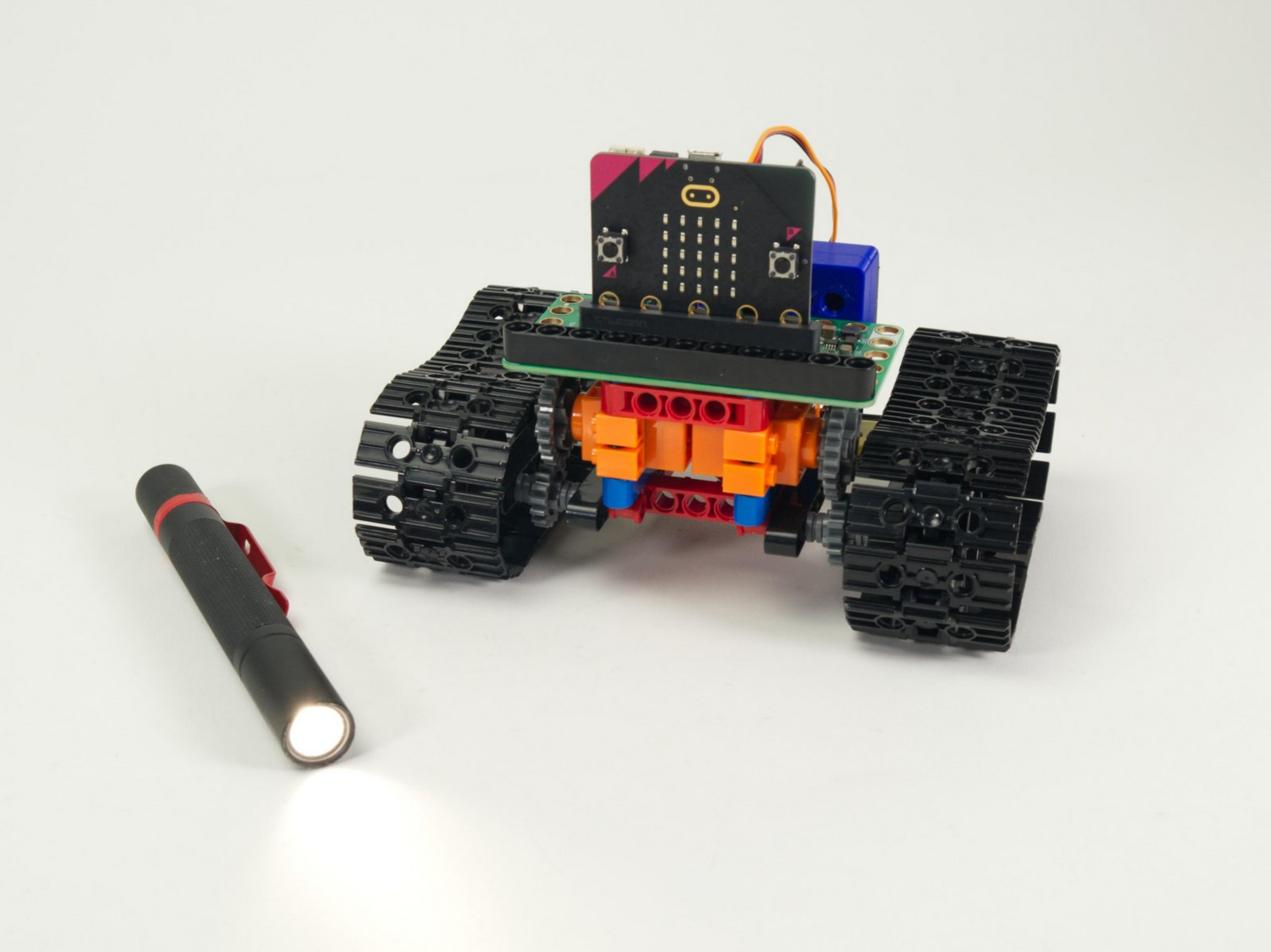
Tank - Light Activated

Remotely control your Tank with a flashlight! Use the built-in light sensor on the micro:bit to make your Tank move around.

Written By: Pete Prodoehl



This document was generated on 2023-08-28 07:31:01 AM (MST).

INTRODUCTION

The Bit Board Tank can take advantage of the built-in sensors found on the micro:bit, and we can use the light sensor to "remotely control" the Tank with a flashlight.

We'll start with simple code to make the Tank roll away from you when you shine a flashlight at it, and then use more complex code to get the Tank turning as well as rolling away.

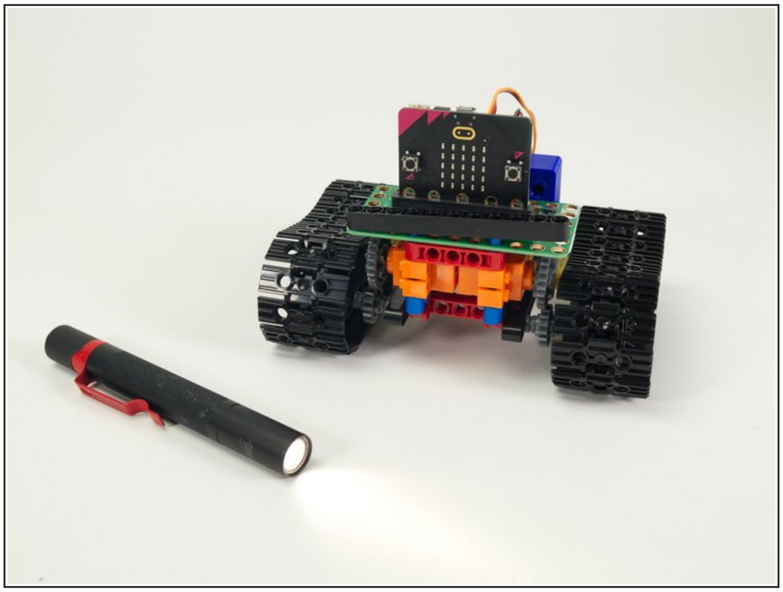
TOOLS:

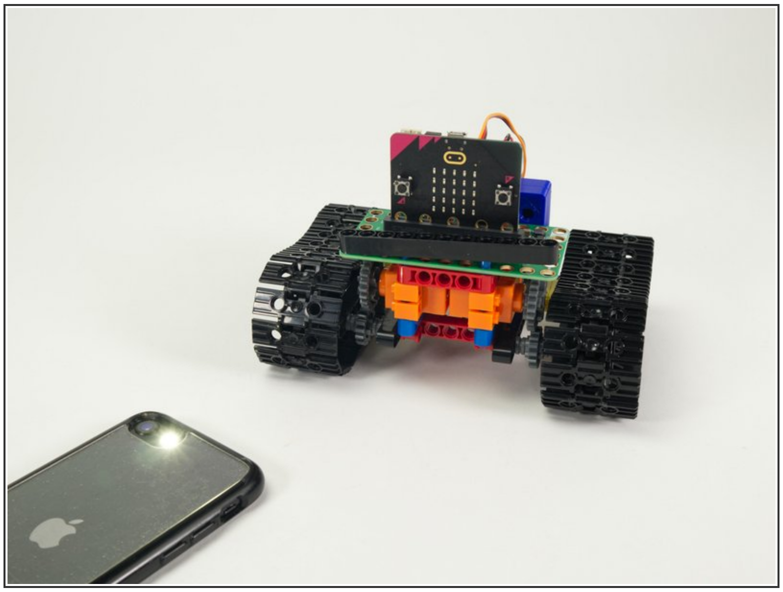
- Computer (1)

PARTS:

- Bit Board Rover Kit (1)
- Flashlight (1)

Step 1 — Prepare Your Tank

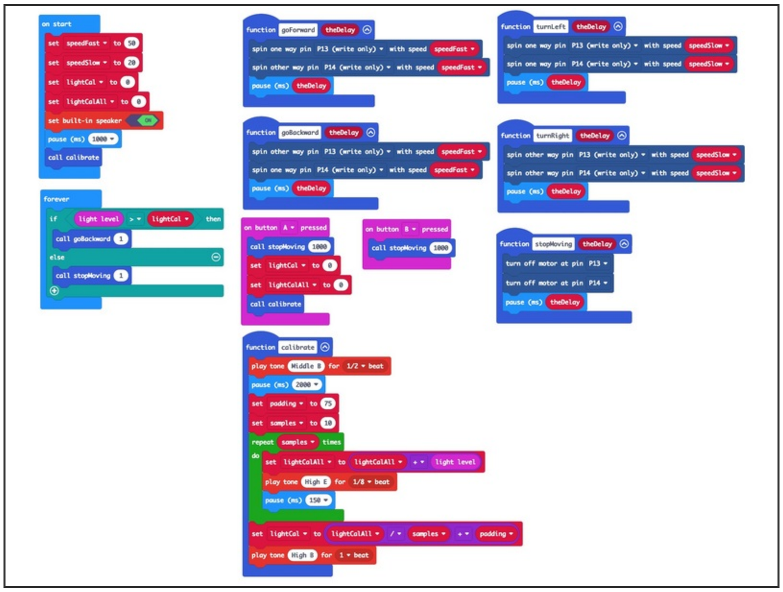




- For this guide you'll need a completed Tank Main Body.
- You'll also need a flashlight you can shine at the Tank to make it move.
- A nice powerful/bright flashlight is preferable, but you can use the "flashlight" feature of a phone in a pinch. You'll just need to get closer.

Our Sensor Showcase covers using the sensors built in to the micro:bit. The light sensor is covered in Step 10.

Step 2 — Load the Code



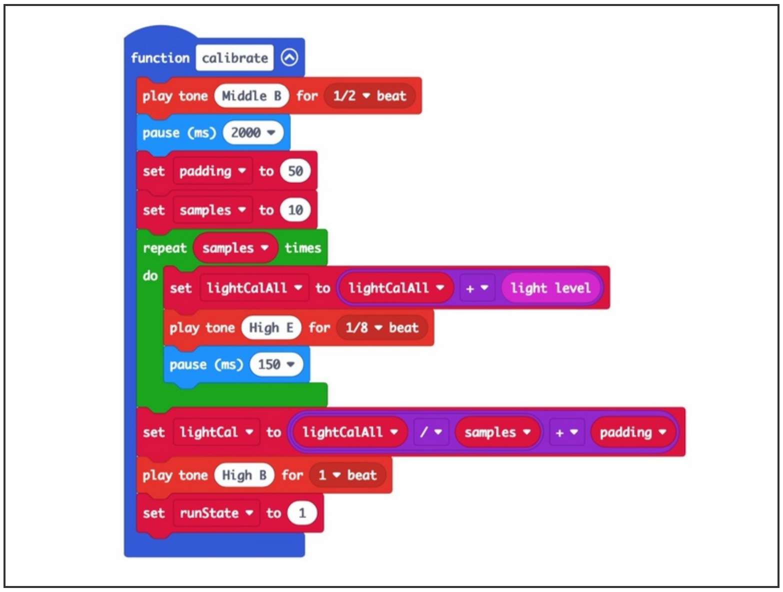
```
on start
    set speedFast to 50
    set speedSlow to 20
    set lightCal to 0
    set lightCalAll to 0
    set built-in speaker ON
    pause (ms) 1000
    call calibrate

forever
    if light level >= lightCal then
        call goBackward 1
    else
        call stopMoving 1

function goForward theDelay
    spin one way pin P13 (write only) with speed speedFast
    spin other way pin P14 (write only) with speed speedFast
    pause (ms) theDelay

function goBackward theDelay
    spin other way pin P13 (write only) with speed speedFast
    spin one way pin P14 (write only) with speed speedFast
    pause (ms) theDelay

on button A pressed
    call stopMoving 1000
    set lightCal to 0
    set lightCalAll to 0
    call calibrate

on button B pressed
    call stopMoving 1000

function calibrate
    play tone Middle B for 1/2 beat
    pause (ms) 2000
    set padding to 75
    set samples to 10
    repeat samples times
    do
        set lightCalAll to lightCalAll + light level
        play tone High E for 1/8 beat
        pause (ms) 150
    set lightCal to lightCalAll / samples + padding
    play tone High B for 1 beat

function turnLeft theDelay
    spin one way pin P13 (write only) with speed speedSlow
    spin one way pin P14 (write only) with speed speedSlow
    pause (ms) theDelay

function turnRight theDelay
    spin other way pin P13 (write only) with speed speedSlow
    spin other way pin P14 (write only) with speed speedSlow
    pause (ms) theDelay

function stopMoving theDelay
    turn off motor at pin P13
    turn off motor at pin P14
    pause (ms) theDelay
```

⚠ If you've never used a micro:bit before you'll want to check out this guide: Bit Board V2 Setup and Use

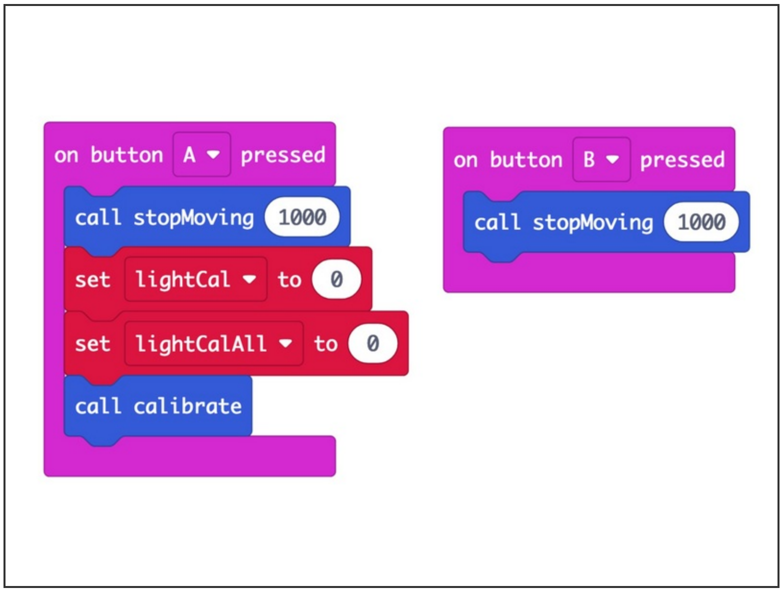
- Note: This **Simple** version of the code is a great starting point for this project, but the **Turner** version in **Step 5** is more fun. This code is easier to understand and the Turner code is a bit more complex.
- We're going to load the following code for our **Tank Light Detecting Simple** program: https://makecode.microbit.org/_JTVdArHwb...
- When you power on the Tank it will run a calibration routine to check the light levels in the room. (We'll cover the calibration in **Step 3**.)
- Once the calibration is done the Tank will sit there waiting for the light levels to go above the threshold that was set in the calibration routine...
- When you shine a flashlight on the face of the micro:bit it will detect the higher light level and the Tank will start rolling away from you!

This document was generated on 2023-08-28 07:31:01 AM (MST).

Step 3 — Calibration



```
function calibrate
  play tone Middle B for 1/2 beat
  pause (ms) 2000
  set padding to 50
  set samples to 10
  repeat samples times
  do
    set lightCalAll to lightCalAll + light level
    play tone High E for 1/8 beat
    pause (ms) 150
  set lightCal to lightCalAll / samples + padding
  play tone High B for 1 beat
  set runState to 1
```

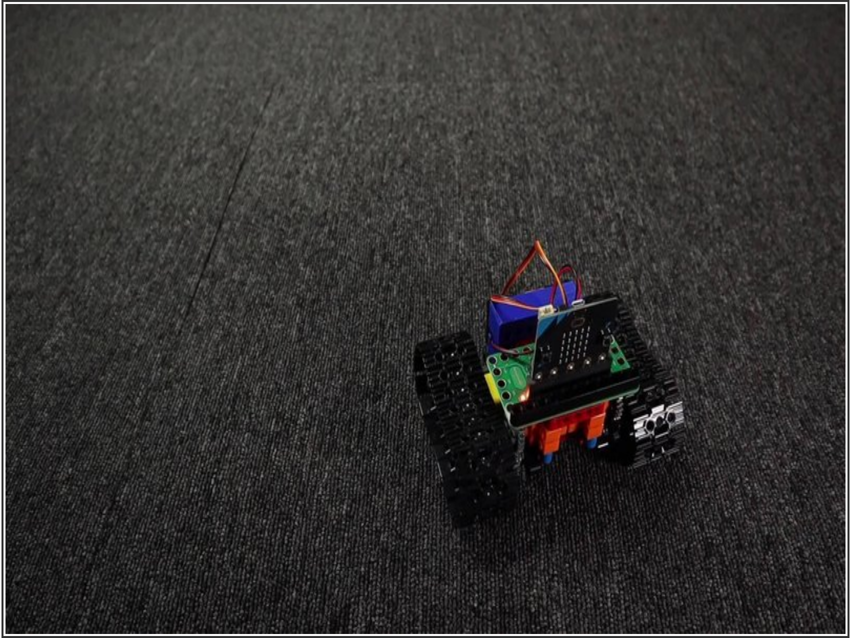


```
on button A pressed
  call stopMoving 1000
  set lightCal to 0
  set lightCalAll to 0
  call calibrate

on button B pressed
  call stopMoving 1000
```

- We need to calibrate the light levels in the room before the micro:bit can tell if a light is shining on it.
- The light level can be anywhere between **0 and 255**. (0 is almost no light and 255 is very bright light.)
- Our code takes 10 readings (with a slight pause between each reading) adds them all together and the divides by the number of samples (10 in this case) to get our final value.
  - Calibration routines often use this **sampling** technique where a number of values are captured and then the *average* of them is used.
- Once we've calculated our ambient light level we pad the number a bit to prevent false triggering. We used **75** for our pad value in this example but you can experiment with lower (or higher) values.

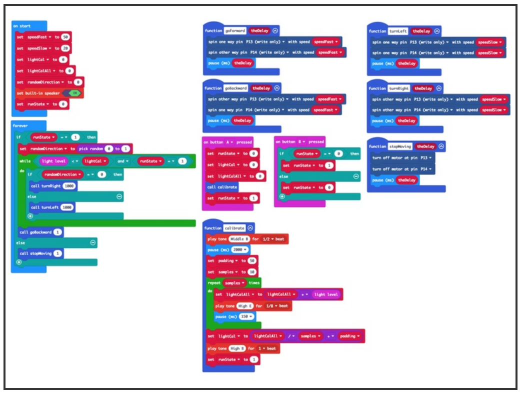
The calibration runs automatically when you power on the Tank but you can also run it by pressing the **A** Button on the micro:bit if you need to recalibrate.

Step 4 — Test it Out!

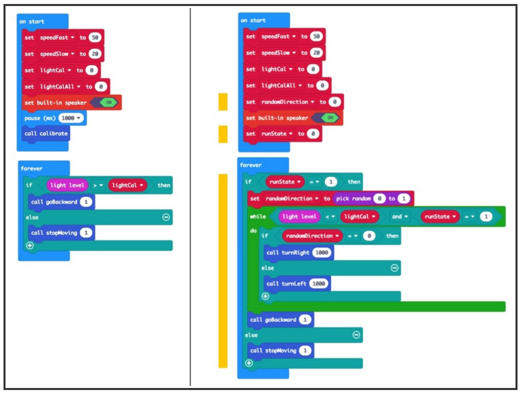


- Power on the Tank, wait for the calibration to complete, and then shine a flashlight at the micro:bit on the Tank.
- The Tank should roll away from the light!
- You can make the Tank stop by moving the flashlight away so it is not shining on the micro:bit
- Alternately the Tank will stop rolling when it gets too far away and the light level drops due to the further distance of the flashlight.

Step 5 — Load New Code

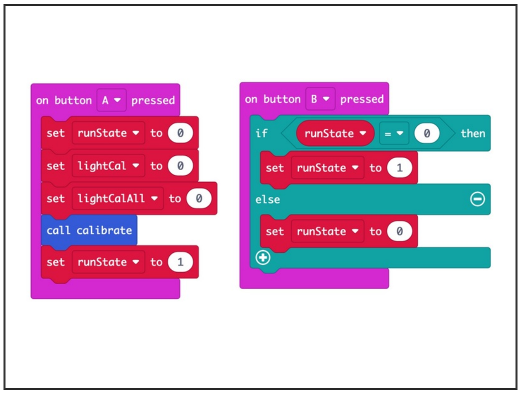


```
on start
    set speedFast to 50
    set speedSlow to 20
    set lightCal to 0
    set lightCalAll to 0
    set randomDirection to 0
    set built-in speaker ON
    set runState to 0

forever
    if runState = 1 then
        set randomDirection to pick random 0 to 1
        while light level < lightCal and runState = 1
        do
            if randomDirection = 0 then
                call turnRight 1000
            else
                call turnLeft 1000
        call goBackward 1
    else
        call stopMoving 1

function goForward theDelay
    spin one way pin P13 (write only) with speed speedFast
    spin other way pin P14 (write only) with speed speedFast
    pause (ms) theDelay

function turnLeft theDelay
    spin one way pin P13 (write only) with speed speedSlow
    spin one way pin P14 (write only) with speed speedSlow
    pause (ms) theDelay

function goBackward theDelay
    spin other way pin P13 (write only) with speed speedFast
    spin one way pin P14 (write only) with speed speedFast
    pause (ms) theDelay

function turnRight theDelay
    spin other way pin P13 (write only) with speed speedSlow
    spin other way pin P14 (write only) with speed speedSlow
    pause (ms) theDelay

on button A pressed
    set runState to 0
    set lightCal to 0
    set lightCalAll to 0
    call calibrate
    set runState to 1

on button B pressed
    if runState = 0 then
        set runState to 1
    else
        set runState to 0

function stopMoving theDelay
    turn off motor at pin P13
    turn off motor at pin P14
    pause (ms) theDelay

function calibrate
    play tone Middle B for 1/2 beat
    pause (ms) 2000
    set padding to 50
    set samples to 10
    repeat samples times
    do
        set lightCalAll to lightCalAll + light level
        play tone High E for 1/8 beat
        pause (ms) 150
    set lightCal to lightCalAll / samples + padding
    play tone High B for 1 beat
    set runState to 1
```



```
on start
    set speedFast to 50
    set speedSlow to 20
    set lightCal to 0
    set lightCalAll to 0
    set built-in speaker ON
    pause (ms) 1000
    call calibrate

forever
    if light level > lightCal then
        call goBackward 1
    else
        call stopMoving 1
```

```
on start
    set speedFast to 50
    set speedSlow to 20
    set lightCal to 0
    set lightCalAll to 0
    set randomDirection to 0
    set built-in speaker ON
    set runState to 0

forever
    if runState = 1 then
        set randomDirection to pick random 0 to 1
        while light level < lightCal and runState = 1
        do
            if randomDirection = 0 then
                call turnRight 1000
            else
                call turnLeft 1000
        call goBackward 1
    else
        call stopMoving 1
```

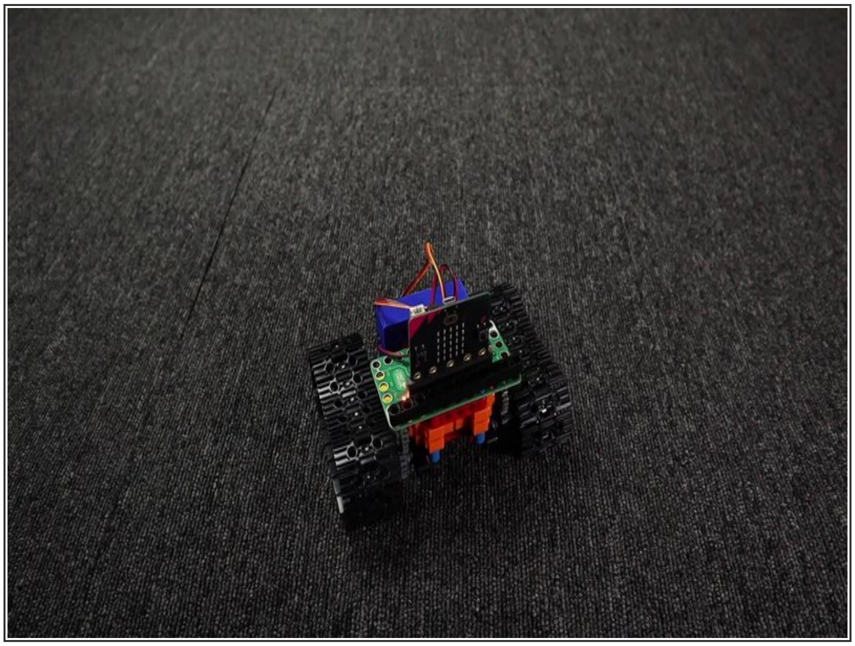


```
on button A pressed
    set runState to 0
    set lightCal to 0
    set lightCalAll to 0
    call calibrate
    set runState to 1

on button B pressed
    if runState = 0 then
        set runState to 1
    else
        set runState to 0
```

- This **Turner** version of the code adds a few features, the most important being that the Tank will no longer sit still.
- When the Tank does not detect the flashlight it will turn in a circle. You can wait for the Tank to turn a certain amount and then shine the flashlight on it to get it rolling again.
- Load the following code for our **Tank Light Detecting Turner** program: https://makecode.microbit.org/_EaVRemHJR...
- We've added a few neat things to the code.
    - The Tank may turn clockwise or anti-clockwise when not rolling. There's a function in the code to randomly choose which way to turn.
    - We've also added a runState variable that allows you to start/stop the Tank from moving by using the **B** button on the micro:bit to toggle the state. One press starts the Tank, the next press stops it.

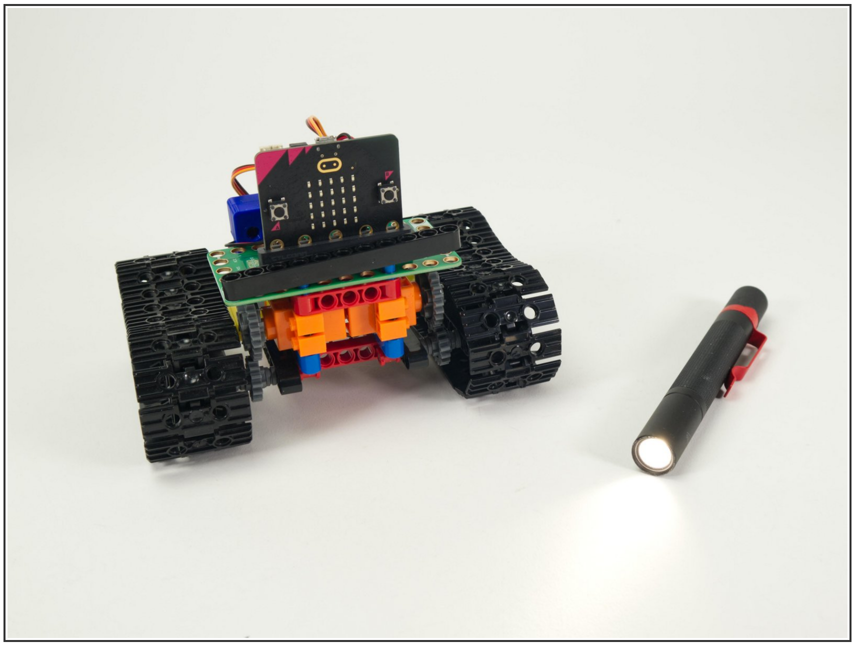
This document was generated on 2023-08-28 07:31:01 AM (MST).

Step 6 — Test it Out!



- Power on the Tank, wait for the calibration to complete, and then shine a flashlight at the micro:bit on the Tank.
- The Tank should start spinning around and then move away from the light when you shine your flashlight on the micro:bit
- When you move your flashlight away from the Tank it will stop moving away and start to spin again.

Step 7 — Take it Further



- The code provided (both the Simple and Turner versions) can serve as a starting point for your own ideas

This document was generated on 2023-08-28 07:31:01 AM (MST).

about how the Tank can react to light.

- You could program a sequence of movements and then try to control them with a flashlight.

This document was generated on 2023-08-28 07:31:01 AM (MST).