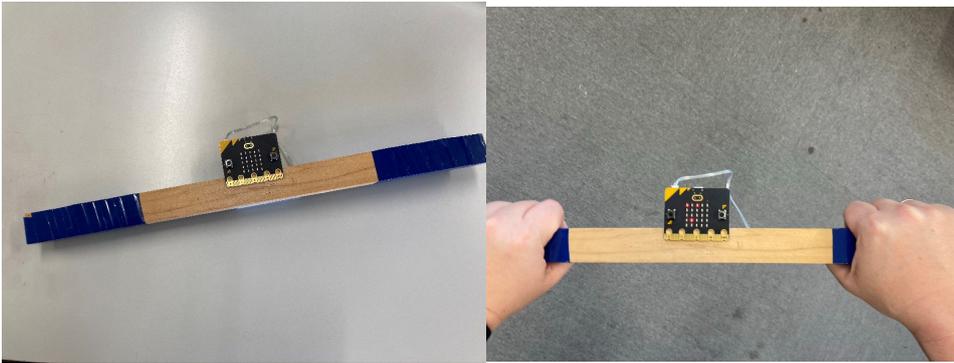


～バイクレースゲーム～

その昔(1990年頃)、ゲームセンターであったバイクゲームをイメージしたマイクロビットプログラムです。加速度センサ(傾きセンサ)を利用して、マイクロビットが左右どちらかに傾いたら、マシン(自機)が動きます。ライバルを追い抜いていくと、どんどんスピードが上がっていきます。ライバルにぶつくと、ゲームオーバーです。

1. 雰囲気を出すための準備

マイクロビットを両面テープなどで棒につけると、バイクのハンドルを握っているような気分になれます。



2. プログラムを実行してみる

どんな動き方で、どんなゲームだったでしょうか？
何をどうすれば、もっとおもしろくなりますか？考えてみましょう。

おもしろくなる追加の^{ようそ}要素をかこう！

→

3. サンプルプログラムを見ながら、仕組みを理解しましょう

サンプルプログラムを改造しながら、どこを直せば、あなたが考えた追加の要素が実現できるでしょうか？いろいろ試してみましょう！

最初だけ

音量を設定する 74

変数 自機 を スプライトを作成 X: 2 Y: 3 にする

変数 難易度 を 100 にする

変数 タイマ を 0 にする

スプライトとは、当たり判定を持ったキャラクタのことです。マイクロビットでは、キャラクタといっても、LEDひとつなのですが(笑)

プログラムリスト (サンプルプログラム)

ずっと

変数 ライバル1 を スプライトを作成 X: 0 から 4 までの乱数 Y: 0 にする

変数 ライバル2 を スプライトを作成 X: 0 から 4 までの乱数 Y: 0 にする

もし ライバル1 の Y が \neq 4 ならくりかえし

もし 加速度 X が < -100 または 加速度 X が > 100 なら

もし 加速度 X が < -100 なら

もし 加速度 X が < -600 なら

自機 の X に 0 を設定する

でなければ

自機 の X に 1 を設定する

+

+

もし 加速度 X が > 100 なら

もし 加速度 X が > 600 なら

自機 の X に 4 を設定する

でなければ

自機 の X に 3 を設定する

+

+

でなければ

自機 の X に 2 を設定する

+

もし タイマ が > 難易度 なら

ライバル1 の Y を 1 だけ増やす

もし 0 = 2 なら

ライバル2 の Y を 1 だけ増やす

+

音を鳴らす 高さ (Hz) 上の下 長さ 1/16 拍

変数 タイマ を 0 にする

もし 難易度 が > 15 なら

変数 難易度 を -1 だけ増やす

+

でなければ

変数 タイマ を 1 だけ増やす

+

もし ライバル1 が他のスプライト 自機 にさわっている または ライバル2 が他のスプライト 自機 にさわっている なら

メロディを開始する パワーダウン くり返し 一度だけ

ゲームオーバーにする

+

一時停止 (ミリ秒) 2

ライバル1 を削除

ライバル2 を削除

ライバル1, 2にぶつからないように自機(バイク)を走らせるイメージです。
1, 2はランダムな位置で、上から下に向かい移動します。

加速度センサにより、傾けると、自分のマシン(自機)が左右に動きます。マイクロビットが、水平に近いと良く反応します。

タイマの数字が難易度より小さければ、ライバル1, 2は動きません。つまり、処理を見送ってゆっくり動くようにしています。

ライバル1, 2を追い抜くと、難易度の数字を1だけ小さくして、タイマの数字と同じ数字になることが速くなります。つまり、処理を見送る時間が短くなるため、ライバル1, 2を追い抜くスピードが速くなり、ぶつかる危険が上がります。

ライバル1, 2が一番下まで来たら、また上から表示するようにします。

プログラムリスト (サンプルプログラム)