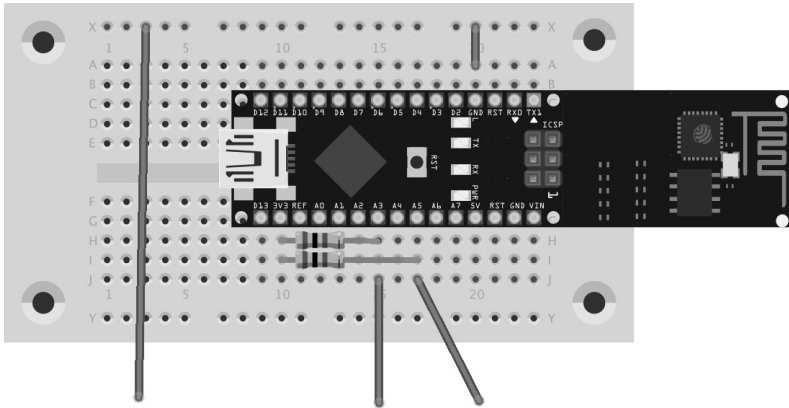
De te zetten kleur kunt u via een keuzeveld selecteren.

15. dag

15e dag

Vandaag in de adventskalender

• 1 x weerstand 20 MOhm



fritzing

Het uiteinde van drie uitstekende draadbruggen gaat telkens opnieuw in een stuk boetseerlei. Omwille van de twee boetseerleicontacten hebt u ook twee weerstanden met een hoge Ohm-waarde nodig.

Verschillende boetseerleicontacten

Nu gebruikt u twee boetseerleicontacten en toont u in een app, welke contacten werden ingedrukt.

Onderdelen: 1 x steekplatine, 3 x stuk boetseerlei, 2 x weerstand 20 MOhm, 4 x draadbruggen (verschillende lengtes)

De sketch

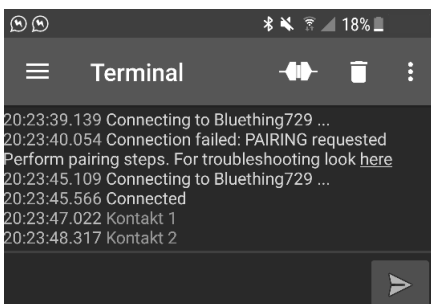
Het programma voor vandaag is Tag15.ino en bevindt zich in de directory Dag15. Zoals reeds op dag 14 wordt het indrukken van een boetseerleicontact via `analogRead` ingelezen. Voor elk boetseerleicontact is er een extra variabele. Via `HC05.print` wordt dan het commando aan de draadloze interface doorgegeven.

```
void loop() {
  sensorValue1 = analogRead(sensorPin1);
  sensorValue2 = analogRead(sensorPin2);

  if (sensorValue1 < 256) {
    HC05.print("Contact 1\n");
  }
  if (sensorValue2 < 256) {
    HC05.print("Contact 2\n");
  }
  if (sensorValue1 < 256 || sensorValue2 < 256) {
    delay(1000);
  }
}
```

De app

Voor de weergave van de teksten wordt de app **Serial Bluetooth Terminal** gebruikt. Maak eerst verbinding met het IoT-board en de teksten moeten verschijnen van zodra u een boetseerleicontact aanraakt.



Het aanraken van een boetseerleicontact leidt tot de geprogrammeerde uitvoer in de terminal.

16e dag

Vandaag in de adventskalender

• 1 x knipper-LED rood met voorweerstand

Knipper-LED via app sturen

Via een knop schakelt u de knipper-LED in en via een andere knop opnieuw uit.

Onderdelen: • 1 x steekplaatje, 1 x knipper-LED rood met voorweerstand

De sketch

Het programma voor vandaag is Tag16.ino en bevindt zich in de directory Dag16. De sketch reageert op Aan en Uit van de draadloze interface:

```
if (Text.startsWith("Aan") || Text.startsWith("AAN") || Text.startsWith("aan")){
  aan();
} else if (Text.startsWith("Uit") || Text.startsWith("UIT") || Text.startsWith("uit")){
  uit();
}
```

Het schakelen van de knipper-LED gebeurt in twee afzonderlijke functies. Daarbij wordt dan alleen de waarde LOW of HIGH via digitalWrite geschreven.

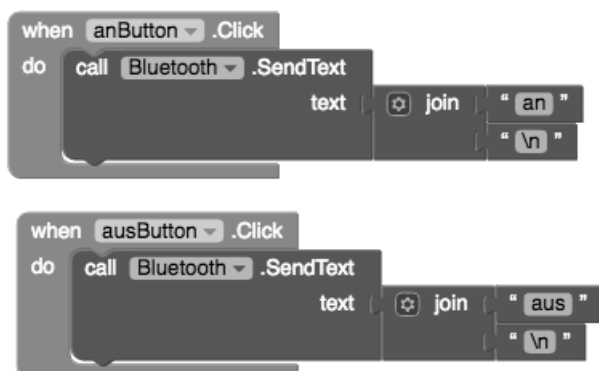
```
void aan() {
  Serial.println("Aan");
  digitalWrite(blinkPin, HIGH);
}

void uit() {
  Serial.println("Uit");
  digitalWrite(blinkPin, LOW);
}
```

De app

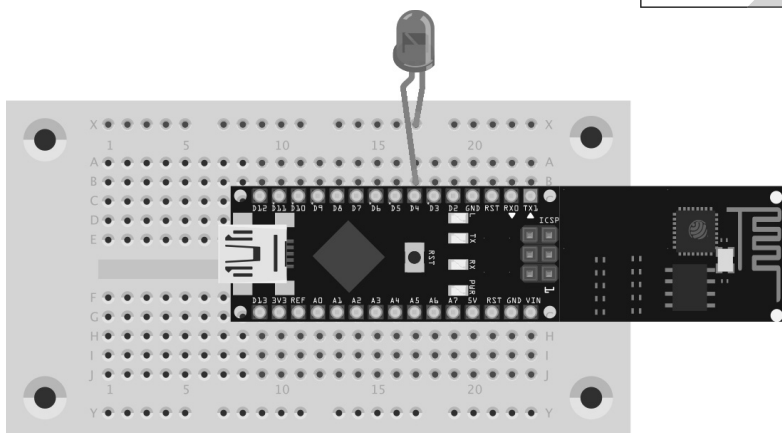
De app Blinker.aia heeft twee knoppen om de knipper-LED in en uit te schakelen.

Voor het verzenden wordt de reeds gekende methode call Bluetooth.SendText gebruikt.



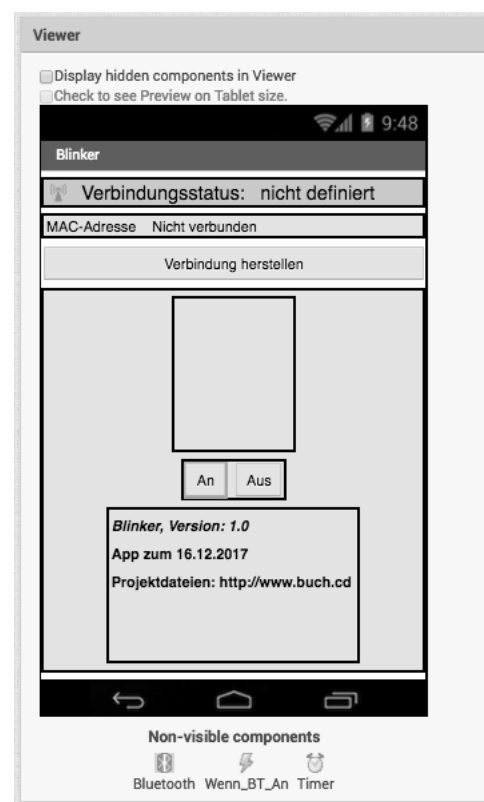
Denk eraan dat u bij het verzenden ook het eindcriterium aan de string hangt.

16. dag



fritzing

Ook de knipper-LED heeft een voorweerstand en daarom heeft deze geen bijkomende elementen voor de schakeling nodig.



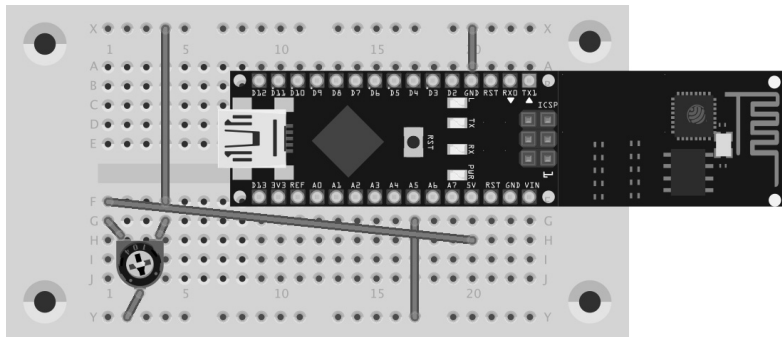
Boven beide knoppen (Aan en Uit) bevindt zich als positioneringselement een **VerticalArrangement**.

17. dag

17e dag

Vandaag in de adventskalender

• 1 x potentiometer



fritzing

Een potentiometer is van uit programmeerzicht niets anders dan een analoge sensor.

Weergave van de weerstandswaarde

Vandaag meet u de waarde van een potentiometer via een analoge ingang en voert u de waarde in een app uit.

Onderdelen: 1 x steekplatine, 1 x potentiometer van 15 kOhm, 4 x draadbruggen (verschillende lengtes)

De sketch

Het programma voor vandaag is Tag17.ino en bevindt zich in de directory Dag17. Eerst worden de twee variabelen `sensorPin` en `sensorValue` gedefinieerd. De potentiometer hangt aan de analoge ingang 5 en daarom wordt `sensorPin` op A5 gezet:

```
int sensorPin = A5;
int sensorValue = 0;
```

In de methode `loop` wordt de waarde via `analogRead` ingelezen en aan de draadloze interface doorgegeven. Daarna wordt 2000 ms gewacht en de waarde wordt opnieuw gelezen:

```
void loop() {
  sensorValue = analogRead(sensorPin);
  HC05.print("Waarde potentiometer = ");
  HC05.println(sensorValue);
  delay(2000);
}
```

De app

Voor de weergave van de teksten wordt de app **Serial Bluetooth Terminal** gebruikt. Maak eerst verbinding met het IoT-board en u ziet de waarden van de potentiometer.

18e dag

Vandaag in de adventskalender

• 1 x jumperkabel

RGB-looplicht

Via een app stuurt u de knipperduur van een RGB-LED. Daarbij knippert de LED in afwisselende kleuren. Het knippen kunt u daarbij ook opnieuw uitschakelen.

Onderdelen: 1 x steekplatine, 1 x RGB-LED met voorweerstand, 1 x jumperkabel

De sketch

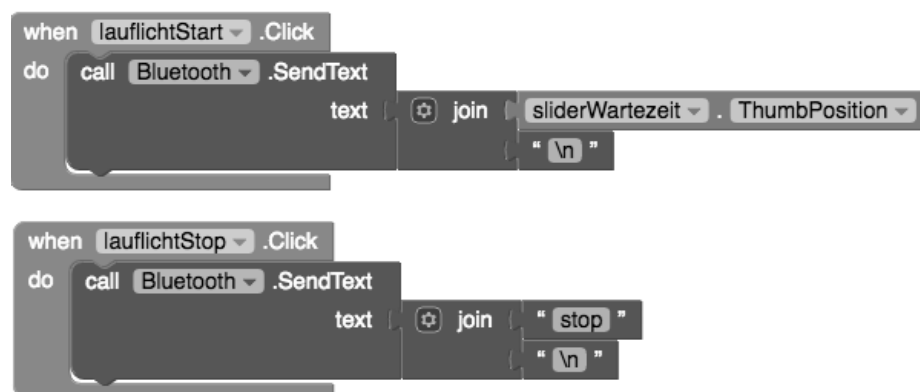
Het programma voor vandaag is Tag18.ino en bevindt zich in de directory Dag18. De sketch reageert op de tekst stop en op een verzonden cijfer. In geval van stop wordt de RGB-LED uitgeschakeld. Als een cijfer aankomt, wordt het als wachttijd tussen het in- en uitschakelen van een van de drie LED's van de RGB-LED gebruikt.

```
if (Tekn == '\n') {
  Serial.println(Text);
  if (Text.startsWith("stop")){
    uit();
    knipperDuur = 0;
  } else {
    knipperDuur = leesKnipperduur(Text);
  }
}
```

De app

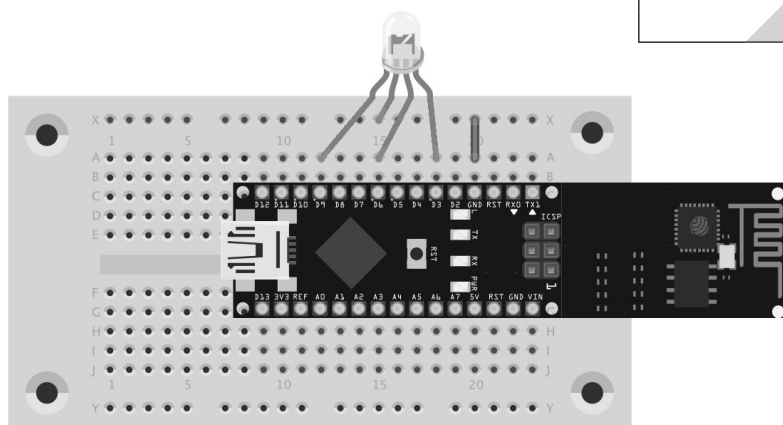
De app RGB_Looplicht.aia heeft een slider en twee knoppen voor het in- en uitschakelen van het knippen.

Via het reeds gekende commando Bluetooth.SendText worden de wachttijd en de tekst stop via de draadloze interface verzonden.



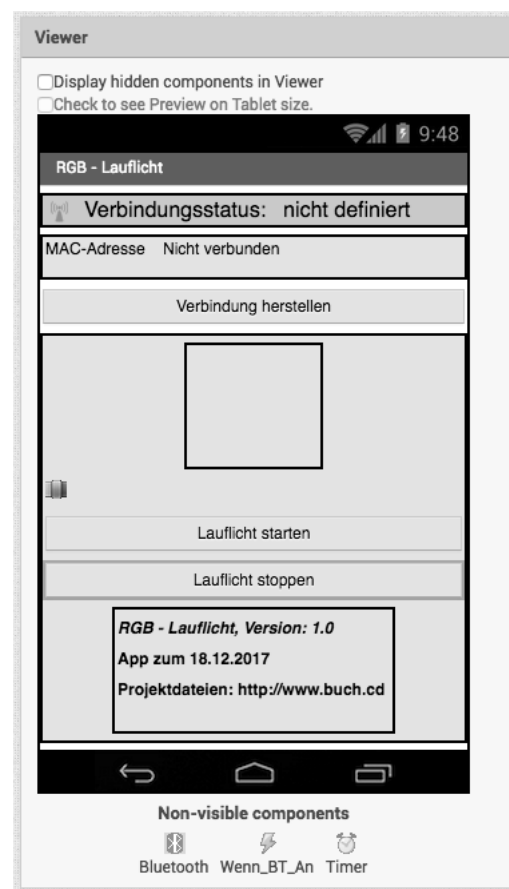
De wachttijd wordt niet extra gecodeerd, maar alleen de waarde van de slider, gevolgd door een afsluitende \n verzonden.

18. dag



fritzing

Door de geïntegreerde voorweerstanden is de schakeling zeer compact.



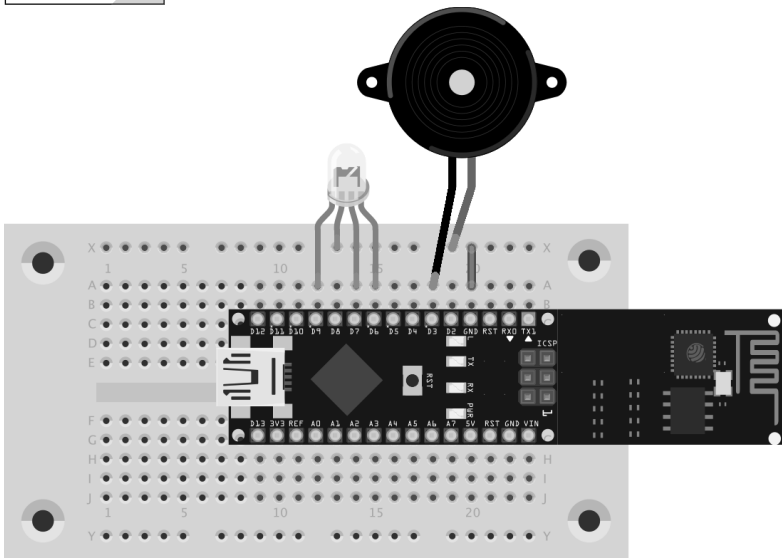
De minimale waarde van de slider is 0 en de maximale waarde is 2000. De waarden staan voor de wachttijd in milliseconden.

19. dag

19e dag

Vandaag in de adventskalender

• 1 x jumperkabel



fritzing

De RGB-LED heeft weer geen voorweerstanden nodig. De jumperkabel gebruikt u om de mas-saverbinding te maken;

app voor selectie van hardware-apps

Vandaag worden twee functies in een schakeling gecombineerd: Geluidswaergave met piezo en knipperen van een RGB-LED. Via de app kiest u de overeenkomstige functie.

Onderdelen: 1 x steekplaatje, 1 x RGB-LED met voorweerstand, 1 x piezo, 1 x jumperkabel

De sketch

Het programma voor vandaag is Tag19.ino en bevindt zich in de directory Dag19. Eerst moeten de juiste pins als variabele worden gedefinieerd:

```
//---- pins van RGB-LED
int redPin = 9;
int greenPin = 7;
int bluePin = 8;

//---- Pin Piezo
int piezo = 3;
```

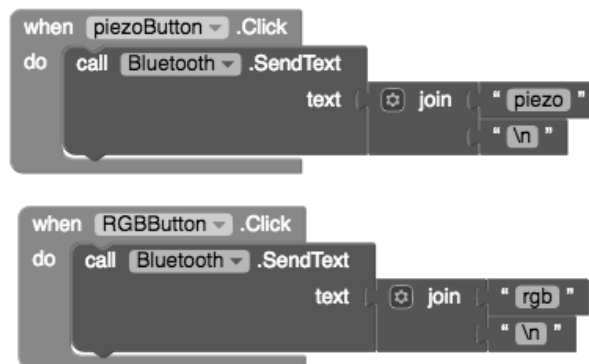
De sketch reageert op de commando's piezo en rgb. De methodes werden uit de voorgaande sketches overgenomen:

```
if (Text.startsWith("piezo")){
  uit();
  playMelody();
} else if (Text.startsWith("rgb")) {
  knipperen(500);
};
```

De app

De app App_Wisselaar.aia heeft twee knoppen om tussen piezo en RGB te wisselen.

De functie is in twee when.Click-blokken afgebeeld. Alleen de te verzenden tekst wordt aangepast.



De blokken zijn zeer gelijkaardig. Pas in zulke gevallen op dat u de juiste knop in het when.Click-blok kiest.



Deze app gebruikt de reeds gekende elementen uit de andere apps om de verbinding te maken. Via **Verbinding maken** maakt u een verbinding.

20e dag

Vandaag in de adventskalender

• 1 x NTC

Warmtesensor in de app

In het huidige project wordt een NTC voor de temperatuurmeting gebruikt. U kunt het project bv. goed testen door met de NTC dicht bij een kaars op de adventskrans te komen.

Onderdelen: 1 x steekplatine, 1 x NTC, 1 x potentiometer van 15 MOhm, 6 x draadbruggen (verschillende lengtes)

De sketch

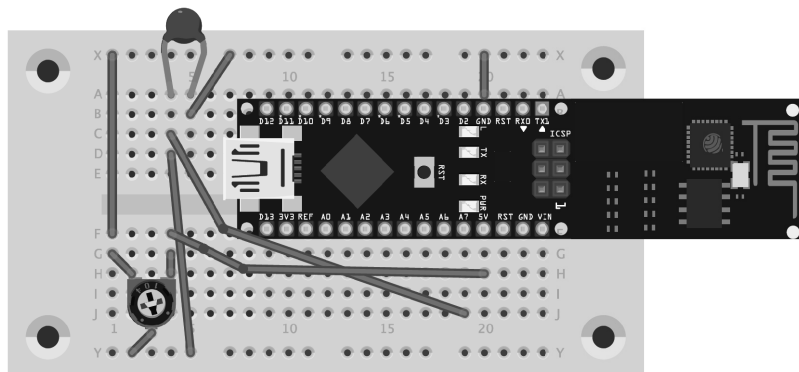
Het programma voor vandaag is `Tag20.ino` en bevindt zich in de directory `Dag20`. De sketch is zeer eenvoudig opgebouwd. Na oproep van `analogRead` wordt de waarde via Bluetooth naar de app verzonden, vervolgens wordt 2 seconden gewacht en dan opnieuw de waarde ingelezen en verzonden:

```
void loop() {
  sensorValue = analogRead(sensorPin);
  HC05.print("Waarde van de NTC = ");
  HC05.println(sensorValue);
  delay(2000);
}
```

De app

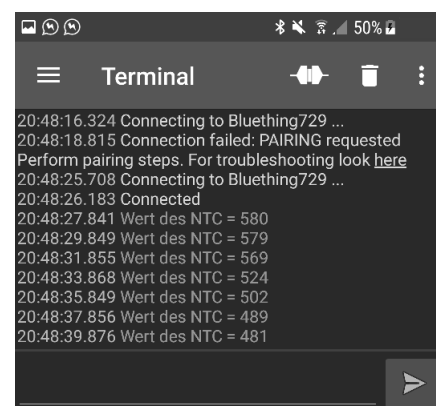
Voor de weergave van de teksten wordt de app **Serial Bluetooth Terminal** gebruikt. Maak eerst verbinding met het IoT-board en u ziet de waarden van de NTC. Wanneer de waarden kleiner worden is dit een teken voor stijgende temperatuur.

20. dag



fritzing

Vergeet niet de weerstand van 10 kOhm te gebruiken anders zult u niet tevreden zijn met de meetresultaten.



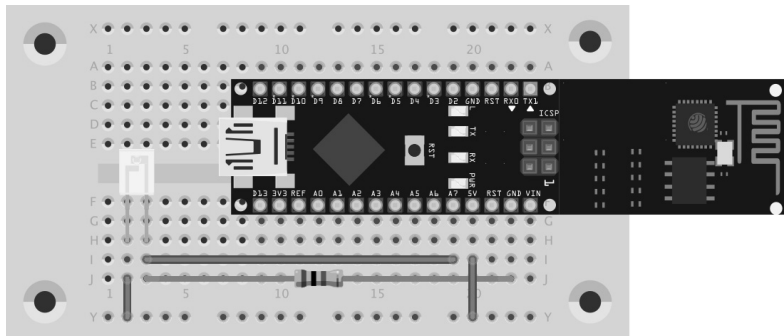
Zoals reeds in de andere projecten worden ontvangen gegevens in groene tekst weergegeven. Vergeet in de sketch niet om **delay** op te roepen anders krijgt u teveel waarden.

21. dag

21e dag

Vandaag in de adventskalender

- 1 x fototransistor
- 1 x weerstand 1 kOhm



Helderheid en duisternis in de app meten

Nu hebt u een bijkomende sensor voor uw projecten. Met de fototransistor kunt u helderheid en duisternis weergeven. Dit leert u vandaag. Met deze kennis kunt u snel een detectiecel bouwen en hebt u zo een alarminstallatie met weergave in een app.

Onderdelen: 1 x steekplatine, 1 x fototransistor, 1 x weerstand 1 kOhm, 3 x draadbruggen (verschillende lengtes)

fritzing

De opbouw is gelijkaardig aan dag 20. Voorzichtig: Er wordt een andere weerstand gebruikt en de verbinding met VCC (+5 V-) is belangrijk.

De sketch

Het programma voor vandaag is `Tag21.ino` en bevindt zich in de directory `Dag21`. De sketch lijkt op de sketch van de dag voordien. Als u meer waarden op uw smartphone wilt zien, moet u de wachtduur bij het oproepen van `delay` verminderen.

```
void loop() {
  sensorValue = analogRead(sensorPin);
  HC05.print(„Waarde van de fototransistor = „);
  HC05.println(sensorValue);
  delay(2000);
}
```

De app

Voor de weergave van de teksten wordt de app **Serial Bluetooth Terminal** gebruikt. Maak eerst verbinding met het IoT-board en u ziet de waarden van de fototransistor. Kijk hoe de waarden bij licht en duisternis veranderen;

22e dag

Vandaag in de adventskalender

- 1 x vochtigheidssensor
- 1 x weerstand 1 kOhm

Vochtigheidsmeter

Vandaag krijgt u nogmaals een sensor, een vochtigheidssensor. Daarmee kunt u de vochtigheid meten.

Onderdelen: 1 x steekplatine, 1 x vochtigheidssensor, 1 x weerstand 1 kOhm, 3 x draadbruggen (verschillende lengtes)

De sketch

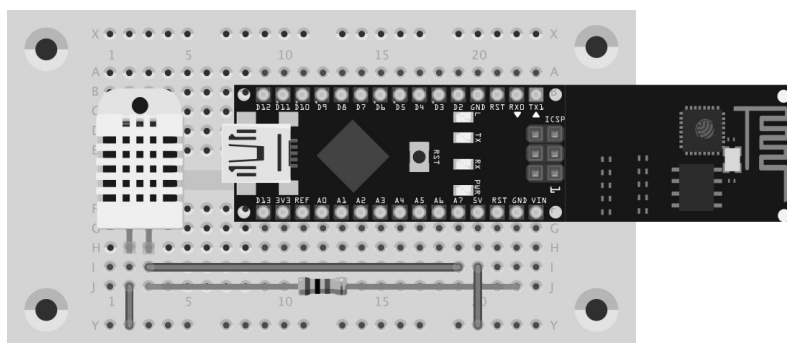
Het programma voor vandaag is `Tag22.ino` en bevindt zich in de directory `Dag22`. Zoals reeds de voorbije dagen het geval was, wordt de sensor via `analogRead` uitgelezen en de waarde aan de draadloze interface doorgegeven:

```
void loop() {
  sensorValue = analogRead(sensorPin);
  HC05.print(„Waarde van de vochtigheidssensor = „);
  HC05.println(sensorValue);
  delay(2000);
}
```

De app

Voor de weergave van de teksten wordt de app **Serial Bluetooth Terminal** gebruikt. Maak eerst verbinding met het IoT-board en u ziet de waarden van de vochtigheidssensor. Neem de vochtigheidssensor tussen twee vingers en kijk naar de ontwikkeling van de waarden.

22. dag



fritzing

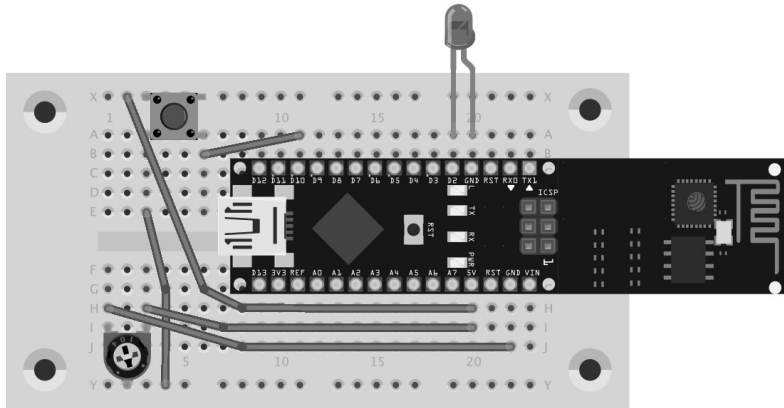
De schakeling gelijkt op de schakeling van dag 21. De sensor moet slechts worden vervangen.

23. dag

23e dag

Vandaag in de adventskalender

• 1 x jumperkabel



Codebreker

Het huidige project is een klein spel: In een app stelt u een code in. Deze code geeft u aan het IoT-board door. Nu moet de code met de knop worden ingegeven. Wanneer de code correct werd ingegeven, brandt de rode LED en in de app wordt weergegeven dat de code correct werd ingegeven. Anders brandt de rode LED niet en in de app wordt weergegeven dat de code verkeerd werd ingegeven.

Onderdelen: 1 x steekplatine, 1 x knop, 1 x LED rood met voorweerstand, 1 x potentiometer van 15 kOhm, 5 x draadbruggen (verschillende lengtes)

fritzing

Opdat de knop ook goed werkt, wordt een spanningsdeler via de potentiometer opgebouwd. In plaats van de draadbruggen kunt u ook de jumperkabels gebruiken.

De sketch

Het programma voor vandaag is Tag23.ino en bevindt zich in de directory Dag23. De code wordt in de vorm van C++ bepaald:

```
if (Text.startsWith("C")){
    aantalKnoppen= Text.substring(Text.
    indexOf("C")+1).toInt();
}
```

In totaal heeft de speler vijf seconden tijd om de code in te voeren.

```
interval = millis() - start;
Serial.println(interval);
while (interval < 5000) {
    pressed += knopAnalyseren();
    delay(100);
    interval = millis() - start;
}
```

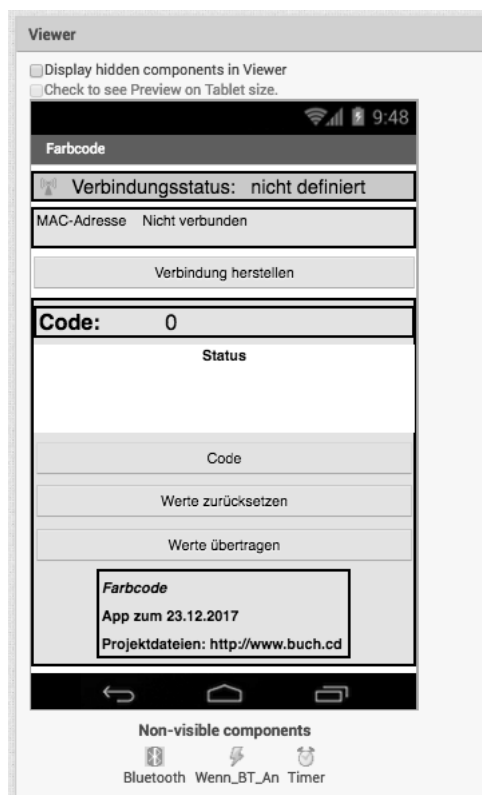
Via de methode knopAnalyseren wordt gecontroleerd of op een knop werd gedrukt. Tussen de controles wordt 100 ms gewacht. Het resultaat wordt aan de app teruggegeven.

```
if (pressed == aantalKnoppen) {
    HC05.print("JA");
    digitalWrite(redPin, HIGH);
    delay(5000);
    digitalWrite(redPin, LOW);
} else {
    HC05.print("NEEN");
}
```

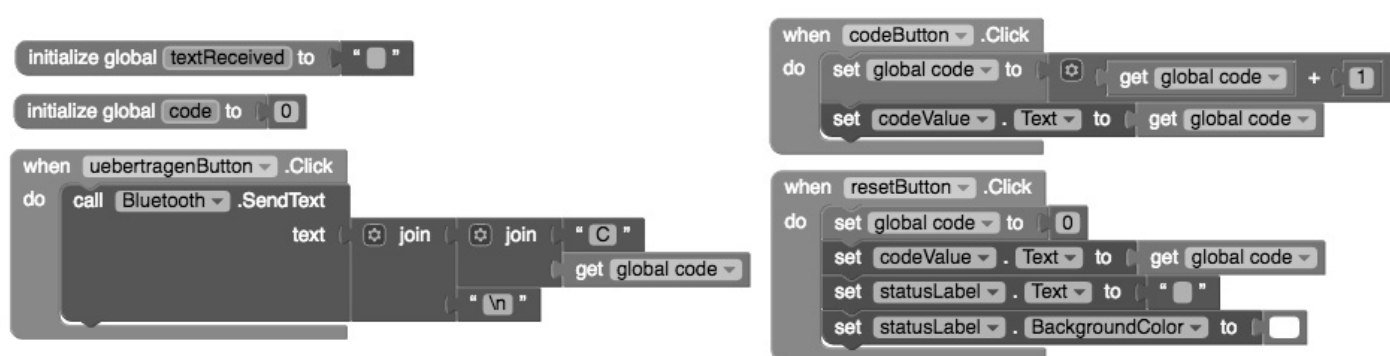
De app

De app Kleurcode.aia heeft drie knoppen en twee labels. Met het ene label wordt de ingevoerde code weergegeven en met het tweede label wordt de speelstatus weergegeven. Als de code correct via de hardwareknop werd ingevoerd, wordt de achtergrondkleur van het label voor de speelstatus op groen gezet, anders op rood.

De code wordt in een globale variabele tussenopgeslagen en dan met Bluetooth.SendText bepaald.

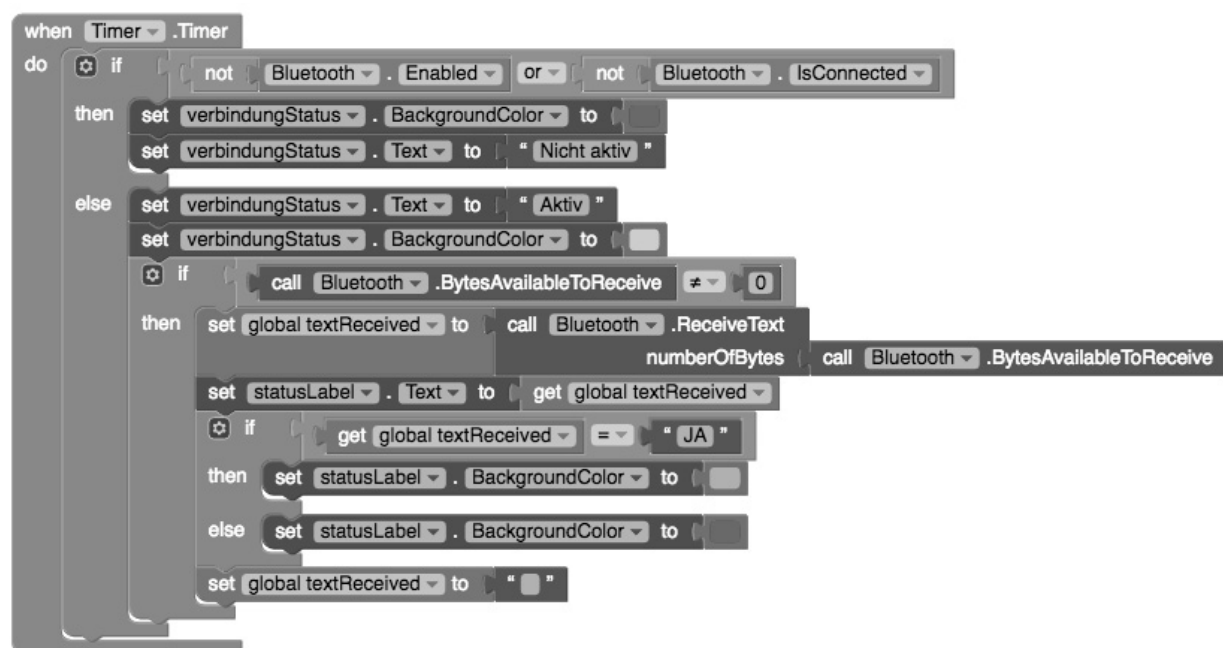


Via een gekleurd label wordt weergegeven of de code via de hardwareknop al of niet correct werd ingevoerd.



Aangezien drie knoppen voor het spel nodig zijn, zijn er ook drie when.Click-blokken.

Het ontvangen van teksten van het IoT-block gebeurt in de periodieke timer. Hierbij wordt de timer gebruikt die reeds voor de weergave van de verbindingstatus wordt gebruikt.



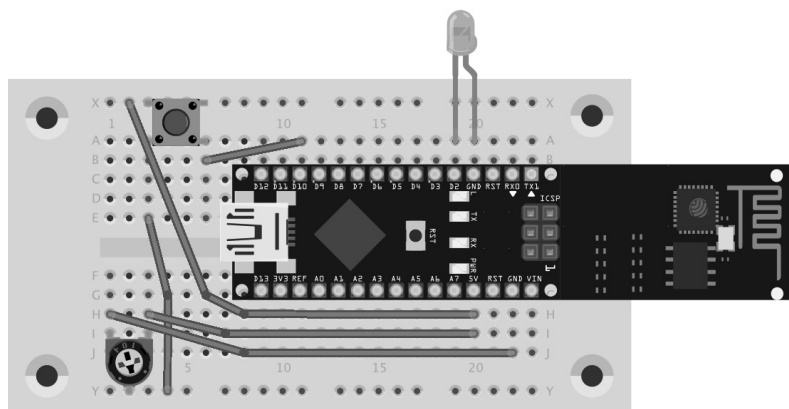
Bij ontvangst van JA wordt de achtergrondkleur op groen gezet, anders op rood.

24. dag

24e dag

Vandaag in de adventskalender

• 1 x knop



De huidige schakeling lijkt op de schakeling van de dag voordien. De rode LED werd slechts door een groene LED vervangen.

Reactiespel

Tot slot van de adventskalender is er nog een klein reactiespel. Via een knop in de app start u het spel. De LED op het steekplatine gaat aan en zodra de LED uit gaat, moet u op de knop drukken; De duur wordt in de app weergegeven.

Onderdelen: 1 x steekplatine, 1 x knop, 1 x LED groen met voorweerstand, 1 x potentiometer van 15 kOhm, 5 x draadbruggen (verschillende lengtes)

De sketch

Het programma voor vandaag is Tag24.ino en bevindt zich in de directory Dag24. Via de functie millis wordt de tijdsduur gemeten tot op de knop wordt gedrukt:

```
if (Text.startsWith("START")){

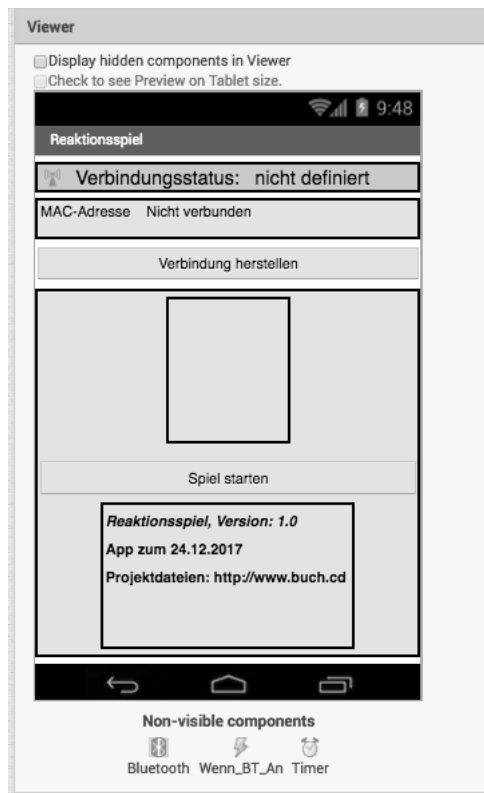
    digitalWrite(greenPin, HIGH);
    delay(1000);
    digitalWrite(greenPin, LOW);

    //Starttijd
    int duur = millis();

    Serial.println(interval);
    while (knopAnalyseren() != 1) {
        delay(100);
    }
    duur = millis() - duur;
    HC05.print(duur);
    digitalWrite(greenPin, LOW);
}
```

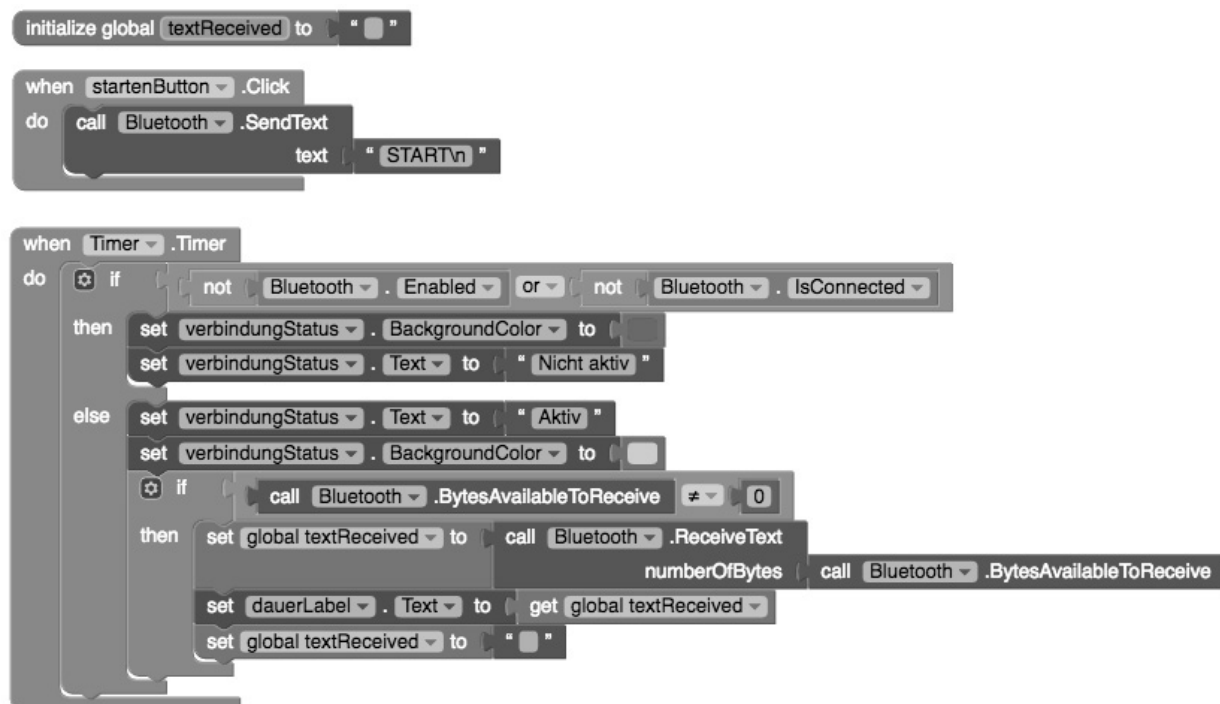
De app

De app Reactiespel.aia heeft een knop met het opschrift **Spel starten**. Bij het indrukken van de knop wordt de tekst via de draadloze interface verzonden. De inkomende tekst van het IoT-board wordt in het label boven de knop weergegeven.



Boven de knop **Spel starten** ligt een label zonder tekst. Pas wanneer een tekst van het IoT-board aankomt, wordt de tekst in het label gezet.

Om het spel te starten wordt bij een klik op de knop **Spel starten**, het blok `when startButton.click` geanalyseerd en de tekst `START` via de draadloze interface bepaald.



Het analyseren van de draadloze interface moet altijd in een timer gebeuren. Want men weet niet precies wanneer een datapakket aankomt.

Zalig kerstmis!