

1 前時のうまくいかなかった点をふり返り、本時の課題をつかむ（5分）。

(1) 前時の学習をふり返り、ソフトウェアコントローラーについて知る。

- ・前時のふり返りに、「コントローラーを改良したい」と書いていた児童が多かったことや、災害想定コースの情報収集中に mBot の操縦を交代しようとして止めた理由を取り上げる。
- ・テスト走行時にうまくコントロールできなかった理由の一つに、赤外線リモコンの特徴があることや mBlock で Bluetooth を利用したソフトウェアコントローラーを制作できることを知らせ、コントローラー制作に興味・関心をもたせる。

(2) 本時の課題とループリックを設定する。

- ・「mBot のコントローラーを改良しよう」

→A mBot をスムーズに操作できるソフトウェアコントローラーをプログラムできる

2 スプライトとメッセージの仕組みについて知り、ソフトウェアコントローラーを設計する(10分)。

(1) スプライトとメッセージの仕組みを知る。

- ・スプライトは、現実世界の人やものにあたることを知らせ、それぞれが別々に動いていることを想起する。そうすることでそれぞれが独立したプログラムをもつことを理解させるとともに、各プログラムが連携するために、スプライト間でのメッセージのやり取りが必要になることを理解させる。

(2) ユーザビリティを考慮したユーザーインターフェイス作りの必要性を考え、ソフトウェアコントローラーの設計をする。

- ・災害対応ロボットがどんなときに、誰に使われるのかを考えさせることで、誰もがすぐに使えるようなコントローラーの製作が必要なことに気付かせる。

3 スプライトの追加方法とメッセージを利用したプログラミング方法を知り、ソフトウェアコントローラーのプログラミングをする(20分)。

- ・上矢印ボタンを前進とした場合のスプライトの追加方法と、メッセージを利用したプログラミング方法について説明し、それをもとに他のボタンを追加していけるようにする。

4 班でタブレット端末を交換して操作し、ソフトウェアコントローラーの操作性を確かめ合う。(5分)。

- ・同じ班のペア通しで交換して mBot を操作させることで、誰もが使いやすいものになっているか確かめ合わせる。

5 本時の学習をふり返り、次時の課題について知る（5分）。

- ・ Bluetooth を用いた操作だと 2 台同時に災害想定コースを探索できることを知らせ、次時の活動への意欲を高める。

※ 実際は、朝学習の時間などと合わせ 60 分ぐらいを想定して実践するとよいと思います。45 分で行うなら、2 の設計をとばし、3 では、まず作例通りに作るようにさせるとよいと思います。