

01-2 : 機械学習プログラム実行までの流れ

ここでは、機械学習を用いたロボットアームの仕分けを体験するまでの手順と操作を示します。

※ SPIKE と PC を接続する手段には USB 方式と Bluetooth 方式の 2 つがありますが、本書では USB 方式を示しています。
Bluetooth 接続を行う場合は 70 ページの「Bluetooth プログラム転送方法」をご確認ください。

1 : 機械学習プログラム実行までの全体概要

以下に、実行までの全体手順を示します。その後は、節ごとに各手順を、項ごとに各操作を示します。
各アイコンは各手順で使うハードウェアもしくはソフトウェアです。

サンプルプログラムの移動

a: サンプルプログラムの移動



PC 上でサンプルプログラムを展開し、デスクトップへ移動させます。

ロボットアームの設置

b: 専用マットの設置



専用マットを印刷して用意します。



c: web カメラの設置



web カメラを PC に接続し、ロボットアームとカメラスタンドを専用マットに置きます。



サンプルプログラムの準備

d: SPIKE と PC を USB 接続



SPIKE アプリを起動し、SPIKE と PC を USB 接続します。



e: SPIKE へプログラム転送



SPIKE アプリを用いて、SPIKE へ SPIKE 側のプログラムを転送します。



f: COM ポート確認



PC の設定から COM ポートを確認します。



g: COM ポート設定



Visual Studio Code を起動し、PC 側のサンプルプログラムを開き COM ポートの設定を行います。



サンプルプログラムを実行

h: SPIKE でプログラム実行



SPIKE 本体を操作し、SPIKE 側のサンプルプログラムを実行します。



i: PC でプログラム実行



コマンドプロンプトを起動し、PC 側のサンプルプログラムを実行します。



j: 仕分け実行



ロボットアームの前に仕分けの対象物を置き、コマンドプロンプトを操作して仕分けを実行します。

2: サンプルプログラムの移動

以下に、サンプルプログラムの移動の手順を示します。

※こちらの手順は CHAPTER-02 以降では不要です。

CHAPTER-02 以降では「d: SPIKE と PC を USB 接続」の手順から実行してください。

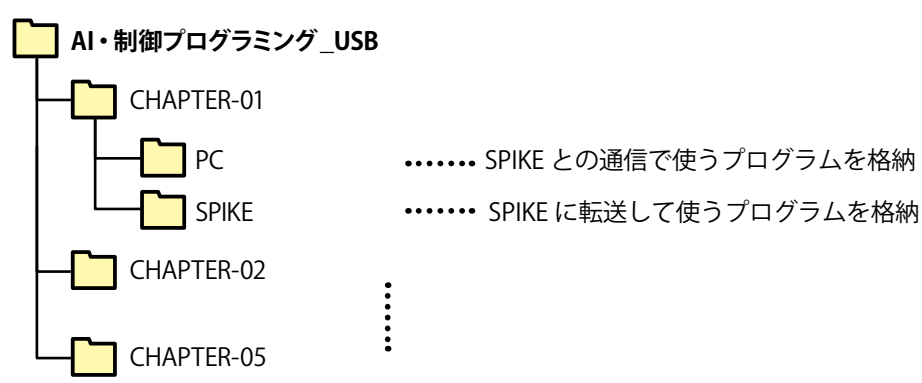
a: サンプルプログラムの移動

PC 上でサンプルプログラムを展開し、デスクトップへ移動させます。

- 1 ダウンロードフォルダにある「AI・制御プログラミング_USB」を展開します。エクスプローラーを開き、「AI・制御プログラミング_USB.zip」ファイルを右クリックし「すべて展開」をクリックします。



- 2 展開先をデスクトップへ設定し、展開します。サンプルプログラムの構成は以下の通りです。
※ここでは、USB 接続をするサンプルプログラムを手順に載せています。Bluetooth 接続の場合は、「AI・制御プログラミング_BT」として操作をお願いします。



3: ロボットアームの設置

以下に、ロボットアームの設置の手順を示します。

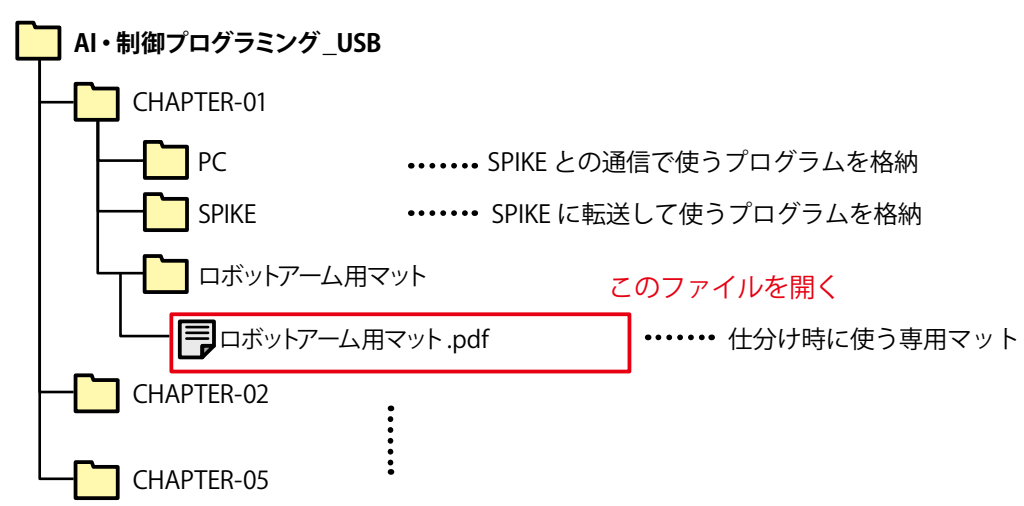
※こちらの手順は CHAPTER-02 以降では不要です。

CHAPTER-02 以降では「d: SPIKE と PC を USB 接続」の手順から実行してください。

b: 専用マットの設置

専用マットを印刷して用意します。

- 1 デスクトップの「AI・制御プログラミング_USB」フォルダの中の「ロボットアーム用マット.pdf」ファイルを開きます。



06-1 : サンプルプログラム解説

本章では、PC 側と SPIKE 側の処理や動作を示しながら、サンプルプログラムの解説を行います。

1 : 通信の流れ

SPIKE 側のサンプルプログラムと PC 側のサンプルプログラムを実行すると、SPIKE と PC で通信を行います。サンプルプログラムの解説の前に、全体のイメージとして PC と SPIKE の通信の流れの全体像を以下に示します。例として、PC 側のサンプルプログラム「sample_PC.py」と SPIKE 側のサンプルプログラム「ONOFF_Control.llsp」を用いて示します。

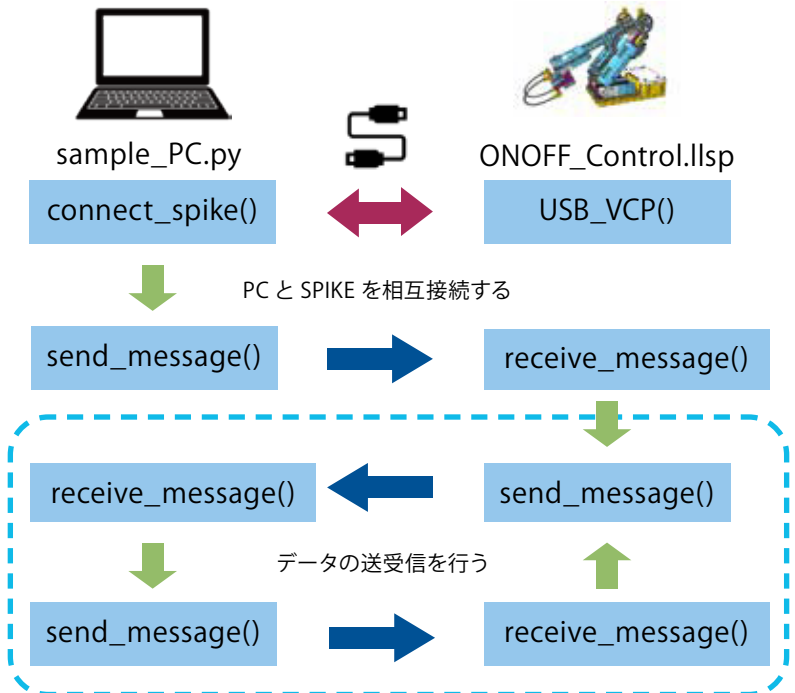
SPIKE 側の「USB_VCP()」関数と PC 側の「connect_spike()」関数が通信を行うプログラムです。

その後、PC からデータを送信する場合は「send_message()」関数を、データを受信する場合は「receive_message()」関数を呼び出します。

SPIKE からデータを送信する場合は「send_message()」関数を、受信する場合は、「receive_message()」関数を呼び出します。

これらのプログラムを繰り返し行うことでデータの送受信を行っています。

コラム：関数とは・・・67 ページ



2 : PC 側のサンプルプログラム

ここでは、PC 側のサンプルプログラムの処理を示し、サンプルプログラムのコードを解説します。

a : sample_PC.py の概要

PC 側のサンプルプログラムの処理の概要を以下に示します。

① SPIKE と通信



sample_PC.py の流れ

SPIKE と通信

SPIKE との通信を確保する。

② 画像データを受信



画像データを受信

web カメラから画像データを受信する。

③ 推論

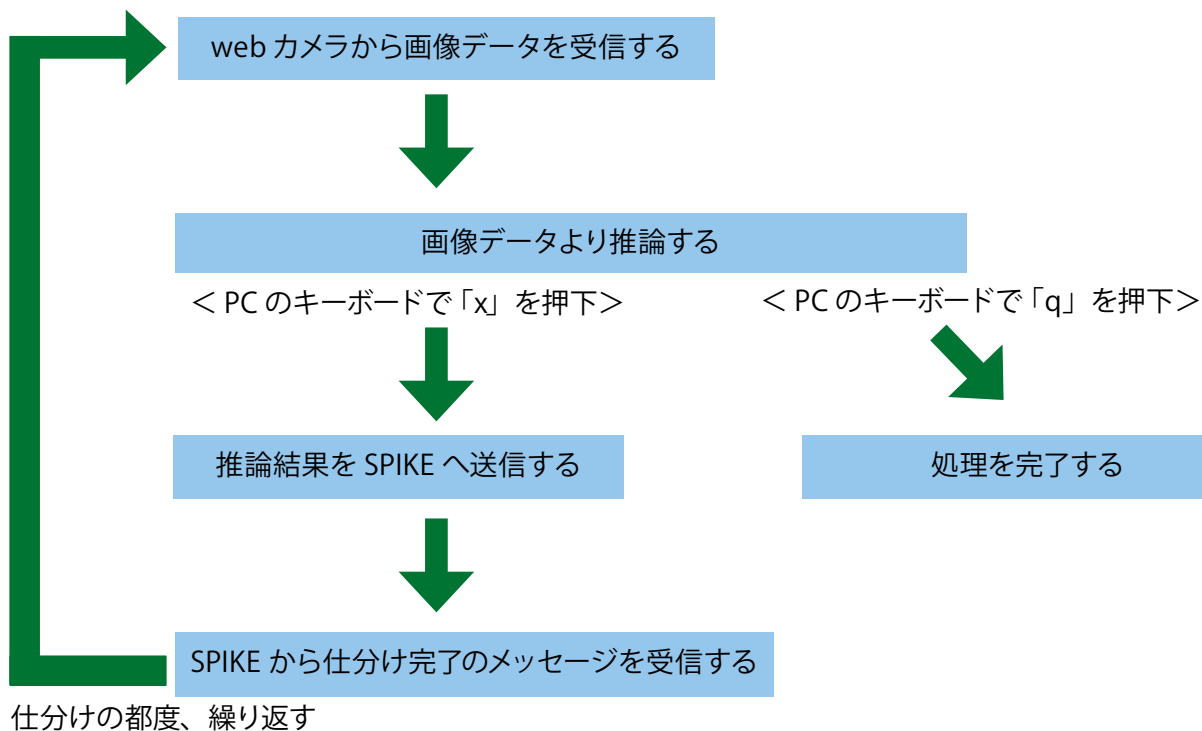


推論

TeachableMachine で作成した機械学習プログラムを使って画像データを推論し、仕分けの都度、推論結果を SPIKE へ送信する。

b : sample_PC.py の処理の流れ

PC 側のサンプルプログラムの処理の流れを以下に示します。

**c : sample_PC.py のソースコード**

PC 側のサンプルプログラムのソースコードを以下に示します。

```

"""
File Name   : sample_PC.py
File Encoding : UTF-8
Copyright © 2022 Afrel Co.,Ltd.
"""

import os
import sys
import serial
import numpy as np
import time
import SPIKE_CONNECTION_MODULES as spike
import tensorflow.keras
from PIL import Image, ImageOps
import cv2
import logging

logging.basicConfig(level = logging.DEBUG, format = '%(asctime)s - %(levelname)s - %(message)s')
#logging.disable(logging.CRITICAL)
logging.debug('START PROGRAM')

```

SPIKE と通信、TeachableMachine で作った機械学習プログラムを利用するために必要なプログラムです。