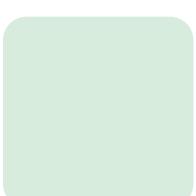
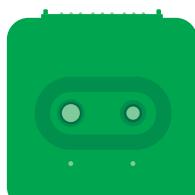
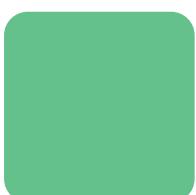
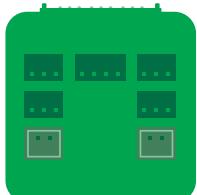
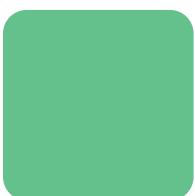
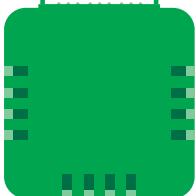
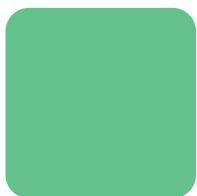
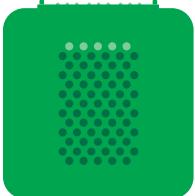
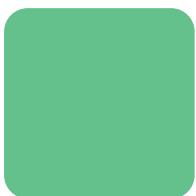


ArTeC® Links

アーテックリンクス

応用セット

教員用





デジタル砂時計を作成しよう

想定人数	2人
想定コマ数	2~4コマ

学習内容

- 加速度センサーについて
- スピーカーについて

達成目標

- 加速度センサーの仕組みの理解
- 出力装置についての理解

活動内容	評価
<h2>導入</h2> <ul style="list-style-type: none"> ● 加速度センサー……………P.2 ○ 加速度センサーの仕組み ○ 加速度センサーブロックの使い方 ● スピーカー……………P.4 ○ スピーカーの仕組み ○ スピーカーブロックの使い方 	知識及び技能① 知識及び技能①
<h2>基本課題① デジタル砂時計を作成しよう</h2> <ul style="list-style-type: none"> ● デジタル砂時計を作成しよう……………P.5 ○ プログラム内容の説明 ○ フローチャートの作成 ○ プログラムの作成 ○ プログラムの実行 ○ 振り返り 	知識及び技能②
<h2>発展課題</h2> <ul style="list-style-type: none"> ● デジタル砂時計をもっと使いやすくしよう……………P.9 ○ アイデアを考える ○ フローチャートの作成 ○ プログラムの作成 ○ プログラムの実行 ○ 発表 	知識及び技能② 思考・判断・表現 学向力・人間性

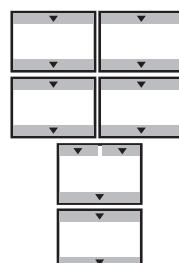


アイコンシート ご活用ください！

生徒がカードを並べて課題解決のヒントにできる補助教材です。

プログラムの

- ① アイデアを整理できる
- ② 流れを考えられる
- ③ 段階的に考えられる



無料
ダウンロード



応用

デジタル砂時計を作成しよう

年 組 番 名前

知識技能①



加速度センサー

加速度センサーについて

この拡張ユニットは加速度を計測することができます。

「センサーがどちらの向きにかたむいているのか?」「速さがどちらの向きに変化しているのか?」をセンサーで計測します。

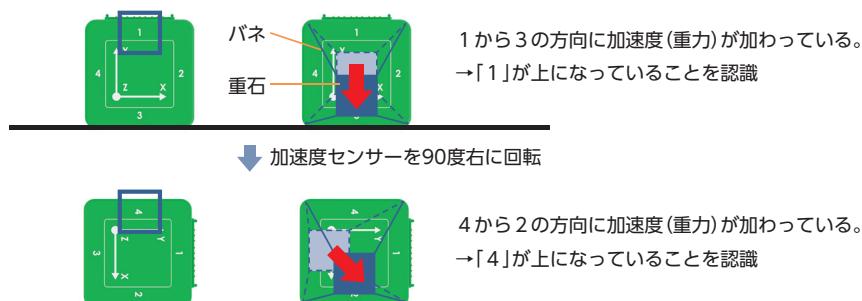
身近な製品では、ゲームのコントローラーやスマートフォンなどで使われています。



加速度センサーの仕組み

加速度センサーは、センサーに対してどの方向から加速度が加わっているかを調べることで、センサーの向きを検知しています。

加速度センサー内部イメージ



加速度センサーブロックの使い方

加速度センサーは、センサーに対してどの方向から加速度が加わっているかを調べることで、センサーの向きを検知しています。

ゆさぶられている ▾

このブロックでは、加速度センサーがどのような動きをしているかを計測することができます。

このブロックをクリックすると、右のような表示になります。この表示を変更することで、計測する条件を任意に設定することができます。

※加速度センサーを傾けて、ユニットボーダーの数値がどのように変化するのかを確認させましょう。

加速度センサーを上下左右に大きく動かし、X, Y, Z のどの数字が変化しているのかに注目するようにしましょう。

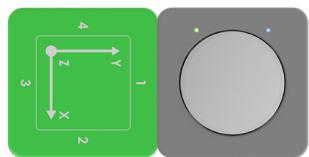


1

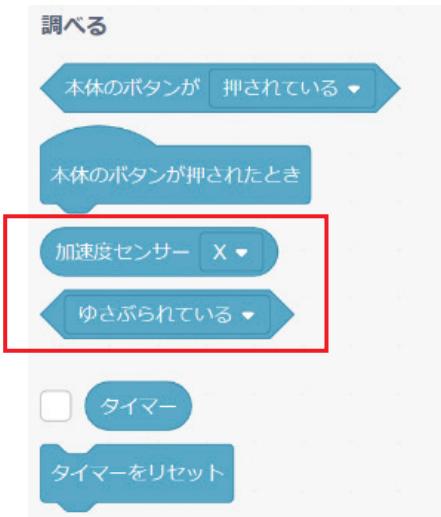
【センサーブロックの表示方法】

(1) メインユニットと加速度センサーチップをつなげ、メインユニットとパソコンを接続します。

メインユニットとセンサーチップをつなげる場所はどこでも構いません。



(2) 接続完了後、カテゴリー **調べる** に以下のようにブロックが表示されていることを確認してください。



※「加速度センサーx」と「ゆさぶられている」ブロックが追加されていない場合

以下の方法でもブロックを表示させることができます。

①タブから「編集」を選択します。

②「ブロックの表示・非表示」を選択します。

③「加速度センサーブロック」を選択します。

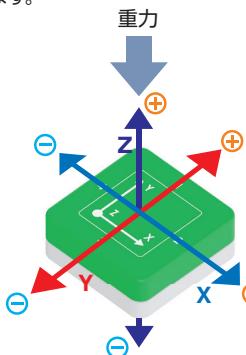
この方法でブロックを「調べる」カテゴリーに表示させることができます。

非表示にしたい場合も同様の手順を行ってください。また、ほかのセンサーブロックについても同様です。

加速度センサー X ▾

このブロックでは、加速度センサーの数値を調べることができます。加速度センサーの数値は、ユニットボードの赤枠内から読み取ることができます。

X, Y, Z の数値はセンサーに加わる加速度（重力）が下のイラストのどの方向に向いているかを表しています。数値は、-1.00 ~ 1.00 の間で変化します。



ユニットボード	
メインユニット	▼
ボタン	OFF
拡張ユニット	▼
加速度センサー X	0.07
加速度センサー Y	-0.06
加速度センサー Z	-0.99

センサーが図の向きの場合、センサーに対して加速度（重力）が Z の $-$ 方向に加わるので、「加速度センサー Z」の数字は、ほぼ -1.00 になっています。



スピーカー

スピーカーについて

この拡張ユニットは、音を鳴らすことができます。

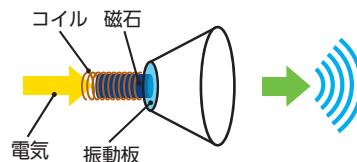
スピーカーに電気が流れると、中に入っている金属の板（振動板）が震えて音が鳴ります。この仕組みはイヤホンやヘッドホンなどに使われています。



スピーカーの仕組み

スピーカーは電気を流すことでも中に入っているコイルが磁力を発生させ、この磁力で振動板を震わせて音を出します。

磁力の強さの違いによって振動板の震え方を変えることができ、音の高低や強弱を調整することができます。



スピーカーブロックの使い方

【スピーカーブロックの表示方法】

- (1) メインユニットとスピーカーをつなげ、メインユニットとパソコンを接続します。
- (2) 接続完了後、カテゴリー 音 に以下のようなブロックが表示されていることを確認してください。

音

スピーカーから 60 を鳴らす

スピーカーから 60 を 1 秒間鳴らす

スピーカーを止める

【スピーカーブロックの使い方】

スピーカーから 64 を鳴らす

このブロックでは、スピーカーから音を鳴らすプログラムを実行することができます。数字の部分を変更することで、音の高低を変更することができます。

スピーカーから 60 を 1 秒間鳴らす

このブロックでは、指定した秒数だけスピーカーから音を鳴らすプログラムを実行することができます。

スピーカーを止める

このブロックでは、スピーカーから鳴っている音を止めるプログラムを実行することができます。



デジタル砂時計を作成しよう

AnTeC Links 応用セット ① デジタル砂時計を作成しよう
アーテックリンクス

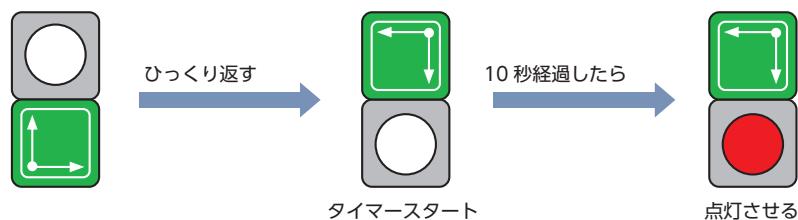


知識技能②

デジタル砂時計を作成しよう

【課題】

加速度センサーの向きを変えるとタイマーがスタートする、デジタル砂時計のプログラムを作成しましょう。



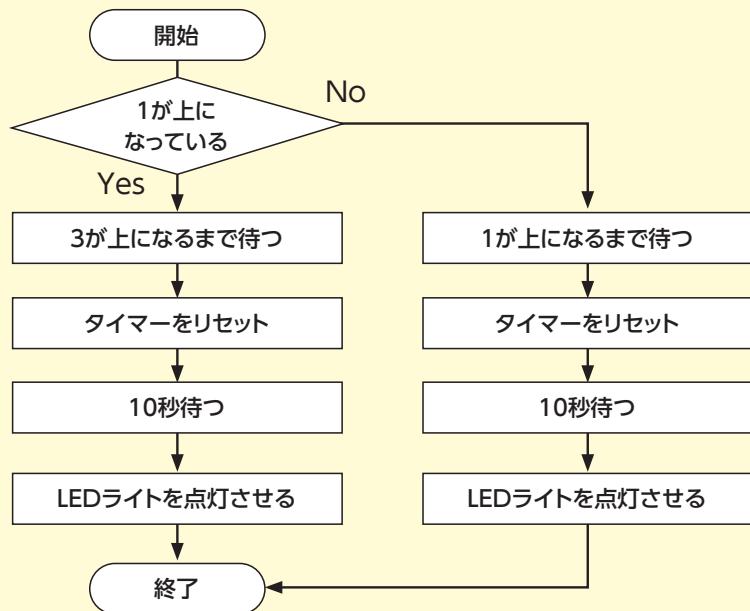
①フローチャートを作成しましょう。「アイコンシート」も参考にして作成しましょう。

グループ学習

「アイコンシート」を並べられる広い場所で行ってください。

知識技能②

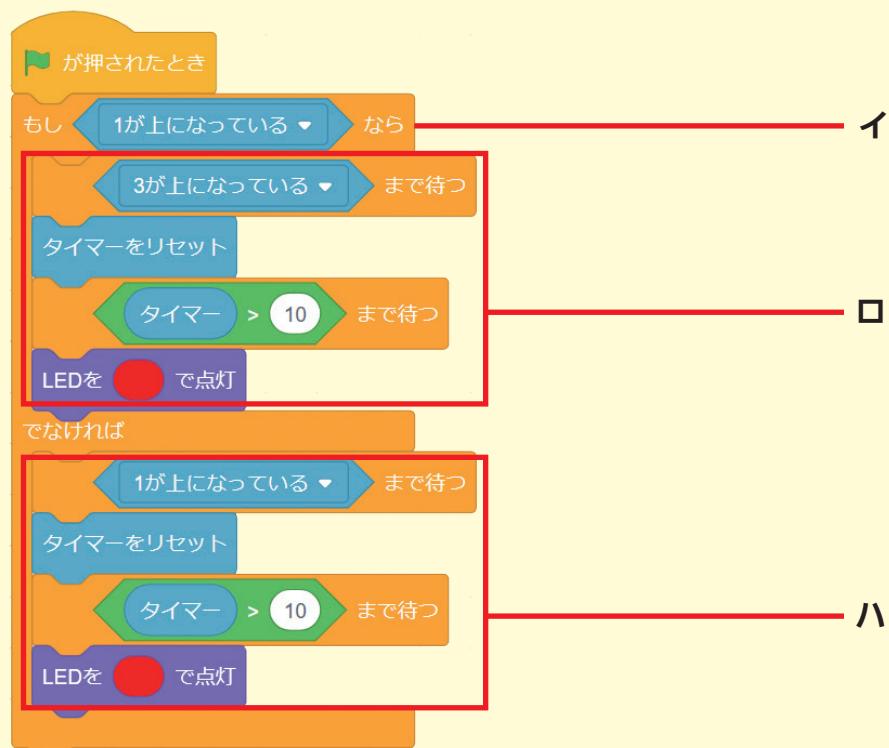
フローチャート例



② 「アイコンシート」と P.5 で作成したフローチャートを参考にしてプログラムを作成しましょう。.....

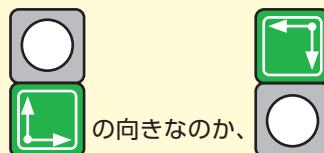
プログラミング実習

プログラム例



解説

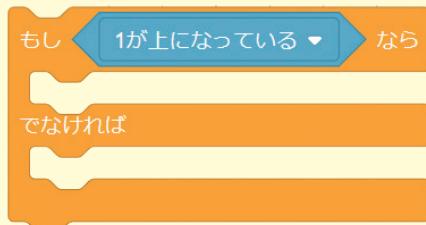
イ) 加速度センサーがどの向きを向いているかを判定するためのプログラム



まず、デジタル砂時計が

の向きなのか、

の向きなのかを計測しなければなりません。最初の向きが認識されていないと、ひっくり返されたことをプログラム上で認識できないためです。そのために以下のプログラムを作成します。



「加速度センサーの数字の 1 が上側にあるかどうか」という条件で、条件分岐を行います。

もし「加速度センサーの数字の 1 が上側にある」(= True) なら、



この向き

もし「加速度センサーの数字の 1 が上側にない」(= False) なら、



この向きと認識することができます。

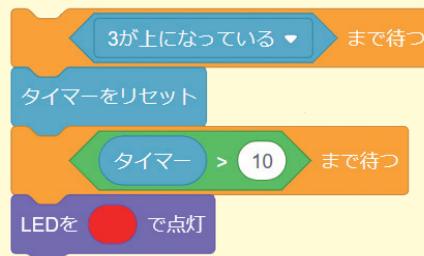
ロ) デジタル砂時計がひっくり返されたときにタイマーをスタートさせるプログラム

まず、デジタル砂時計がひっくり返されたことを認識するためのプログラムを作成します。



このブロックを使って、デジタル砂時計がひっくり返される（加速度センサーの向きが「1 が上の状態」から「3 が上の状態」になる）まで、プログラムを一時停止します。

ひっくり返されたことを計測した後、タイマーをリセットします。10 秒カウントして、LED ライトを赤色で点灯させます。



ハ) (ロ) と同様に、ひっくり返されたときにタイマーをスタートさせるプログラム

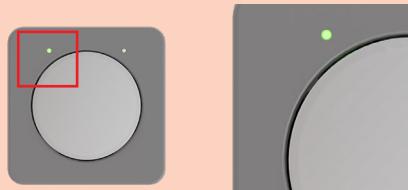
知識技能②

③プログラムが完成したら、プログラムを実行しましょう。

加速度センサーをひっくり返し、10秒後にメインユニットのLEDライトが点灯することを確認してください。

※ USBケーブルを外して、プログラムを実行させたい場合、プログラムの転送を行ってください。

- ①タブから「転送」を選択します。
- ②「接続」を選択し、ユニットボードの表示が消えたら転送完了です。
- ③USBケーブルを外し、バッテリーを接続してください。
- ④バッテリーの電源をオンにし、下図の赤枠のLEDが光っていることを確認します。
- ⑤加速度センサーをひっくり返して、正しくプログラムが実行されるか確認してください。



知識技能②

④達成できた項目にチェックを入れましょう。

- 加速度センサーの仕組みについて理解できた
- スピーカーの仕組みについて理解できた
- デジタル砂時計を作成することができた

③プログラムが完成したら、転送を行い、プログラムを実行しましょう。

加速度センサーをひっくり返し、10秒後にメインユニットのLEDライトが点灯することを確認しましょう

④達成できた項目にチェックを入れましょう。

- 加速度センサーの仕組みについて理解できた
- スピーカーの仕組みについて理解できた
- デジタル砂時計を作成することができた

応用

デジタル砂時計を作成しよう

年 組 番 名前

思考・判断・表現

デジタル砂時計をもっと使いやすくしよう

学向力・人間性①

【課題】

デジタル砂時計をキッチンタイマーとして使いたい場合、どのような工夫をすればもっと使いやすくなるかを考えましょう。



グループ学習
「アイコンシート」を並べられる広い場所で行ってください。

① 「アイコンシート」を使ってプログラムの流れを考え、どのような工夫ができるかを考えましょう。またその工夫によって、どのような課題を解決することができるかを以下の表に書き入れましょう。

課題点	LED ライトが光るだけでは気づかない場合がある。
工夫	スピーカーを使って音が鳴るようにした。

課題点	手がふさがっていると、タイマーをひっくり返すことができない。
工夫	人感センサーを使って、手をかざしたときに、タイマーをスタートできるようにした。

課題点	10 秒など、決まった時間のタイマーしか設定できない。
工夫	ボタンを押した回数ごとに、設定するタイマーの秒数を変更する。

人感センサーについて

人感センサーの詳細な説明は次章の P.16 で行っています。そちらを参考にしてください。

「アイコンシート」にも人感センサーの具体的な動作が記載されているので、そちらも参考にしてください。

知識技能②

学向力・人間性②

②P.5で書いた「課題点と工夫」から1つ選んで、プログラムを作成しましょう。プログラム完成後、プログラム実行しましょう。「アイコンシート」でつくったプログラムの流れも参考にしましょう。

③作成したプログラムをみんなに発表しましょう。

プログラミング
実習

【アイデア名】

【アイデア概要】

【課題点】

【課題に対して工夫した点】



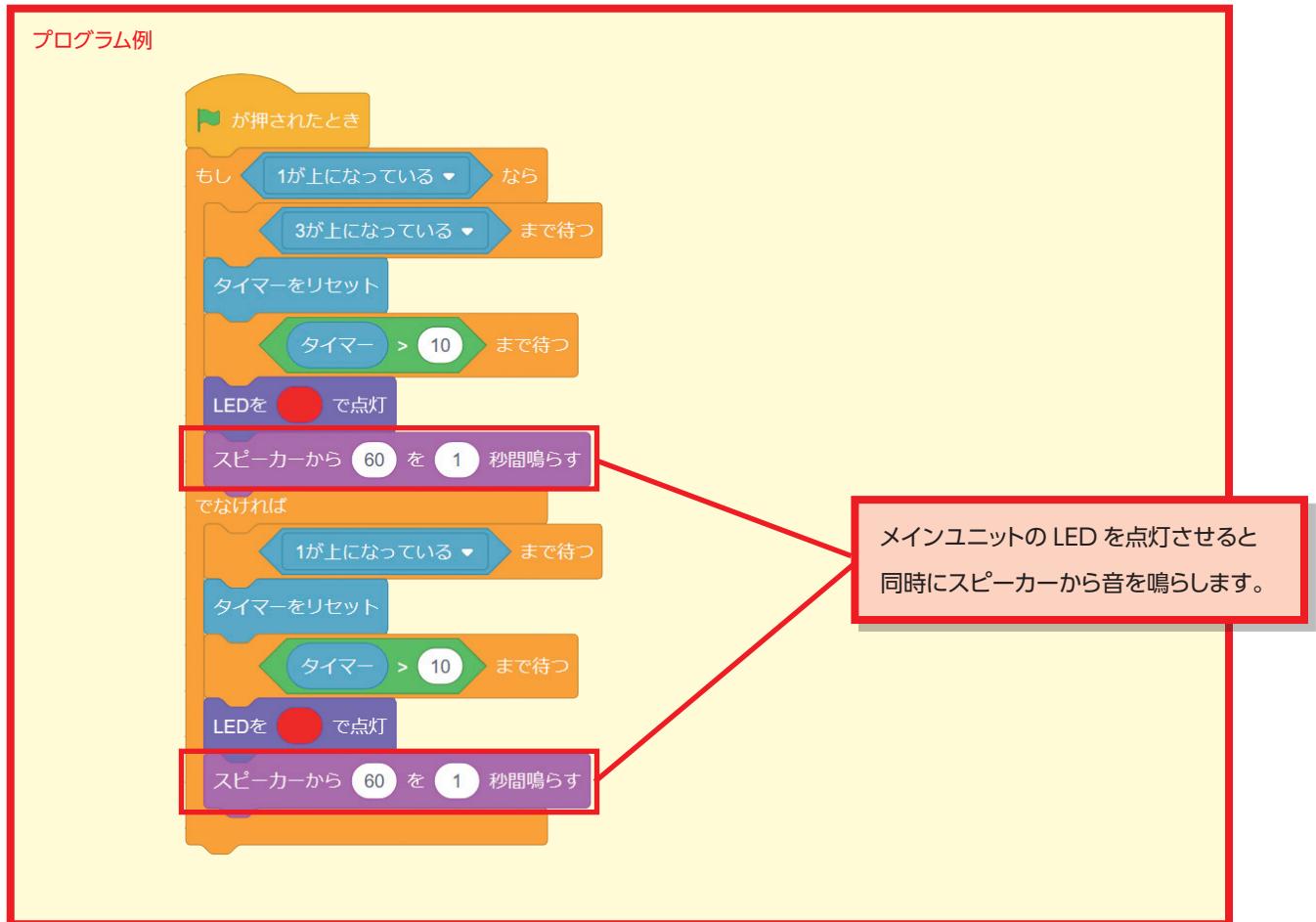
【今後解決したい改善点】

〔解説〕

例 1

課題点：LED ライトが光るだけでは気づかない場合がある。

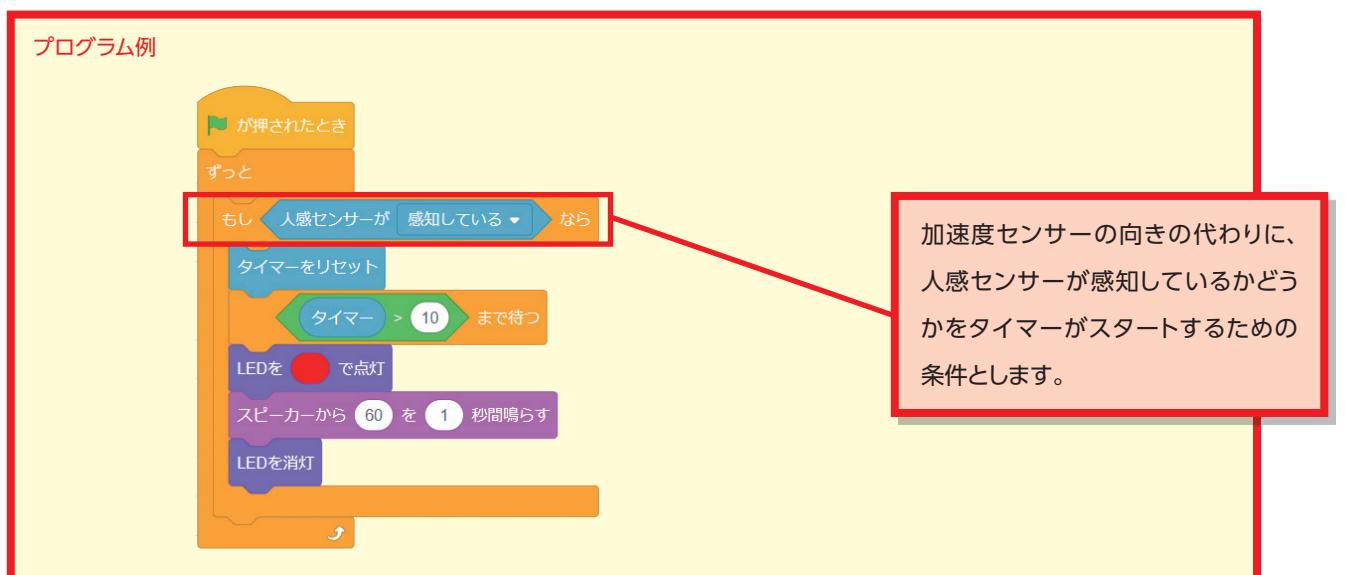
工 夫：スピーカーを使って音が鳴るようにした。



例 2

課題点：手がふさがっていると、タイマーをひっくり返すことができない。

工夫：人感センサーを使って、手をかざしたときに、タイマーをスタートできるようにした。



例 3

課題点：10秒など、決まった時間のタイマーしか設定できない。
工夫：ボタンを押した回数ごとに、設定するタイマーの秒数を変更する。

プログラム例

```
when green flag clicked
    [本体のボタンが押されたとき]
        [押された回数 ▾ を 0 にする]
        [5 秒待つ]
        [もし 押された回数 = 1 なら]
            [タイマーをリセット]
            [タイマー > 10 まで待つ]
            [LEDを (red) で点灯]
            [スピーカーから 60 を 1 秒間鳴らす]
            [LEDを消灯]
        [もし 押された回数 = 2 なら]
            [タイマーをリセット]
            [タイマー > 5 まで待つ]
            [LEDを (yellow) で点灯]
            [スピーカーから 60 を 1 秒間鳴らす]
            [LEDを消灯]
    end
end
```

(イ) 変数でボタンが押された回数を計測するプログラム

(ロ) ボタンが押された回数によって、計測する秒数を変更するプログラム

イ) 変数でボタンが押された回数を計測するプログラム

ボタンが押されるごとに変数の値が 1 つずつ変わるプログラムを作成します。



次にプログラムが実行されるたびに変数の値を 0 に戻さなければなりません。前回の変数の値がそのまま残っていると、ボタンが押された回数を正しく計測できないためです。

また「5秒待つ」のプログラムを入れることで、ユーザーがボタンを押す時間を確保します。

このプログラムがない場合、プログラムの実行と一緒に条件分岐が行われてしまうため、ボタンが押された回数によって、計測する時間を変更することができません。



MEMO

MEMO



防犯ライトを作成しよう

想定人数	2人
想定コマ数	2~4コマ

学習内容 ●人感センサーについて

達成目標 ●人感センサーの仕組みの理解

活動内容	評価
導入 ●人感センサー ○人感センサーの仕組み ○人感センサー ブロックの使い方	P.16 知識及び技能①
基本課題 ●防犯ライトの作成 ○プログラム内容の説明 ○フローチャートの作成 ○プログラムの作成 ○プログラムの実行 ○振り返り	P.18 知識及び技能②
発展課題 ●学校の防犯システムを考えよう ○アイデアを考える ○フローチャートの作成 ○プログラムの作成 ○プログラムの実行 ○発表	P.21 知識及び技能② 思考・判断・表現 学向力・人間性

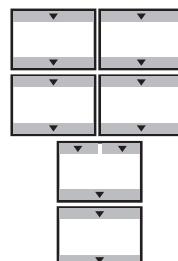


アイコンシート ご活用ください！

生徒がカードを並べて課題解決のヒントにできる補助教材です。

プログラムの

- ①アイデアを整理できる
- ②流れを考えられる
- ③段階的に考えられる



無料
ダウンロード



アイコンシート：ダウンロード URL : <https://www.artec-kk.co.jp/arteclinks/school/juniorhigh/>

応用

防犯ライトを作成しよう

年 組 番 名前



人感センサー

人感センサーについて

この拡張ユニットは、センサーで赤外線を計測することで、人やものの動きを計測することができます。

身近な製品では、自動ドアなどに使われています。

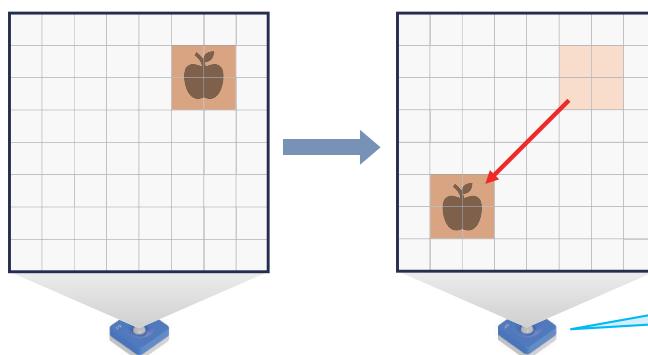


人感センサーの仕組み

人やものはその温度（体温）に応じて、赤外線を放っています。温度を持った人やものが動くと、その場の赤外線が変化します。

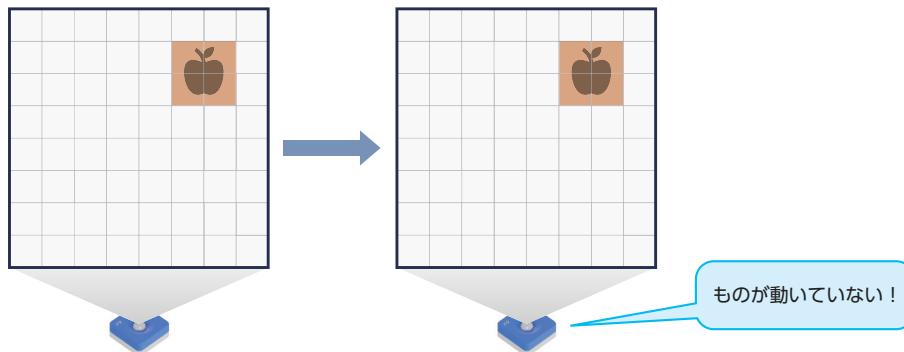
人感センサーは、赤外線の数値の変化を読み取って、人やものの動きを検知しています。

例 りんごが右上から左下に動いた場合



人感センサーの上に手をかざして、ユニットボードの表示がどのように変化するか確認してください。

例 リンゴが動いていない場合



このように人感センサーは、人やものから放出される赤外線の変化をセンサーで検知して、人やものの動きを計測しています。

人感センサーの使い方

【人感センサー ブロックの表示方法】

(1) メインユニットと人感センサーをつなげ、メインユニットとパソコンを接続します。メインユニットと人感センサーをつなげる場所はどこでも構いません。



(2) 接続完了後、カテゴリーに以下のようなブロックが表示されていることを確認してください。



【人感センサーの使い方】

人感センサーが 感知している ▾

このブロックでは、人感センサーが感知しているかどうかを調べることができます。

このブロックの△をクリックすると、以下のようない表示になります。この表示を変更することで、人感センサーが感知しているかどうかの条件を設定することができます。



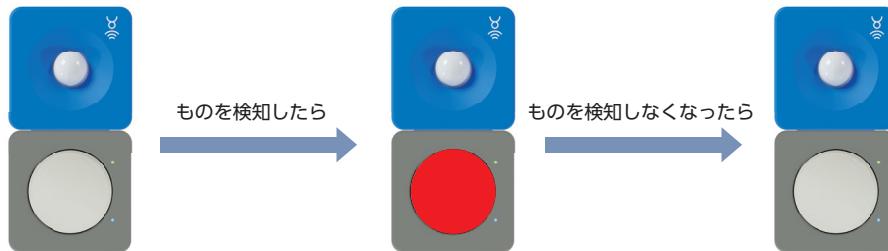
人感センサーは「感知している」→「感知していない」にセンサーの値が変化するまでに、数秒のタイムラグが発生します。



防犯ライトを作成しよう

【課題】

人感センサーが感知したときに、メインユニットのLEDが点灯するプログラムを作成しましょう。



① 「アイコンシート」を使用して、プログラムの流れを考えましょう。

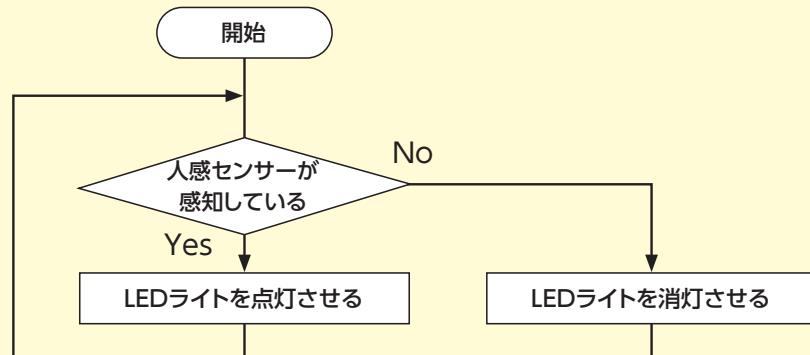
② ①で作成した「アイコンシート」を参考にして、フローチャートを作成しましょう。

グループ学習

「アイコンシート」を並べられる広い場所
で行ってください。

知識技能②

フローチャート例



(3) 「アイコンシート」と P18 で作成したフローチャートを参考にしてプログラムを作成しましょう。.....



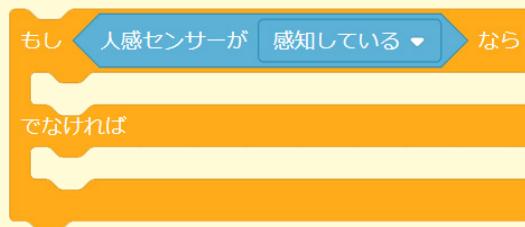
プログラム例



解説

人感センサーが感知しているかどうかを計測するために、条件分岐を行います。

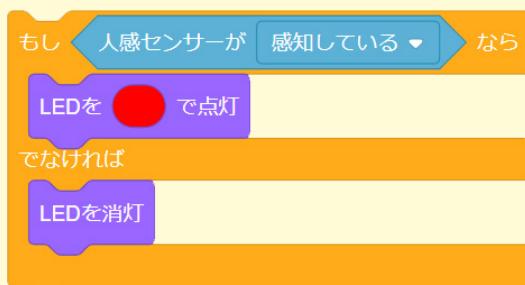
そのために以下のプログラムを作成します。



人感センサーが「感知しているかどうか」という条件で、条件分岐を行います。

もし「人感センサーが感知している」(= True) → LED を点灯

もし「人感センサーが感知していない」(= False) → LED を消灯



以上のプログラムだと一瞬でプログラムが終了してしまいます。そのため、「ずっと～する」のブロックを使用します。

こうすることで、常に人感センサーの条件分岐を行うことができます。

知識技能②

④プログラムが完成したら、プログラムを実行しましょう。

人感センサーに手をかざして、かざしている間のみ LED ライトが点灯することを確認してください。

人感センサーは「感知している」→「感知していない」にセンサーの値が変化するまでに、数秒のタイムラグが発生します。

学向力・人間性②

⑤達成できた項目にチェックを入れましょう。

- 人感センサーの仕組みについて理解できた。
- 防犯ライトを作成することができた。

学校の防犯システムを考えよう

【課題】

これまでに学習したユニットを使って、教室の防犯力を向上させるアイデアを考えましょう。教室のどこにアーテックリンクスをとりつけ、どのような動きをさせるか考えましょう。



①「アイコンシート」を使ってプログラムの流れを考えながらアイデアを考えましょう。またそのアイデアによって、どのような課題を解決することができるかを以下の表に書き入れましょう。

設置する場所	
プログラムの動き	<p>グループ学習 「アイコンシート」を並べられる広い場所 で行ってください。</p>
解決する課題	

設置する場所	
プログラムの動き	
解決する課題	

②「アイコンシート」を参考にして、プログラムを作成しましょう。

③ プログラムを実行してみましょう。

④ 企画書を作成しましょう。

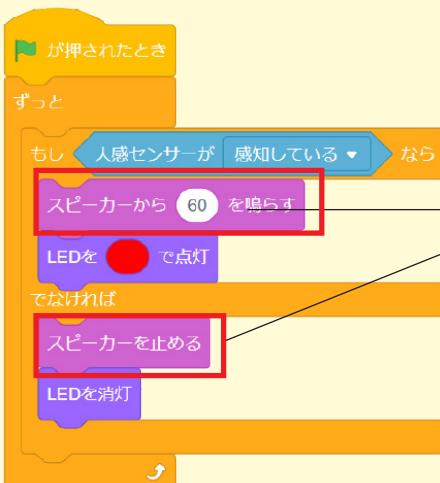
作成したプログラムの特徴と、工夫した点をまとめて製品企画書を作成します。

アイデア例とプログラミング例

[例 1] プログラムの動き：人感センサーが反応したときに、LED とブザー音でお知らせする。

設置する場所	教室のドア
プログラムの動き	人感センサーが反応したときに、LED とブザー音でお知らせする。
解決する課題	LEDが点灯しただけでは気づかないが、ブザー音も合わせることで気づかせることができる。

プログラム例



The Scratch script consists of the following blocks:

- When green flag clicked
- Forever loop:
 - If [人感センサーが感知している] then [60] do [スピーカーから 60 を鳴らす]
 - Else [LEDを [red] で点灯]
 - [でなければ]
 - [スピーカーを止める]
 - [LEDを消灯]

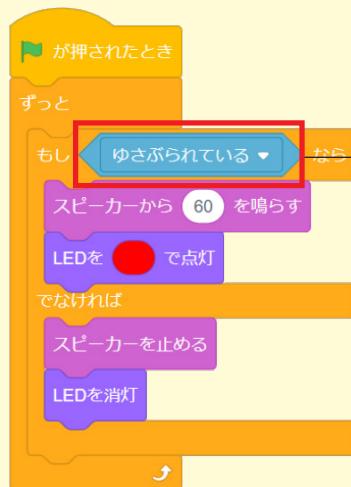
実習

人感センサーが検知しているときのみ、ブザーを鳴らします。

[例 2] プログラムの動き：加速度センサーを窓に取り付ける。窓が動いたら LED とブザーでお知らせする。

設置する場所	教室の窓
プログラムの動き	加速度センサーを窓に取り付ける。窓が動いたら LED とブザーでお知らせする。
解決する課題	教室の窓からの侵入を防ぐことができる。

プログラム例



加速度センサーが動きを感知しているときに、ブザーと LED ライトを点灯します。

[例 3] プログラムの動き：加速度センサーと人感センサーをものに取り付ける。加速度センサーと人感センサーが両方ともに検知しているときに LED とブザーを鳴らす。

設置する場所	持ち出し禁止のもの
プログラムの動き	加速度センサーと人感センサーをものに取り付ける。加速度センサーと人感センサーが両方ともに検知しているときに LED とブザーを鳴らす。
解決する課題	人感センサーと加速度センサーを組み合わせることで、防犯力を強化する。

プログラム例



ものが動かされている、かつ人感センサーが反応しているときに、このセンサーが取り付けられたものが移動されていることを認識します。

これ以降のページは、応用セットのセンサーでつくれる例の紹介です。生徒の興味・関心に応じて参考にしてください。
ユニットを組み合わせて、以下のデジタル楽器を作成していきましょう。

【目次】

- 電子オルゴールを作成しよう P. 25

メインユニットのボタンを押したときに、「きらきら星」を演奏するプログラムを作成します。

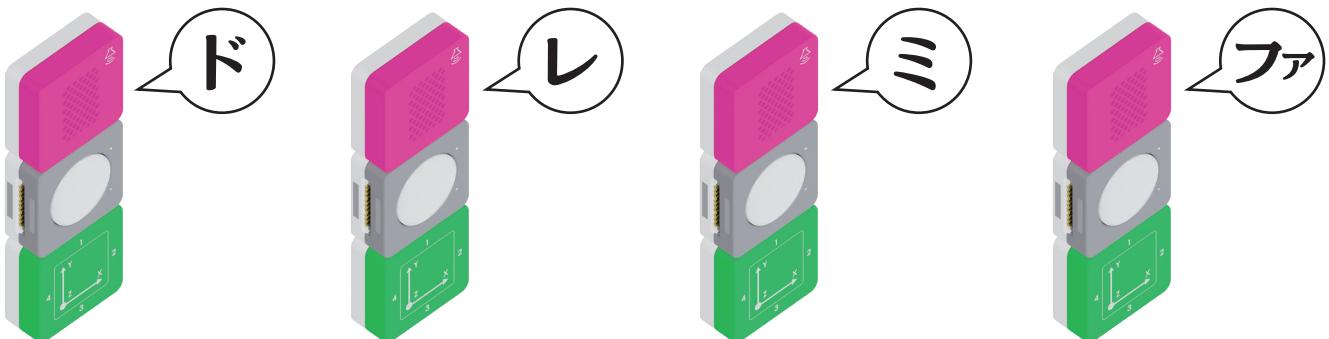
(使用ユニット：メインユニット、スピーカー)



- デジタルハンドベルを作成しよう P.29

加速度センサーによって、揺さぶられたことを検知し、スピーカーから音を鳴らします。

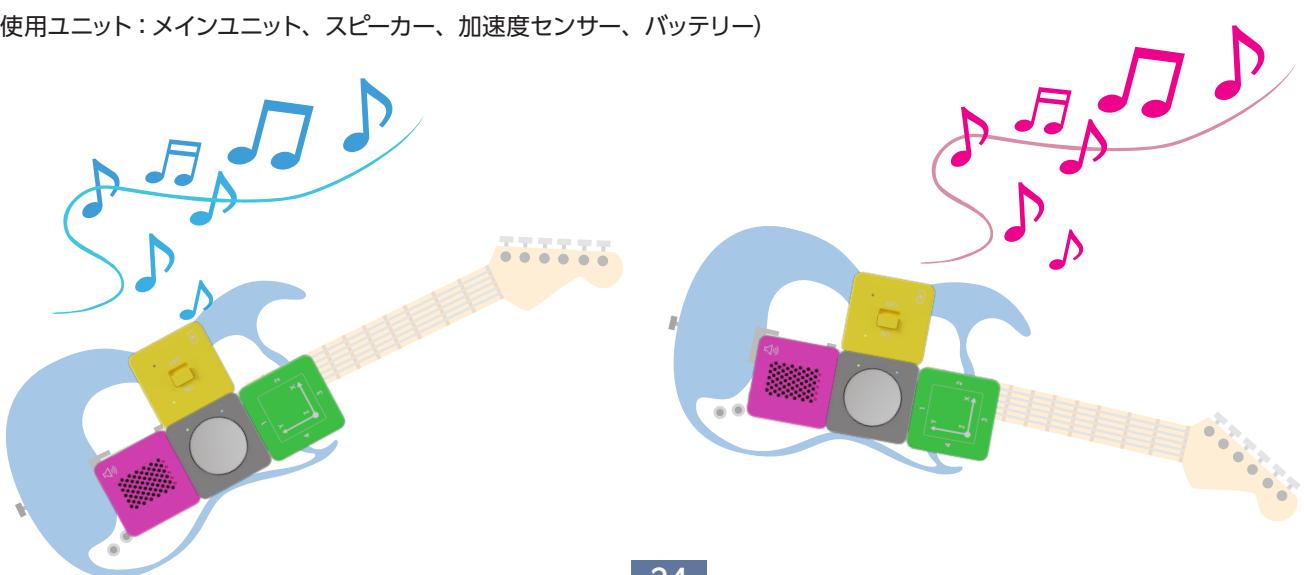
(使用ユニット：メインユニット、スピーカー、加速度センサー)



- デジタルエアギターを作成しよう P.32

加速度センサーによって傾きを調べ、傾きによって異なる音を鳴らします。

(使用ユニット：メインユニット、スピーカー、加速度センサー、バッテリー)



電子オルゴールを作成しよう

【課題】

ボタンを押したときに、「きらきら星」を演奏する、電子オルゴールのプログラムを作成しましょう。



① きらきら星は3種類のメロディーでつくられています。

青枠の「ド・ド・ソ・ソ・ラ・ラ・ソ」のメロディーを作成しましょう。

Three staves of musical notation on a staff. The top staff is blue and labeled 'ド ド ソ ソ ラ ラ ソ'. The middle staff is green and labeled 'ソ ソ フア フア ミ ミ レ'. The bottom staff is blue and labeled 'ド ド ソ ソ ラ ラ ソ'. Below each staff are the corresponding note names in Japanese: 'ド', 'ド', 'ソ', 'ソ', 'ラ', 'ラ', 'ソ' for the first staff; 'ソ', 'ソ', 'フア', 'フア', 'ミ', 'ミ', 'レ' for the second; and 'ド', 'ド', 'ソ', 'ソ', 'ラ', 'ラ', 'ソ' for the third.

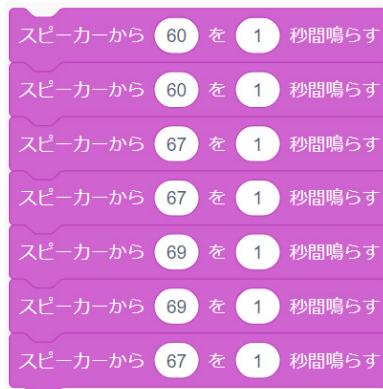
(1) カテゴリー 音 から スピーカーから 60 を 1 秒間鳴らす をスクリプトエリアにドラッグします。「60」の部分をクリックすると、鍵盤が表示されます。

A screenshot of the Scratch script editor. On the left is the script palette with categories like '動き', '見た目', '音', etc. A red arrow points from the '音' category to a 'Speaker' sound block in the script area. This block has parameters '60' and '1'. A blue arrow points down to a piano keyboard block below. The piano keys are labeled with Japanese note names: 'ド(60)', 'レ', 'ミ', 'フア', 'ソ', 'ラ', 'シ', and 'C(72)'. The '60' in the piano block corresponds to the '60' in the speaker block.

この鍵盤は音階通りに配置されていて、鍵盤をクリックするとブロックの数字が変更されます。プログラム上では、この数字によって音の違いを判定しています。

(2) メロディーが鳴るようにプログラムを作成していきます。楽譜に合わせてブロックを並べていきます。

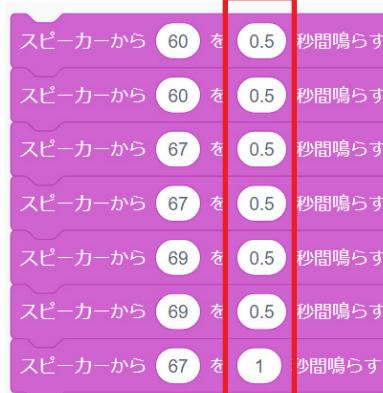
【プログラム例】



(3) 音の鳴る長さを決定します。

(2) で作成したプログラムをクリックして、実際に音を再生してみましょう。テンポがおかしいと感じた場合は、音を鳴らす秒数を変更してみましょう。

【プログラム例】



(4) 作成したプログラムを関数でまとめます。

「きらきら星」は6小節の曲ですが、3種類のメロディーで構成されています。6小節分のプログラムを作成するとプログラムが長くなってしまうため、同じメロディーは関数を使ってまとめます。

①カテゴリー から「ブロックを作る」をクリックします。

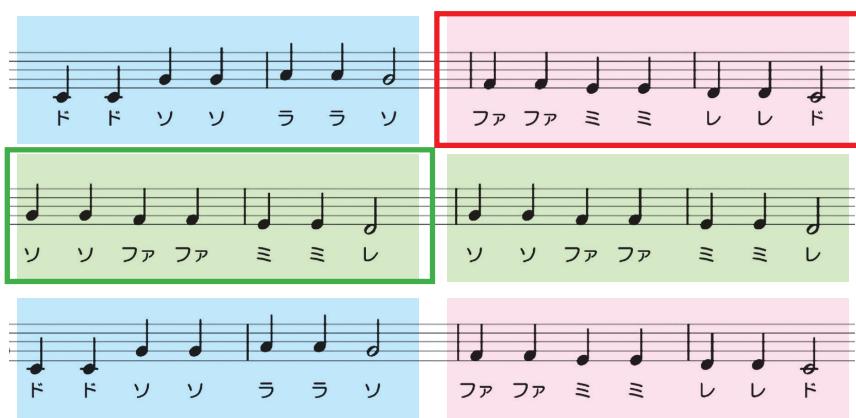
関数

②関数の名前を設定します。今回の例では、「ドドソソララソ」としています。この関数にまとめるプログラムに関連した名前だと、整理しやすくなります。

③「OK」をクリックすると関数のブロックがパレット上に表示されます。



(5) 関数を作成すると、スクリプトエリアに  が作成されます。このブロックの下にまとめたいプログラムをつなげます。 を実行することで、関数でまとめたプログラムを実行することができます。



②赤枠と緑枠のメロディーを作成します。

①の(1)～(5)を参考にして、以下のプログラムを作成しましょう。

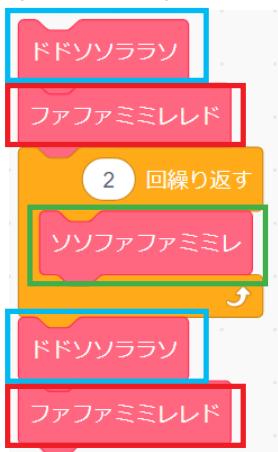
<p>(プログラム例・赤枠)</p>  <pre> function [ソソファファミミレ] speaker say [スピーカーから (67) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (67) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (65) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (65) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (64) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (64) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (62) を (1) 秒間鳴らす] until [] end </pre>	<p>(プログラム例・緑枠)</p>  <pre> function [ファファミミレレド] speaker say [スピーカーから (65) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (65) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (64) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (64) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (62) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (62) を (0.5) 秒間鳴らす] until [] speaker say [スピーカーから (60) を (1) 秒間鳴らす] until [] end </pre>
--	---

③「きらきら星」のメロディーを作成します。



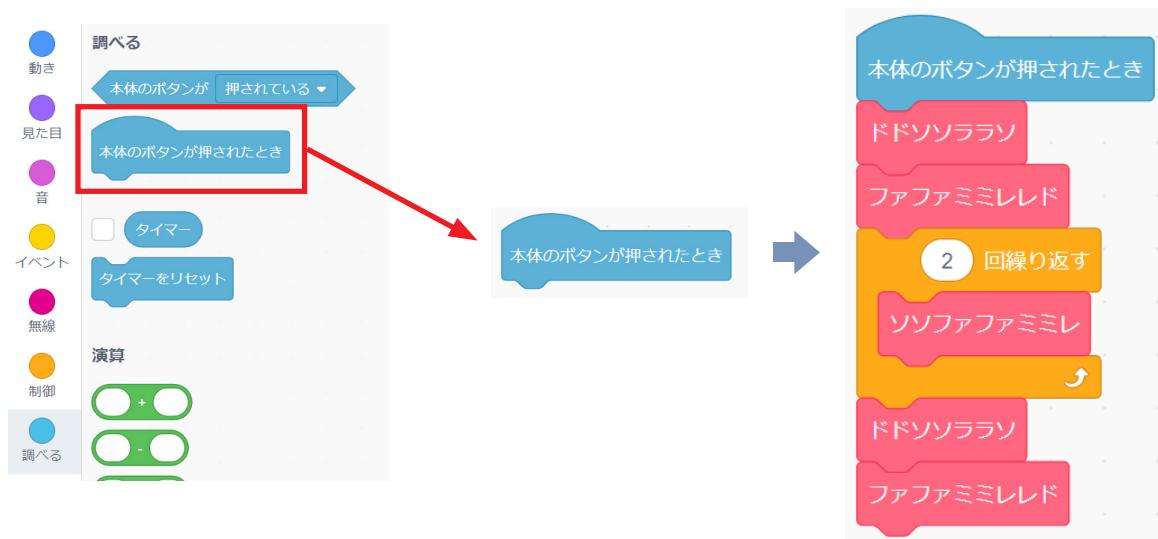
メロディーが鳴るように、作成した関数のブロックをつなげていきます。

(プログラム例)



④ボタンを押したときにメロディーが再生されるようにします。

カテゴリー から をスクリプトエリアにドラッグし、③で作成したプログラムの上につなげます。



⑤メインユニットのボタンを押して、実際に「きらきら星」が演奏されるか試してみましょう。

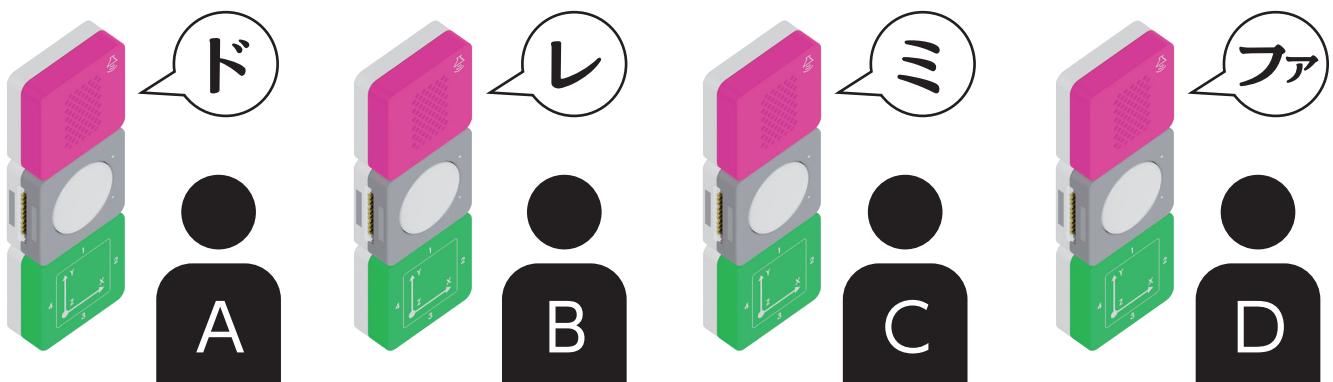
デジタルハンドベルを作成しよう

【課題】

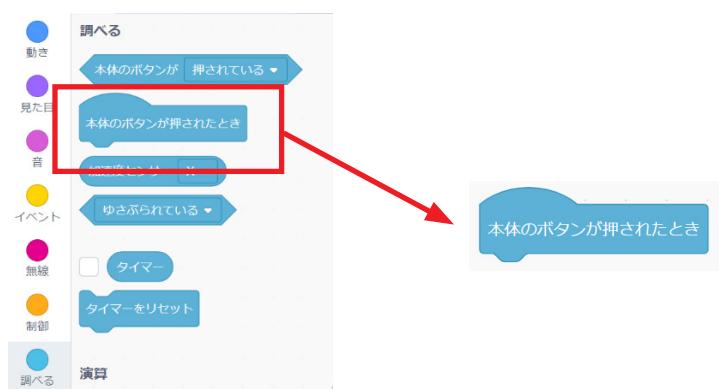
アーテックリンクスをゆさぶったときに、設定した音を鳴らすプログラムを作成しよう。



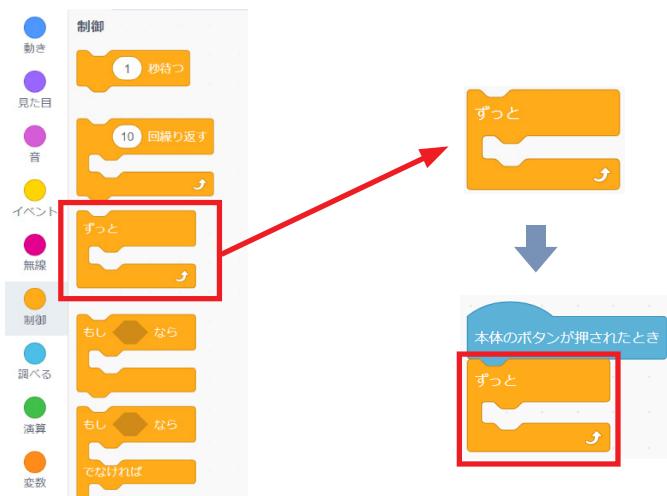
今回のプログラムは1つのメインユニットに1つの音を保存します。実際のハンドベルのように演奏する場合は、演奏する人数でグループをつくってプログラムを作成してください。



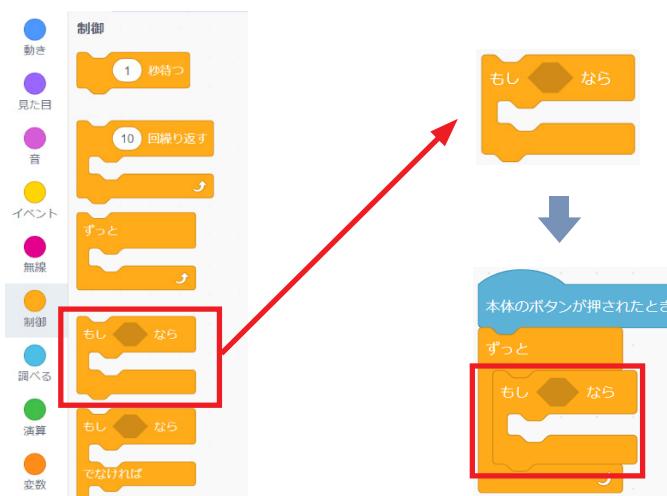
①カテゴリー 調べる から 本体のボタンが押されたとき をスクリプトエリアにドラッグします。



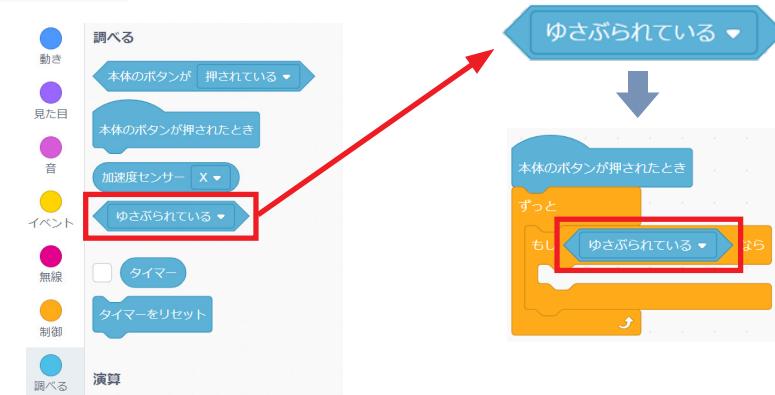
②カテゴリー  から 制御 をスクリプトエリアにドラッグします。①で作成したプログラムの下につなげます。



③カテゴリー  から 制御 をスクリプトエリアにドラッグします。②で作成したプログラムの間に挿入します。

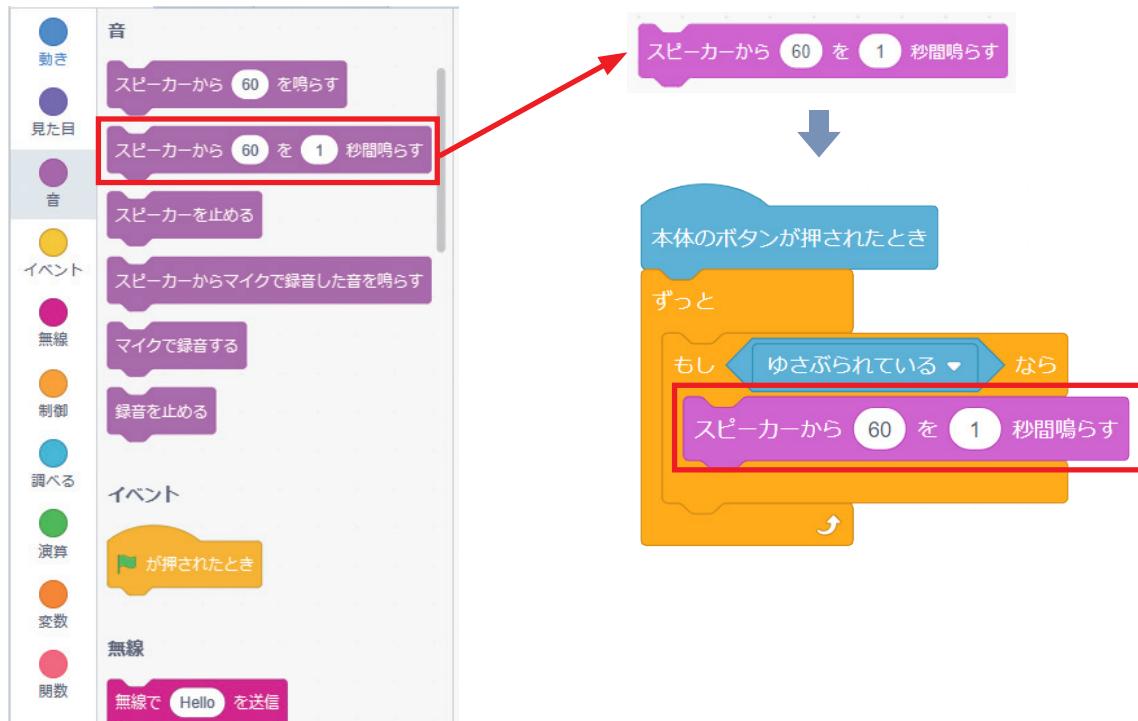


④カテゴリー  から 調べる をスクリプトエリアにドラッグします。③で作成したプログラムの
「もし...なら」の  の部分に挿入します。



⑤カテゴリー 音 から 

もし なら のブロックの間に挿入します。

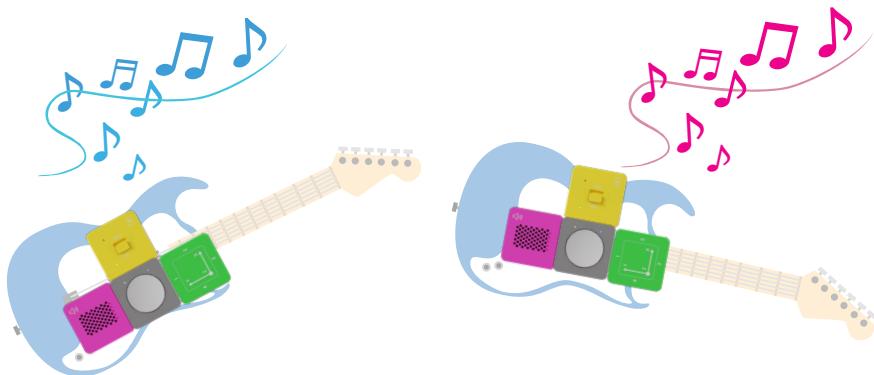


以上でプログラムは完成です。作例では、ゆさぶったときに「ド」の音が流れるようにしました。ほかの音も流れるようにプログラムを作成してみましょう。

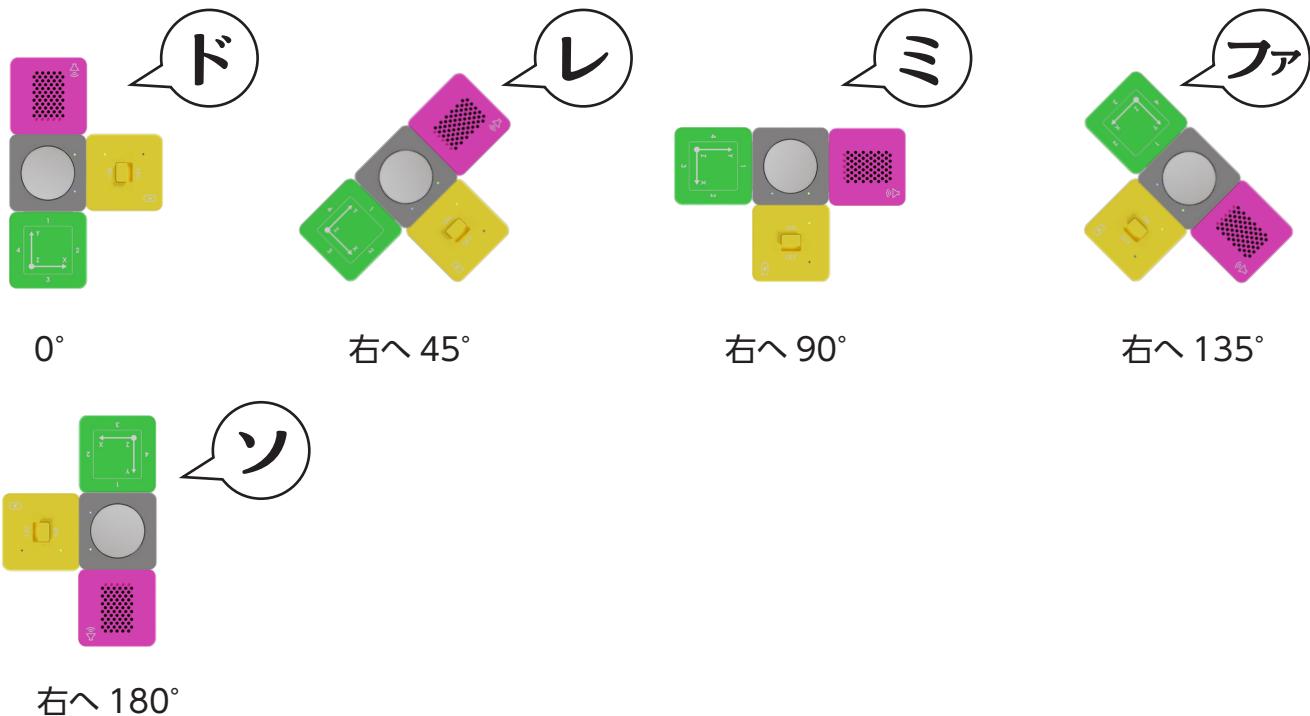
デジタルエアギターを作成しよう

【課題】

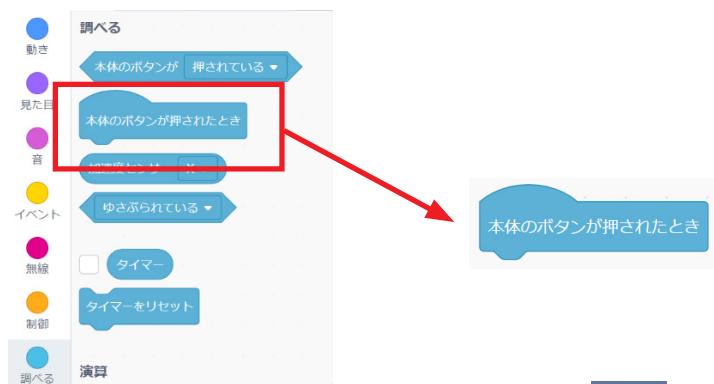
アーテックリンクスの傾きによって、音が変わるプログラムを作成しましょう。



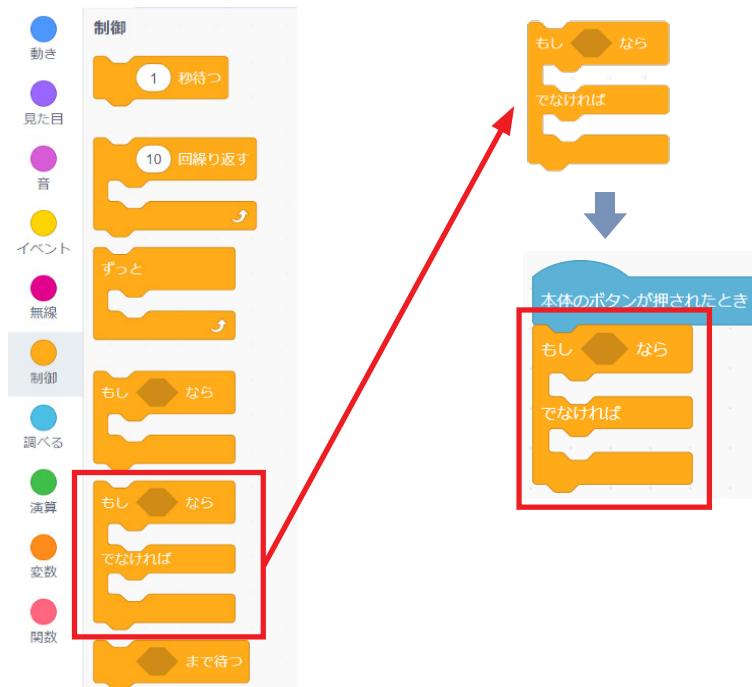
アーテックリンクスが以下のように傾いているときに、設定した音をスピーカーから流します。



①カテゴリー 調べる から 本体のボタンが押されたとき をスクリプトエリアにドラッグします。



②カテゴリー  から 制御 をスクリプトエリアにドラッグします。①で作成したプログラムの下につなげます。

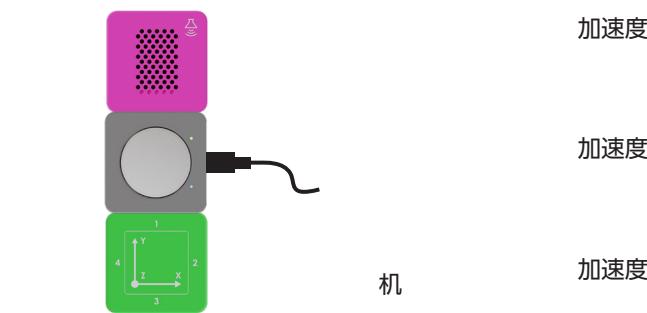


③加速度センサーの傾きを調べるために、「センサーボード」から加速度センサーの値を確認します。

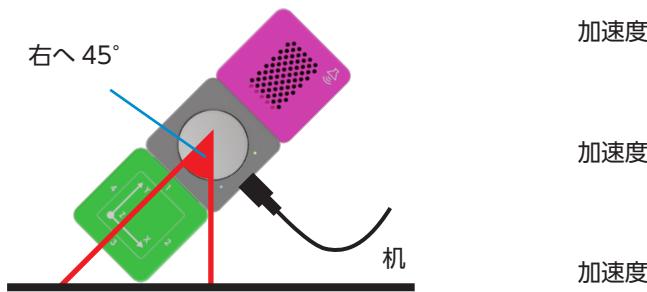
以下の図のようにアーテックリンクスを置いたときの、加速度センサー X, Y, Z の値をそれぞれ確認しましょう。

加速度センサーは図の通りに置いてください。図と異なる場合、例と値が変わることがあります。

(1)



(2)



(3)		加速度センサー X	0.99
		加速度センサー Y	-0.02
		加速度センサー Z	-0.05
(4)		加速度センサー X	0.83
		加速度センサー Y	0.51
		加速度センサー Z	-0.01
(5)		加速度センサー X	-0.05
		加速度センサー Y	0.98
		加速度センサー Z	0.02

(1) と (5) を比較すると加速度センサー Y の値が最も大きく変化していました。そのため、加速度センサー Y の値を鳴らす音を変えるための条件とします。

	(1)	(2)	(3)	(4)	(5)
加速度センサー X	0.03	0.86	0.99	0.83	-0.05
加速度センサー Y	-1.01	-0.51	-0.02	0.51	0.98
加速度センサー Z	0.05	-0.13	-0.05	-0.01	0.02

④しきい値を設定します。

加速度センサーの傾きを区別する場合に、2つの状態の境目となる値を決めて判断します。このような境目となる値のことを「しきい値」と言います。③で確認した加速度センサー Y の値を元にしきい値を決定して、プログラムを作成していきます。

【しきい値の決定】

しきい値は2つの値の中央となる値にします。

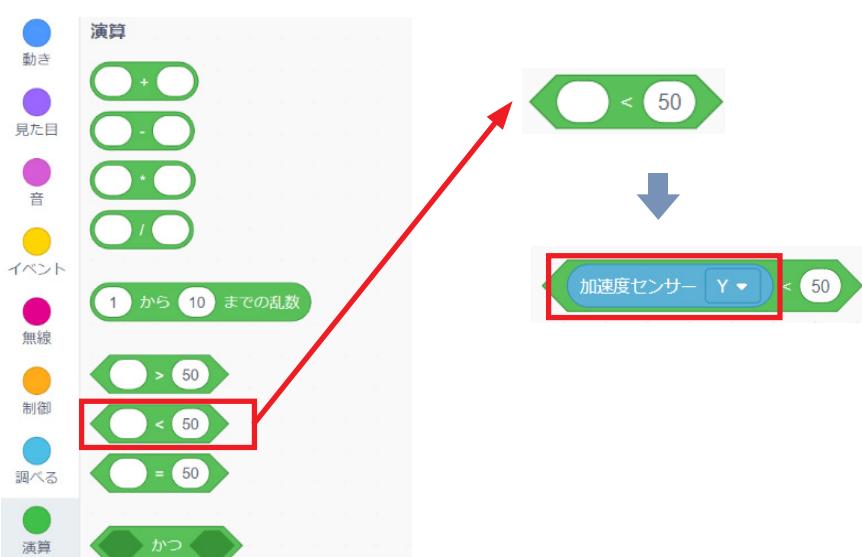
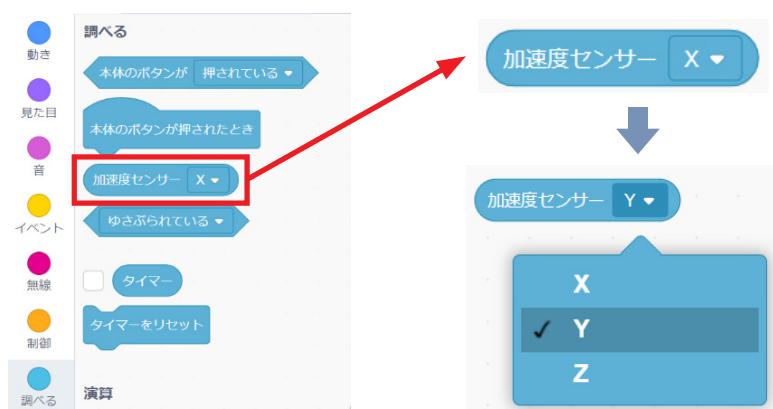
(③の(1)と(2)の状態を区別するためのしきい値)



⑤カテゴリー 演算 から をスクリプトエリアにドラッグします。また、カテゴリー 調べる から

加速度センサー X ▾ をスクリプトエリアにドラッグし、「X」の部分を「Y」に変更します。

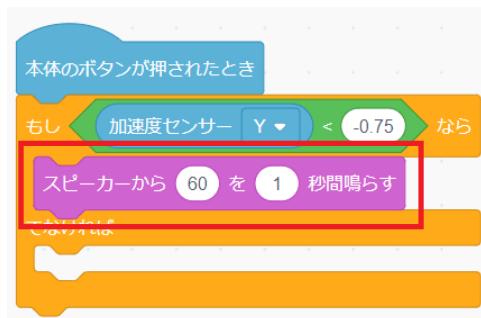
このブロックに を以下のように挿入します。



⑥ ⑤で作成したプログラムにしきい値を入力していきます。③の(1)と(2)のしきい値を入力します。ここで作成したプログラムを②で作成したプログラムに挿入します。



⑦ カテゴリー から をスクリプトエリアにドラッグします。⑥で作成したプログラムに挿入します。



⑧ ②～⑦を参考にして以下のプログラムを作成し、⑦で作成したプログラムの「～でなければ」の下に挿入します。

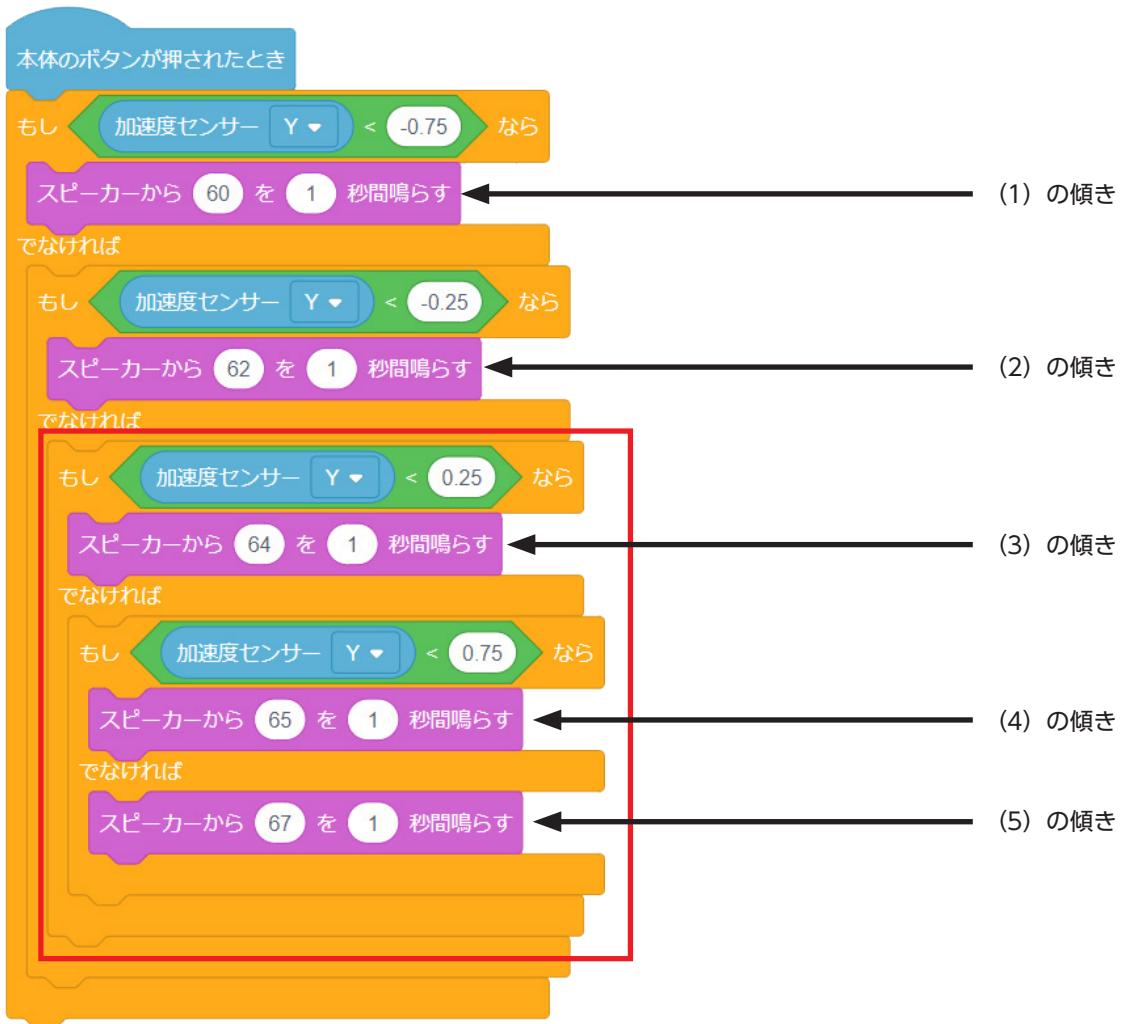


⑨ ⑧を参考にして、③で確認した値ごとにプログラムを作成します。

③の(5)のしきい値のプログラムでは、加速度センサーYの値に上限を定める必要がないため、を取り除いてください。



のブロック



⑩以上でプログラムは完成です。加速度センサーの傾きを変えて、メインユニットのボタンを押してみましょう。

傾きを変えるごとにスピーカーから流れる音が変わっていくことを確認しましょう。

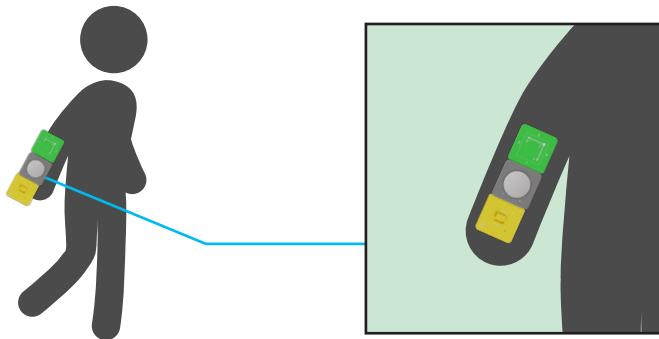
ユニットを組み合わせて、以下のトレーニング補助装置を作成していきましょう。

【目次】

○ウェアラブル端末を作成しよう① P. 39

加速度センサーを使って、歩数を計測するプログラムを作成します。

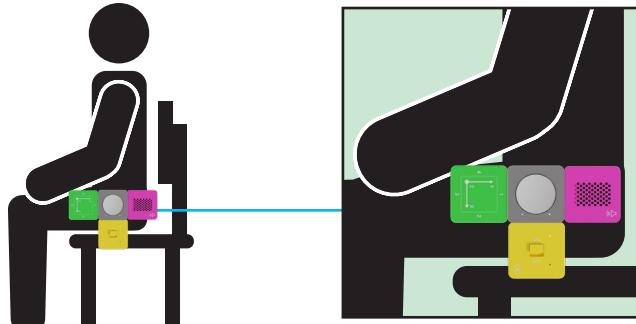
(使用ユニット：メインユニット、加速度センサー、バッテリー)



○ウェアラブル端末を作成しよう② P.45

加速度センサーとスピーカーを使って、座りすぎなどの運動不足のときに警告するプログラムを作成します。

(使用ユニット：メインユニット、スピーカー、加速度センサー、バッテリー)



○ダンベル挙げカウント計測装置を作成しよう P.49

加速度センサーとスピーカーを使って、ダンベルを持ち上げた回数を数えることができるプログラムを作成します。

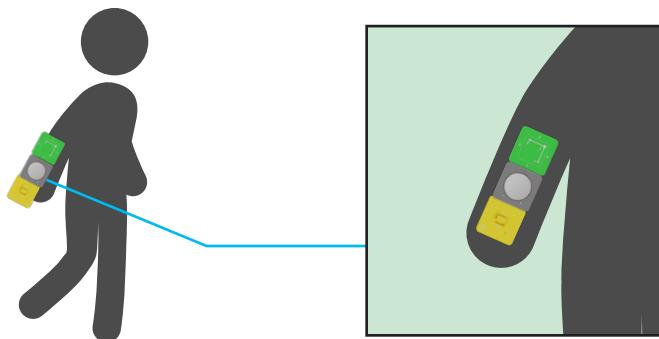
(使用ユニット：メインユニット、スピーカー、加速度センサー、バッテリー)



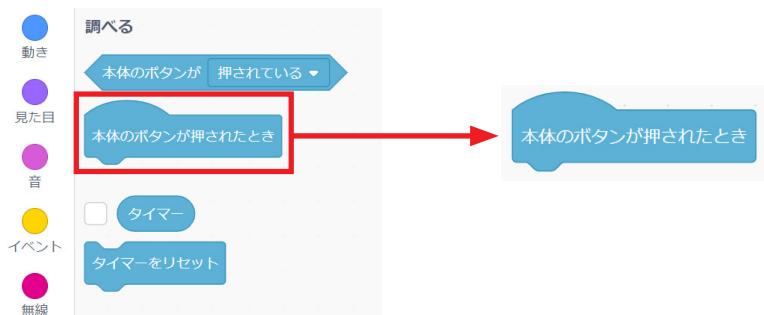
ウェアラブル端末を作成しよう①

【課題】

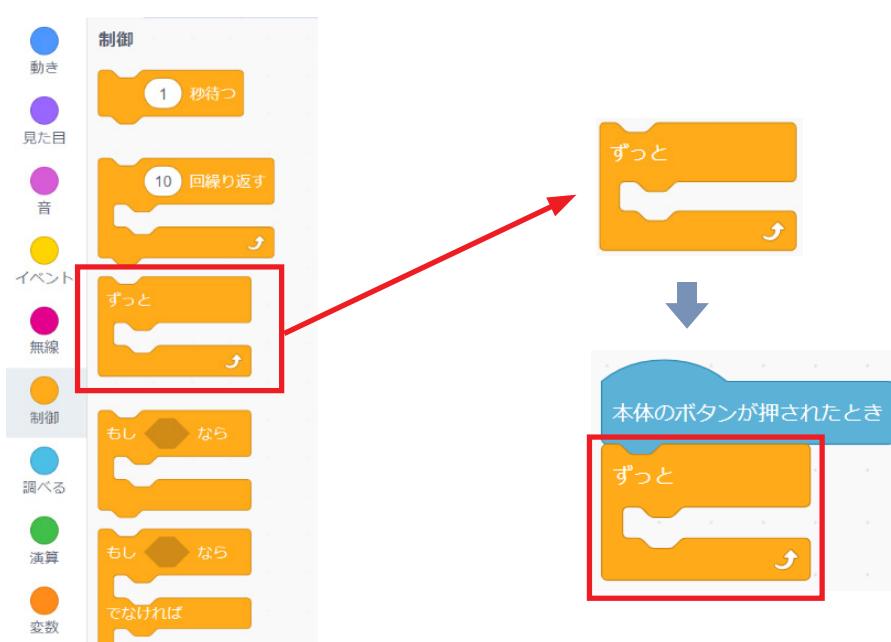
加速度センサーを使って、歩数を計測するプログラムを作成します。



① カテゴリー 調べる から 本体のボタンが押されたとき をスクリプトエリアにドラッグします。



② カテゴリー 制御 から ずっと をスクリプトエリアにドラッグします。①で作成したプログラムの下につなげます。



③しきい値を確認します。

今回は腕に取り付けるため、腕を振ったときに、加速度センサーの値がどのように変化するのかを確認します。以下の3つの場合を考えます。



腕が体の前にあるとき



腕が体の横にあるとき



腕が体の後ろにあるとき

(1) 腕が体の前にあるとき



加速度センサー X

-0.49

加速度センサー Y

-0.85

加速度センサー Z

-0.21

(2) 腕が体の横にあるとき



加速度センサー X

0.06

加速度センサー Y

-1.05

加速度センサー Z

-0.13

(3) 腕が体の後ろにあるとき



加速度センサー X

0.53

加速度センサー Y

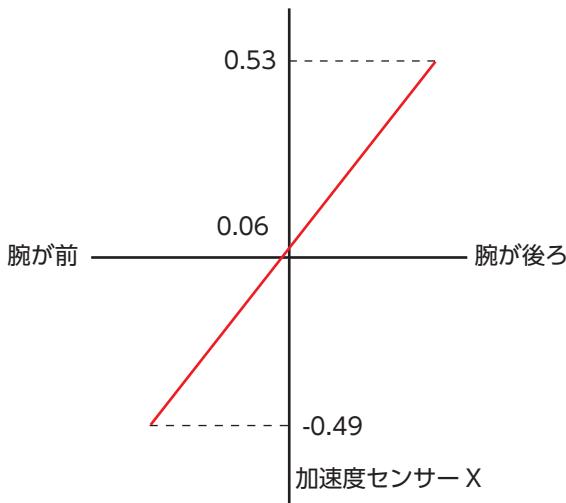
-0.84

加速度センサー Z

-0.19

今回のプログラムでは、しきい値として、確認した値のなかで最も変化が大きかった「加速度センサー X」を使います。

今回の例では、「加速度センサー X」の値は以下のように変化しています。



歩数を計測するプログラムを作成するためには、腕が前と後ろに振られたときのみに回数を計測する必要があります。そこで今回確認した値の場合では、腕が前か後ろに振られているかどうかを判定するためのしきい値を「0.06」とします。

④カテゴリー **制御** から をスクリプトエリアにドラッグし、②で作成したプログラムの「ずっと～する」の間に挿入します。

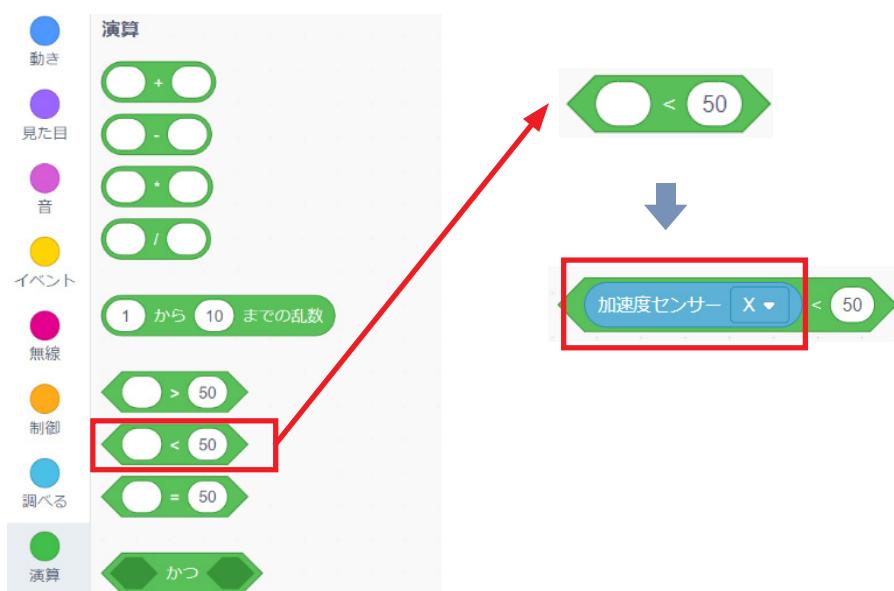


⑤歩数を計測するための変数を作成します。

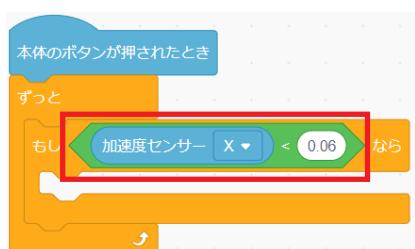
カテゴリー **変数** から「変数を作る」を選択し、変数名を設定します。今回の例では、新しい変数名を「歩数」としています。



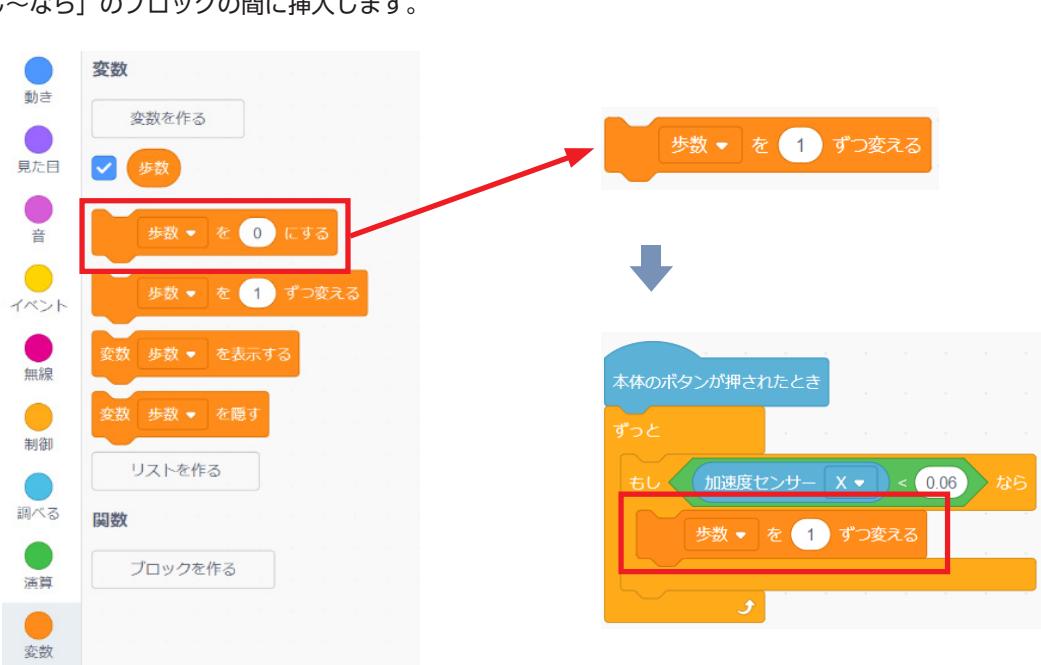
⑥ カテゴリー  から  をスクリプトエリアにドラッグします。また、カテゴリー  から  をスクリプトエリアにドラッグします。このブロックに  を以下のように挿入します。



⑦ ③で確認したしきい値を入力し、④で作成したプログラムに挿入します。

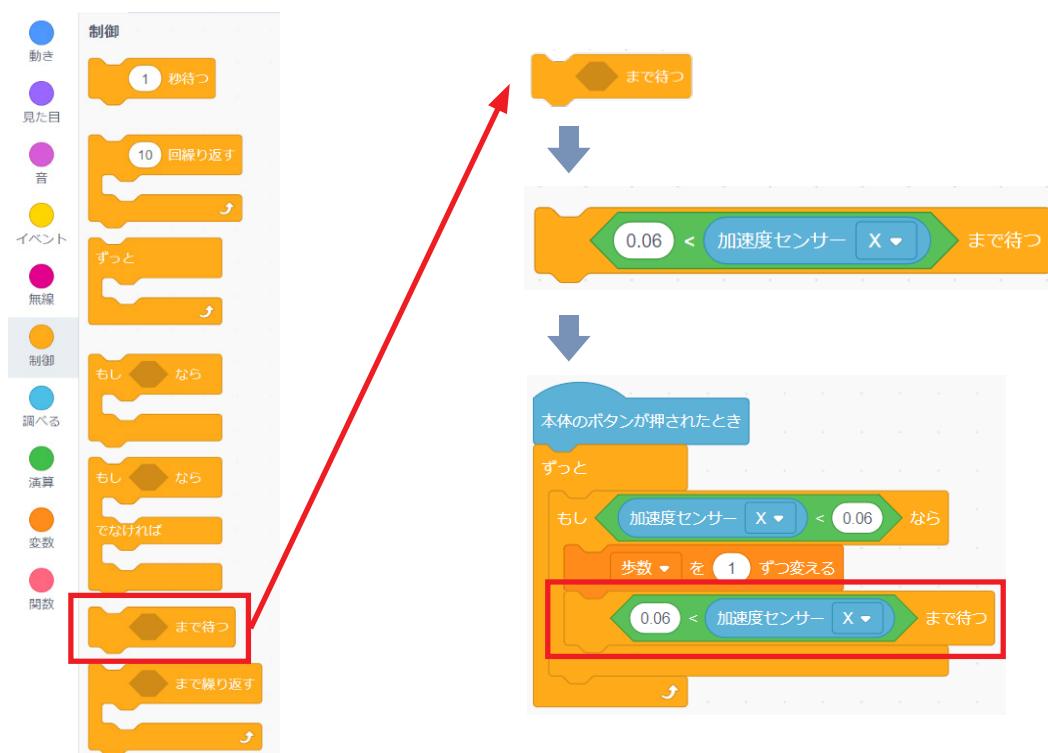


⑧カテゴリー  から  をスクリプトエリアにドラッグし、⑦で作成したプログラムの「もし～なら」のブロックの間に挿入します。



⑨カテゴリー  から  をスクリプトエリアにドラッグします。⑥の「腕が体の横にあるとき」を参考にして

以下のブロックを作成し、⑧で作成したプログラムに挿入します。



⑩以上でプログラムは完成です。実際にアーテックリンクスを取り付け、歩数を計測してみましょう。

【～まで待つブロック】

「～まで待つブロック」はある条件を満たすまで次のプログラムの実行を止めるためのブロックです。

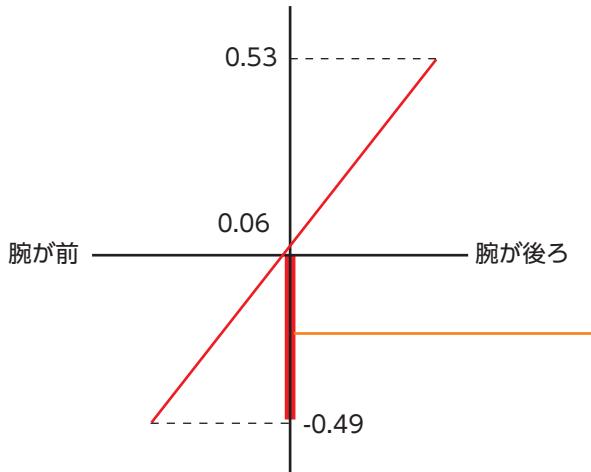
このブロックを挿入することによって、歩数を数えすぎることを防ぎます。

③でしきい値を設定しましたが、このままだと加速度センサーXの値が「0.06」より小さい場合に、繰り返し

歩数 ▾ を 1 ずつ変える

のプログラムを実行することになってしまいます。

(「～まで待つブロック」がない場合)



腕が前にあるとき ($-0.49 < \text{加速度センサーX} < 0.06$)

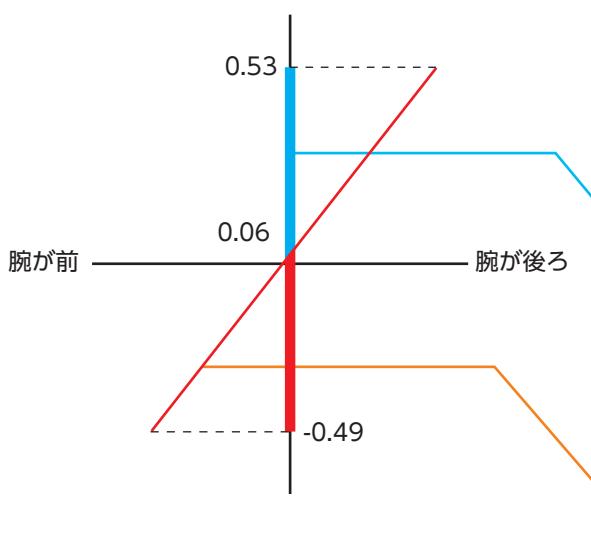
歩数 ▾ を 1 ずつ変える

を繰り替えし実行する。

歩数が
1, 2, 3, 4, 5, ...



(「～まで待つブロック」がある場合)



腕が後ろに来る ($0.06 < \text{加速度センサーX}$) まで歩数を数えない

歩数が 1

を実行する。



腕が前にあるとき ($-0.49 < \text{加速度センサーX} < 0.06$)

歩数 ▾ を 1 ずつ変える

を実行する。

歩数が 2



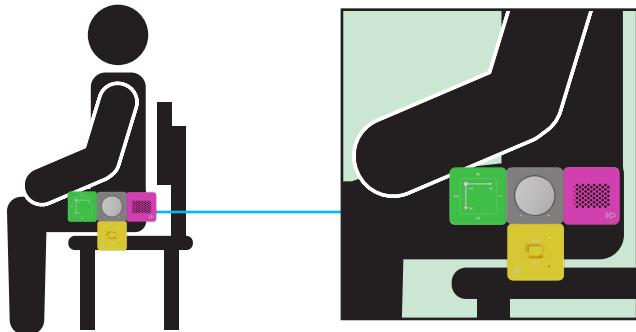
このように加速度センサーXの値が「0.06」より大きくなるまで、プログラムの実行を停止することによって、腕を振ったタイミングのみ 歩数 ▾ を 1 ずつ変える のプログラムを実行させることができます。

ウェアラブル端末を作成しよう②

【課題】

加速度センサーとスピーカーを使って、座りすぎているときに警告するプログラムを作成します。

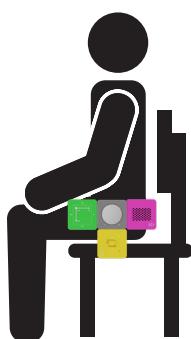
ポケットに入れて使用します。



①しきい値を設定します。

座っているときの加速度センサーの値と立っているときの加速度センサーの値を比較します。

(座っているとき)



加速度センサー X

0.99

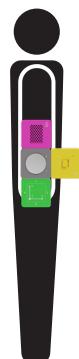
加速度センサー Y

0.02

加速度センサー Z

0.05

(立っているとき)



加速度センサー X

0.04

加速度センサー Y

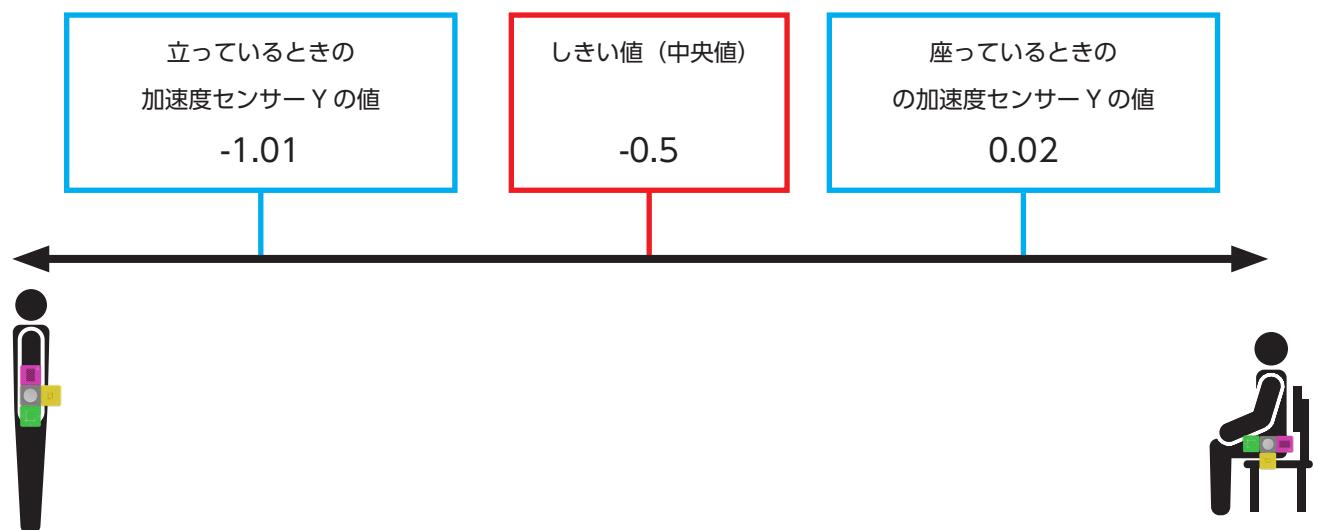
-1.01

加速度センサー Z

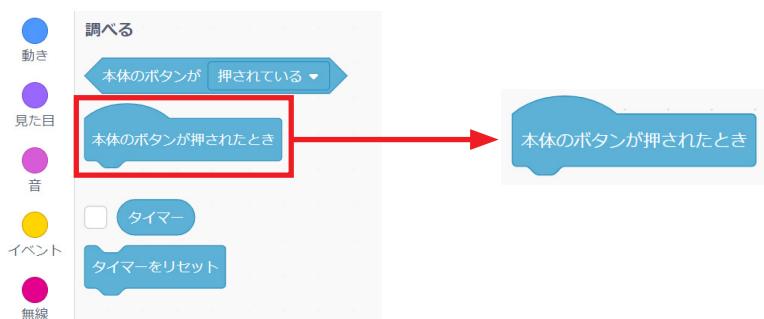
0.06

今回の例では、センサーの値の変化の大きかった加速度センサー Y の値からしきい値を設定します。

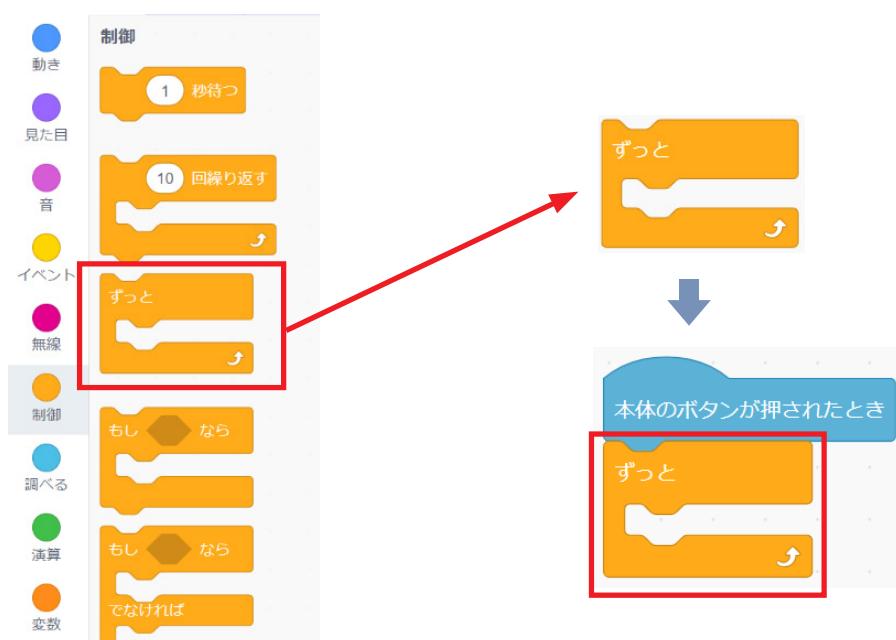
今回は加速度センサー Y 「-0.5」をしきい値として設定します。



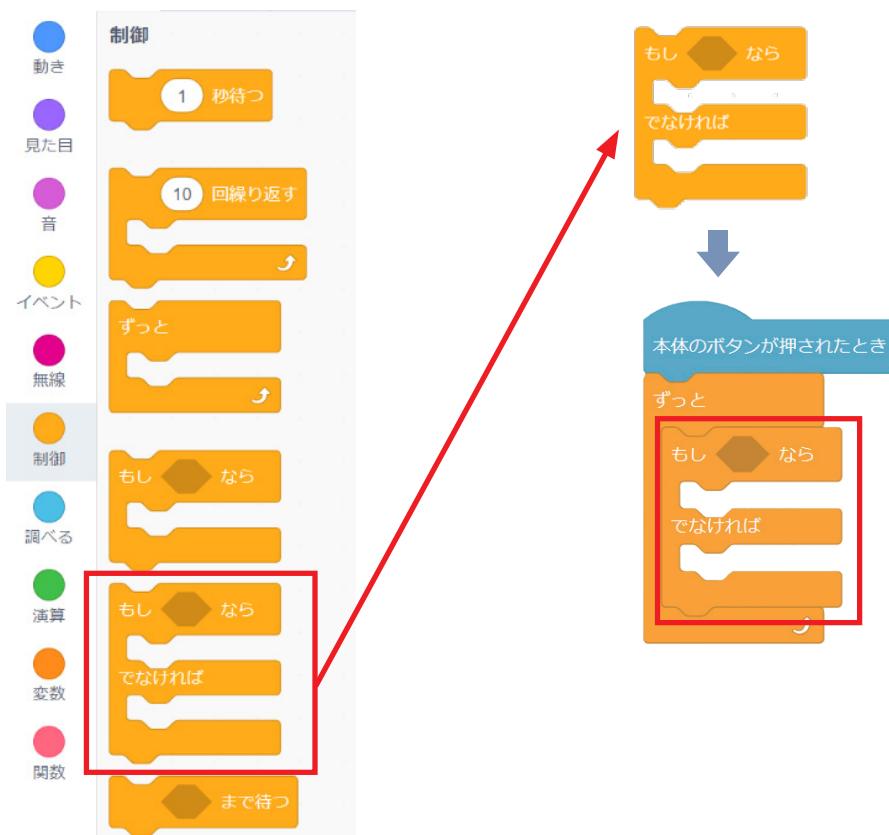
②カテゴリー 調べる から 本体のボタンが押されたとき をスクリプトエリアにドラッグします。



③カテゴリー 制御 から ずっと をスクリプトエリアにドラッグします。②で作成したプログラムの下につなげます。



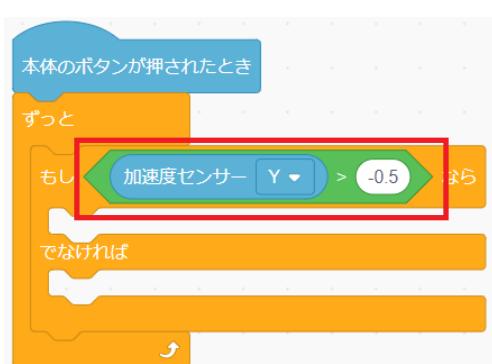
④ カテゴリー  から 制御 をスクリプトエリアにドラッグし、③で作成したプログラムの「ずっと～する」の間に挿入します。



⑤ ①で確認したしきい値を入力します。座っていることをセンサーの値から認識するために、以下のプログラムを作成します。



⑥ ⑤で作成したプログラムを④で作成したプログラムの「もし～なら、でなければ～」の  に挿入します。



⑦どれだけの時間座っているかを計測するプログラムを作成します。

カテゴリー  から  をスクリプトエリアにドラッグします。「タイマー」ブロックを使って、座っている時間を計測します。

もし 30分座り続けているときに警告する場合は、以下のプログラムを作成します。



タイマーの数値が 30 分 (= 1800 秒) を過ぎた場合に、LED を点灯し、スピーカーから音を流します。

⑧ ⑦で作成したプログラムを⑥で作成したプログラムの「もし～でなければ」の間に挿入します。



⑨立ち上がっている場合には、タイマーをリセットして警告しないようにプログラミングします。

カテゴリー  から  をスクリプトエリアにドラッグします。⑧で作成したプログラムの「～でなければ」のブロックの下に挿入します。



⑩以上でプログラムは完成です。実際にアーテックリンクスをポケットに入れてプログラムを実行してみましょう。

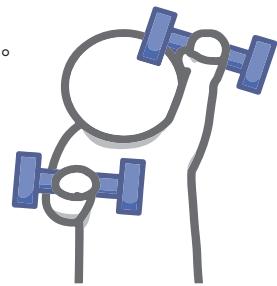
またテストとしてプログラムを実行する際は、タイマーの数値を 60 秒にするなどしてください。

ダンベル挙げカウント計測装置を作成しよう

【課題】

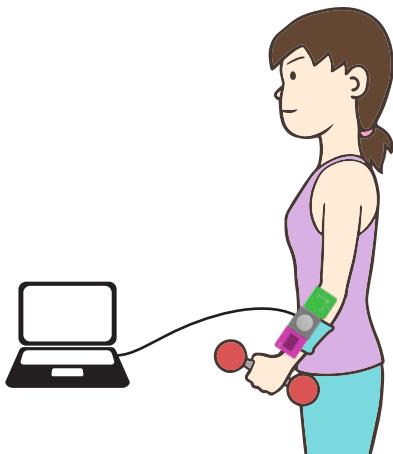
加速度センサーを使って、ダンベルを持ち上げたときに自動で回数を数えるプログラムを作成しよう。

アーテックリンクスを腕に取り付ける際は、マジックテープやバンドを使用してください。



①しきい値を設定します。

腕を下げているときと、腕を上げているときの加速度センサーの値を確認します。



加速度センサー X

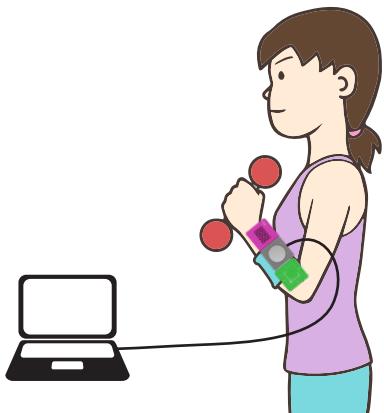
-0.27

加速度センサー Y

0.92

加速度センサー Z

-0.16



加速度センサー X

-0.14

加速度センサー Y

-1.00

加速度センサー Z

-0.15

今回の例では、加速度センサー Y 「-0.04」をしきい値として設定します。

挙げているときの
加速度センサー Y の値

-1.00

しきい値（中央値）

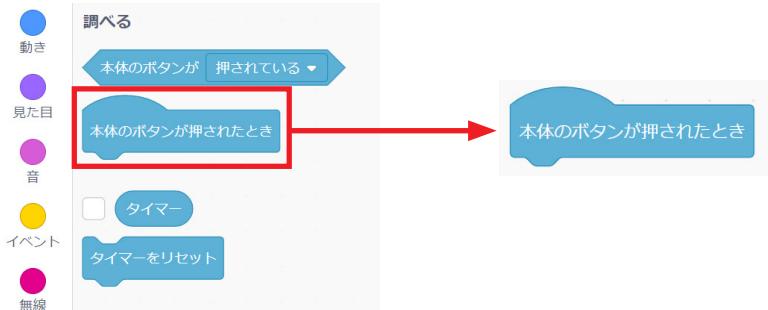
-0.04

下げているときの
加速度センサー Y の値

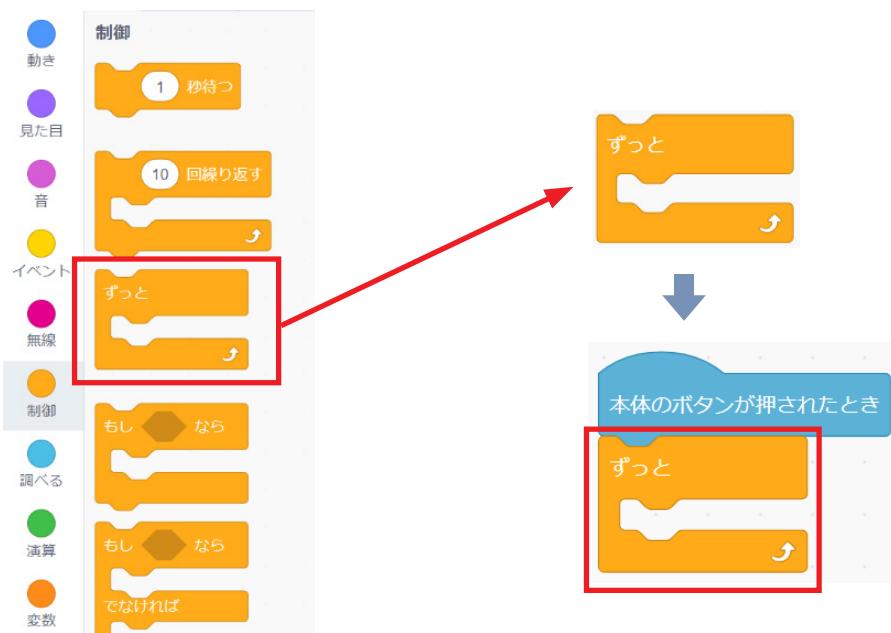
0.92

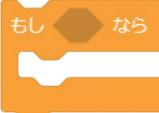


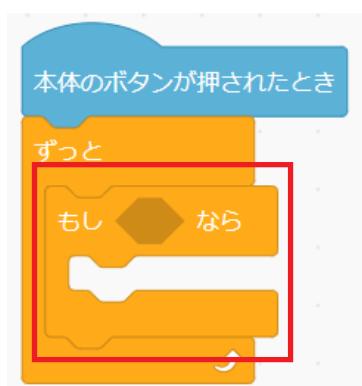
②カテゴリー 調べる から  本体のボタンが押されたとき をスクリプトエリアにドラッグします。



③カテゴリー 制御 から  ずっと をスクリプトエリアにドラッグします。②で作成したプログラムの下につなげます。



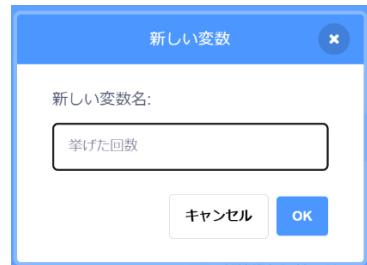
④カテゴリー 制御 から  もし なら をスクリプトエリアにドラッグし、③で作成したプログラムの「ずっと～する」の間に挿入します。



⑤歩数を計測するための変数を作成します。

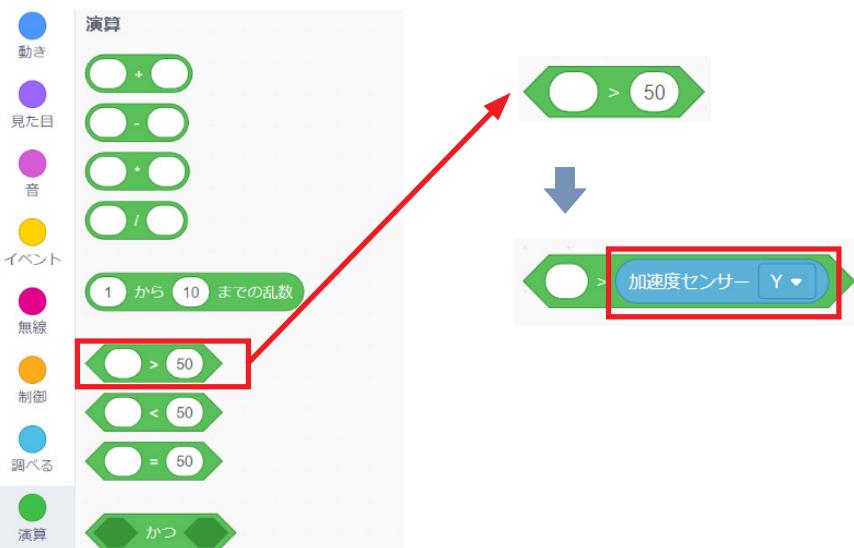
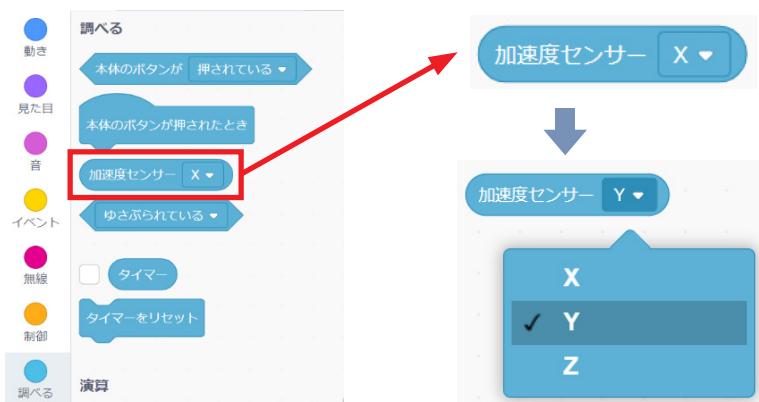
カテゴリー  から「変数を作る」を選択し、変数名を設定します。今回の例では、新しい変数名を「挙げた回数」としています。

変数



⑥カテゴリー  から  をスクリプトエリアにドラッグします。また、カテゴリー  から  をスクリプトエリアにドラッグし、「X」の部分を「Y」に変更します。

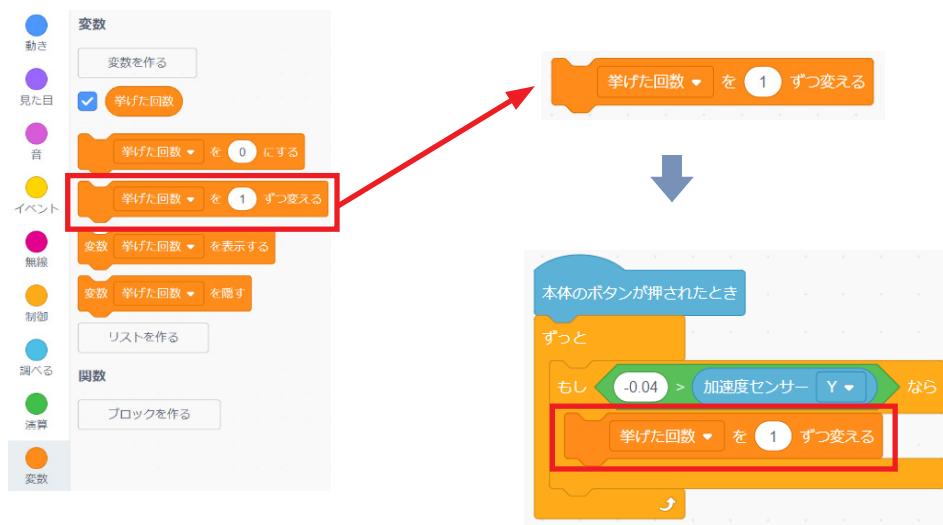
このブロックに  を以下のように挿入します。



⑦ ①で確認したしきい値を入力し、④で作成したプログラムに挿入します。



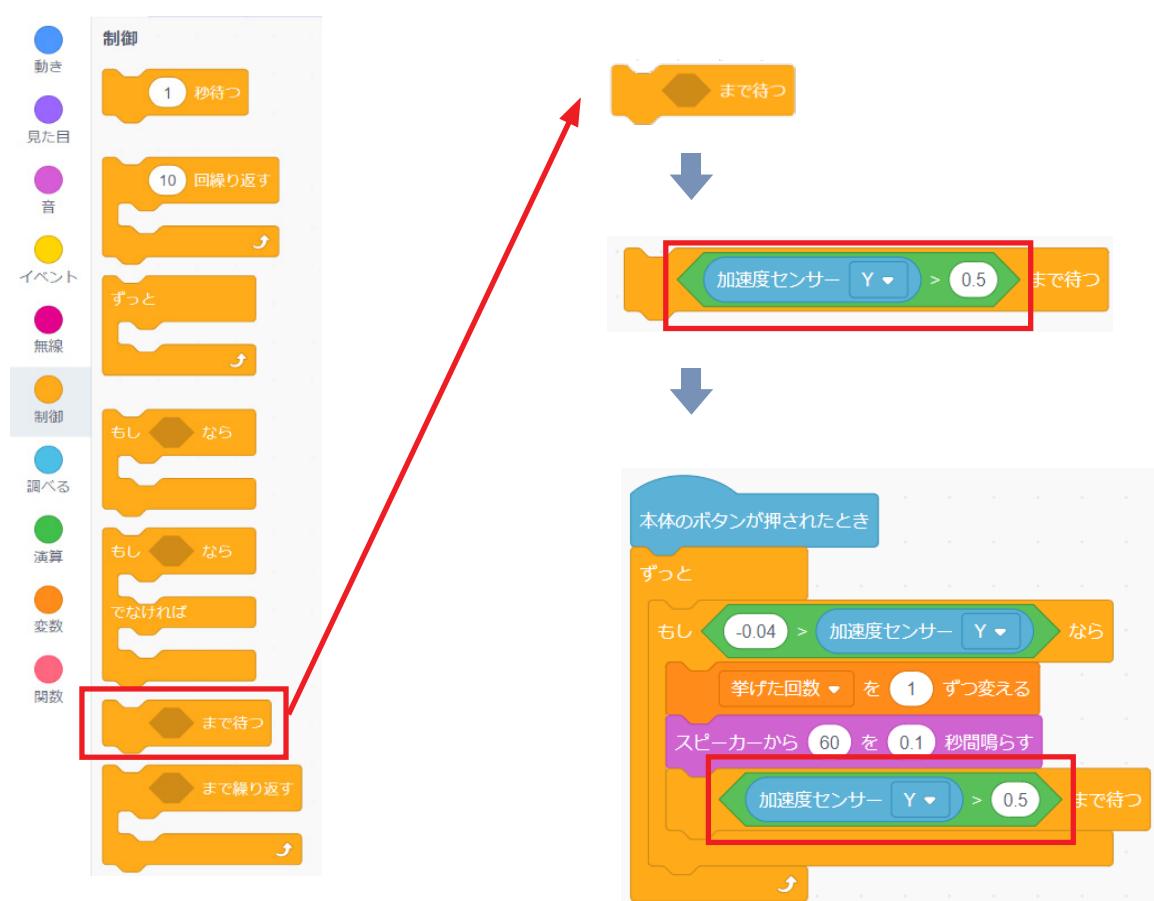
⑧ カテゴリー 变数 から 「挙げた回数 ▼ を 1 ずつ変える」 をスクリプトエリアにドラッグし、⑦で作成したプログラムの「もし～なら」のブロックの間に挿入します。



⑨ カテゴリー 音 から 「スピーカーから [60] を [1] 秒間鳴らす」 をスクリプトエリアにドラッグします。⑧で作成したプログラムに挿入します。



⑩カテゴリー  から  まで待つ をスクリプトエリアにドラッグします。⑦を参考にして以下のブロックを作成し、⑨で作成したプログラムに挿入します。



以上でプログラムは完成です。実際にアーテックリンクスを取り付けてプログラムを実行してみましょう。

ArTeC Links
アーテックリンクス

応用セット 教員用

テキストに関するお問い合わせ

株式会社 アーテック お客様相談窓口
 ◀Webからのお問い合わせはこちら
<https://www.artec-kk.co.jp/contact/>
 お電話でのお問い合わせはこちら
TEL 072-990-5656

B095764 K0424