

## 画像認識応用実験テキスト (WRO ARC 体験)

### ◎実験を始める前に

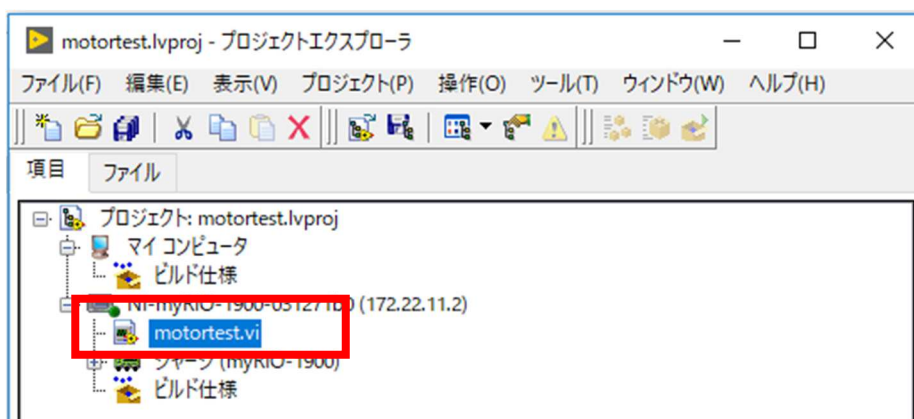
- ・マシンのモータ可動域の確認

(1) 図①のようにデスクトップ上の「motortest」のフォルダを開き、「motortest.lvproj」を開く。



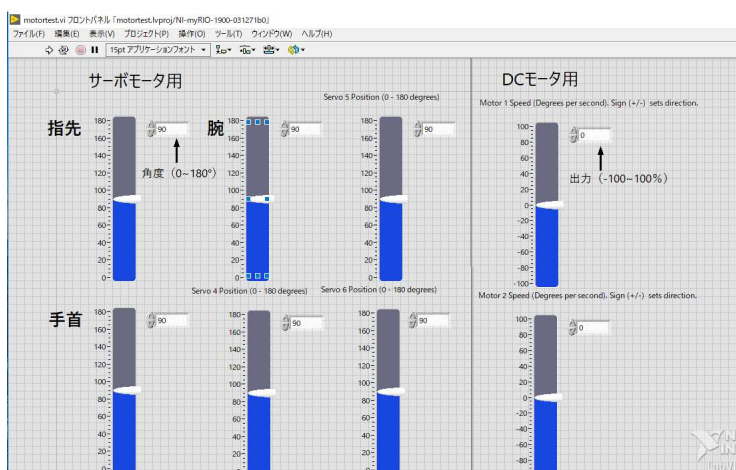
図① motortest フォルダ

(2) 図②に示すようなプロジェクトエクスプローラが表示されたら、NI-myRIO-1900(---...-.-)を展開し、「motortest.vi」をダブルクリックして開く。



図② motortest.lvproj のプロジェクトエクスプローラ

(3) 図③のようなプログラムが起動するので実行し、各モータの可動域の確認や動作確認を行う。



図③ 実行プログラム

### ※注意

サーボモータは負荷をかけすぎたり、かかっている状態で放置すると壊れるので必ず可動範囲内で使用すること。時々、モータに触れ、高温になっていないかを確認しながら使用する。高温になったら、速やかに電源を OFF にし、モータが冷えるまで使用しない。

## 1 実行プログラム

- (1) デスクトップの **ImageProcessing** のフォルダを開き、「**ImageProcessing.lvproj**」を開く。
- (2) 図 1 に示すようなプロジェクトエクスプローラが表示されたら、NI-myRIO-1900(---.---.---)を展開し、「**Move\_cube\_lift.vi**」をダブルクリックして開く。

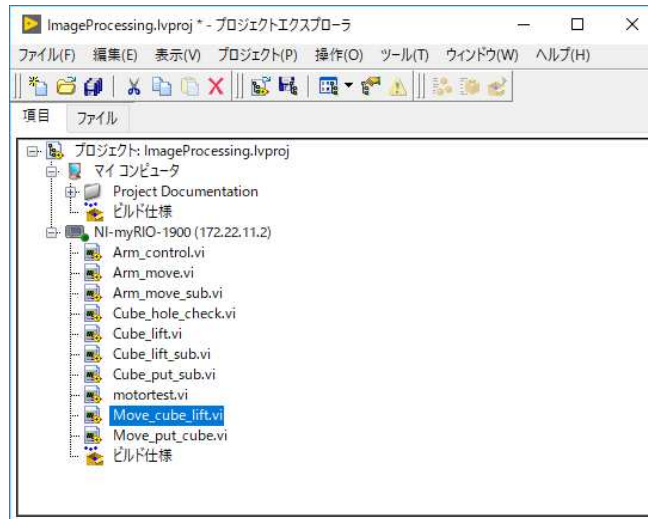


図 1 ImageProcessing.lvproj のプロジェクトエクスプローラ

- (3) 図 2 に示すような「**Move\_cube\_lift.vi**」のフロントパネルが表示されたら、CTRL キーと E キーを同時に押して、図 3 に示すような「**Move\_cube\_lift.vi**」のブロックダイアグラムを表示する。

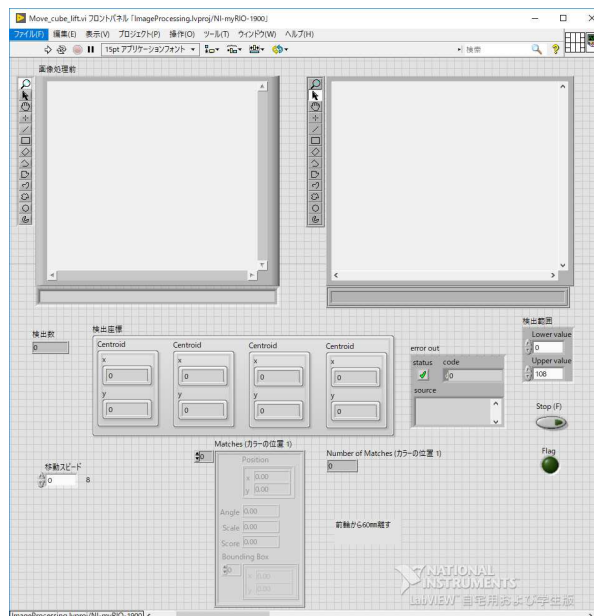


図 2 Move\_cube\_lift.vi のフロントパネル

持ち上げるブロックの画像データを保存するため一度プログラムを実行し、画像データを保存する。方法は色識別のテキスト p.3 と同様に行う。

(4) 図3にはプログラムの設定箇所を記してある。各赤丸で囲った箇所の調整を行い、ブロックの持ち上げプログラムを完成させる。

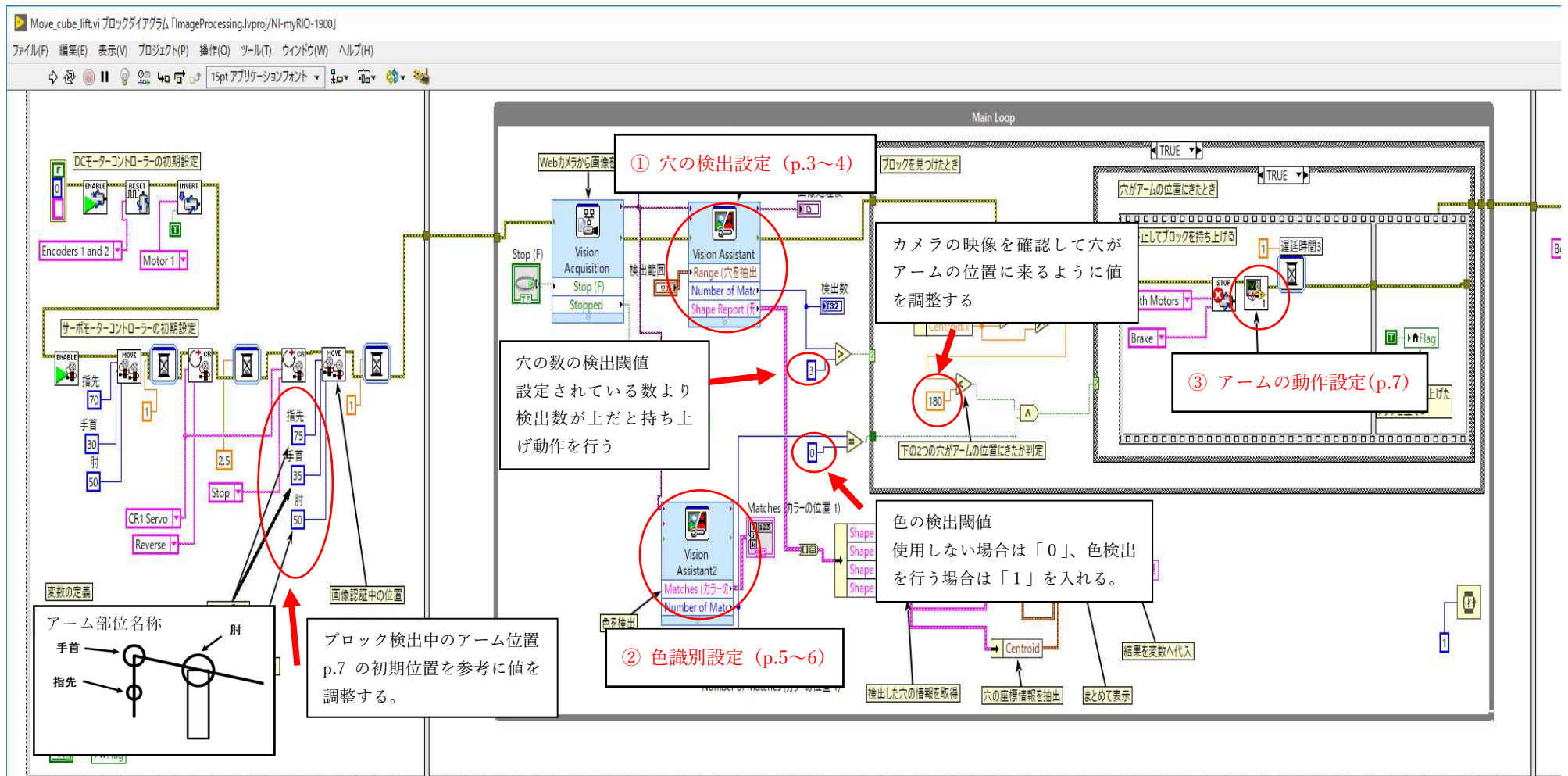


図3 「Move\_cube\_lift.vi」のブロックダイアグラム

## ① 穴の検出設定

(1) 図 3 に示すブロックダイアグラムの赤丸①で囲まれた「Vision Assistant」をダブルクリックして、図 4 に示すような画面を開く。

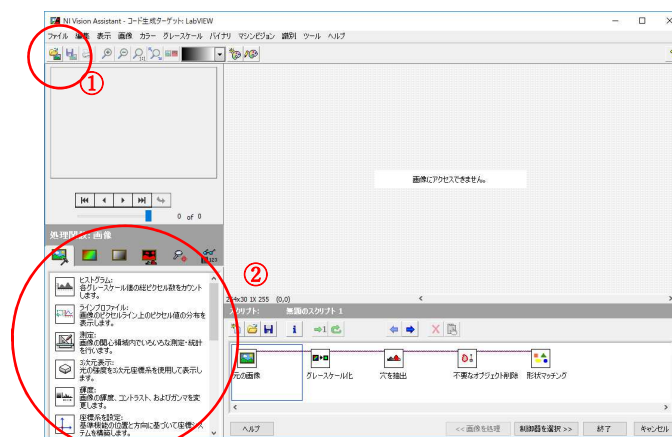


図 4 Vision Assistant の設定画面

(2) メニューバーの「ファイル」から「画像を開く」をクリックし、先ほど保存した画像ファイルを開く（図 4 の①参照）。

あらかじめプログラムを組んでいるので画像を選択した時点で検出は可能になるが、どのような処理を行っているか下記に示す。実験環境によってはパラメータの変更が必要になるのでよく読んでおくこと。

1. 始めに画像のグレースケール化を行う。画面左下の処理関数：カラーのカラープレーン抽出を選択し、「HSV-明度(V)プレーン」に設定し、OK ボタンをクリックする（以下、図 4 の②番参照）。
2. 次に、穴の抽出を行う。画面左下の処理関数：グレースケールの 2 値化を選択し、メインタブの「ステップ名」を任意で「穴を抽出」などの適切な名前に書き換え、2 値化タブの「検索」を暗いオブジェクトにし、「メソッド」を「手動 2 値化」、「ヒストグラムの上限值」を「100」に設定し、OK ボタンをクリックする。
3. 次に、不要なオブジェクトの削除を行う。画面左下の処理関数：バイナリの上級モフォロジーを選択し、「ステップ名」を任意で「不要なオブジェクト削除」などの適切な名前に書き換え、「小さいオブジェクトを削除」を選択し、「反復」を「4」にして、OK ボタンをクリックする。
4. 最後に、形状マッチングを行う。画面左下の処理関数：バイナリの形状マッチングを選択し、「テンプレートを作成」をクリックすると、図 5 に示すような画面が表示される。穴の部分のマウスの左クリックで囲みながら選択した後、OK ボタンをクリックする。すると、テンプレート名を入力する画面が表示されるので、C ドライブ内「sample\_labview」フォルダに任意の名前で保存する。ただし、このときにファイル名を日本語で保存すると、後にエラーが発生する可能性があるため、ファイル名はアルファベットで保存すること。また、「最小スコア」を「700」とし、「スケール影響なし」にチェックを入れ、OK ボタンをクリックする。各パラメータは状況によって最適な値があるので各自、変更すること。

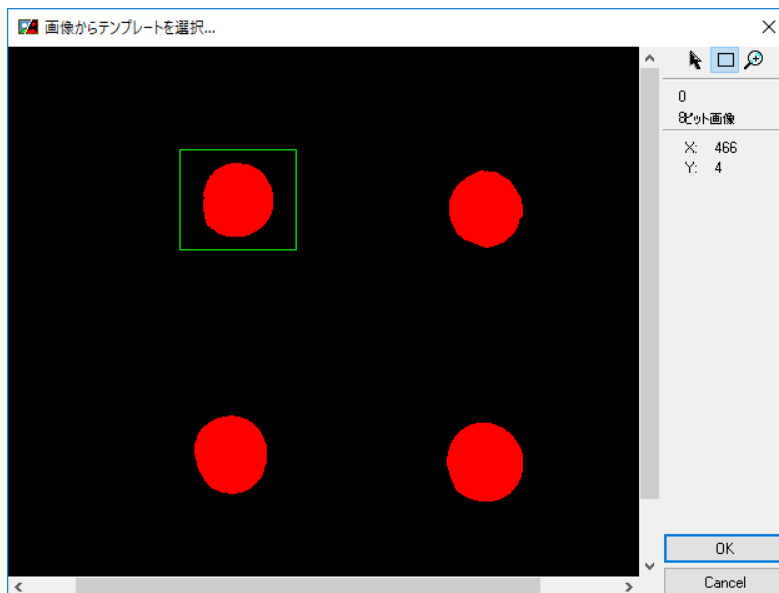


図 5 形状マッチングのテンプレート作成画面

5. また、「制御器の選択」をクリックすると、図 6 に示すような画面が表示される。赤丸の「形状マッチング」の「Number of Matches」と「Shape Report」にチェックを入れ、終了ボタンをクリックする。

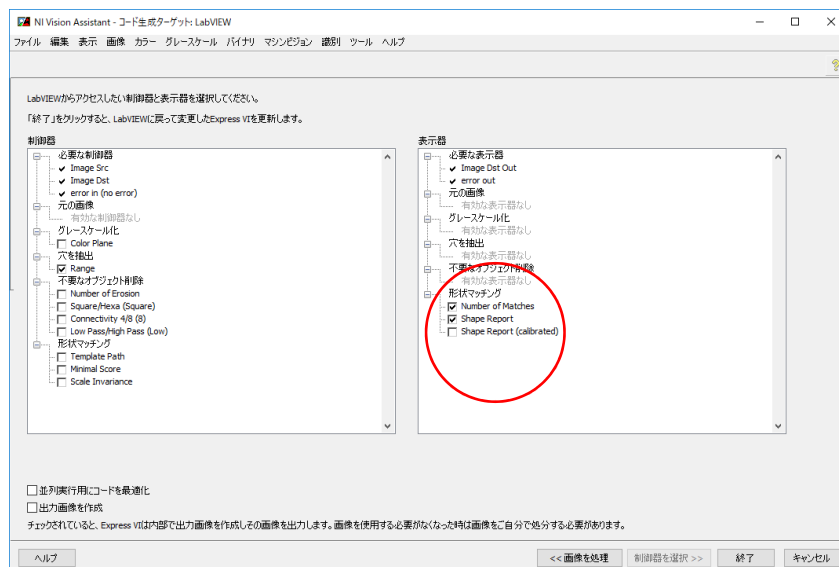


図 6 VissionAssistant の制御器の選択画面

6. Authenticate ウィンドウが開く場合、何も入力せずに Cancel ボタンをクリックする。

## ② 色認識の設定

(1) 図3に示すようなブロックダイアグラムの赤丸(図3の②)で囲まれた「VissionAssistant2」をダブルクリックして、図7に示すような画面を開く。

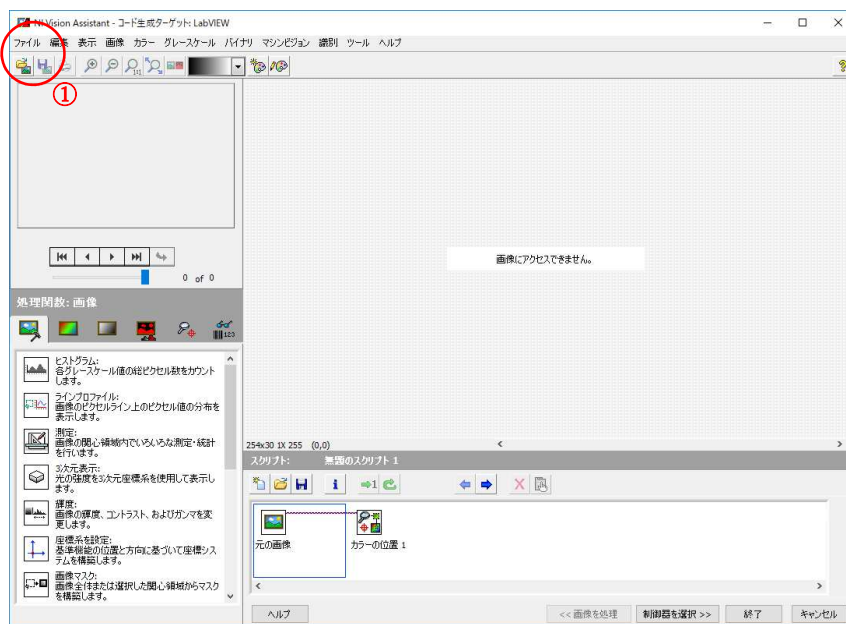


図7 VissionAssistant2の初期画面

(2) メニューバーの「ファイル」から「画像を開く」をクリックし、保存した画像ファイルを開く(図7の①参照)。

(3) ここでは指定した色の識別を行う。画面左下の処理関数: カラーの「カラーの位置」を選択し、テンプレートタブの「テンプレートを作成」をクリックすると、図8のような画面が表示される。ブロックの黄色の部分のマウスの左クリックで囲みながら選択した後、OKボタンをクリックすると、テンプレート名を入力する画面が表示されるので、Cドライブ内「sample\_labview」フォルダに任意の名前で保存する。前述の通り、このときにファイル名を日本語で保存すると、後にエラーが発生する可能性があるので注意すること。

「白と黒を無視」を無視とし、「彩度しきい値」を「80」に設定する。また、設定タブの「検出するマッチ数」を「1」、「最小スコア」を「700」、「検索方法」を「精度優先」、「カラー感度」を「低感度」として、OKボタンをクリックする。各パラメータは状況によって最適な値があるので各自、変更すること。

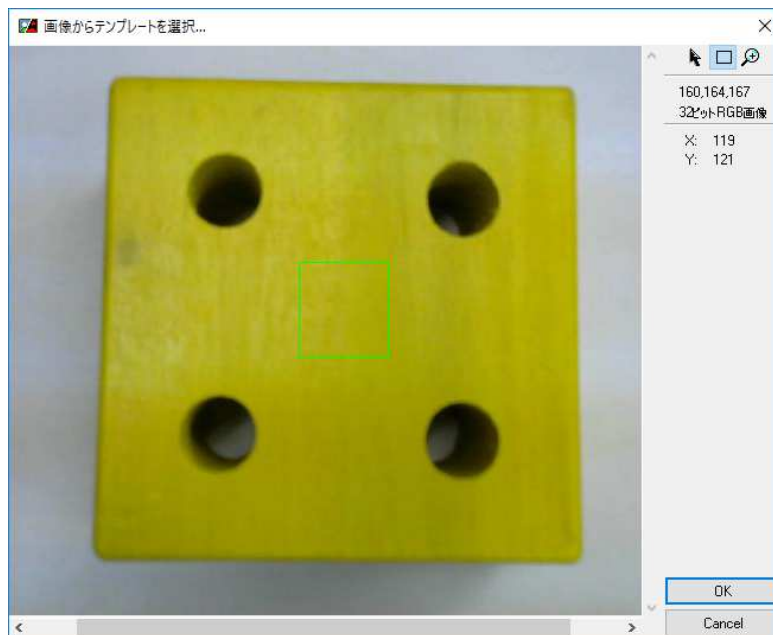


図 8 カラー識別のテンプレート作成画面

(4) また、「制御器の選択」をクリックすると、図 9 に示すような画面が表示される。赤丸の「形状マッチング」の「Match」と「Number of Match」にチェックを入れ、終了ボタンをクリックする。

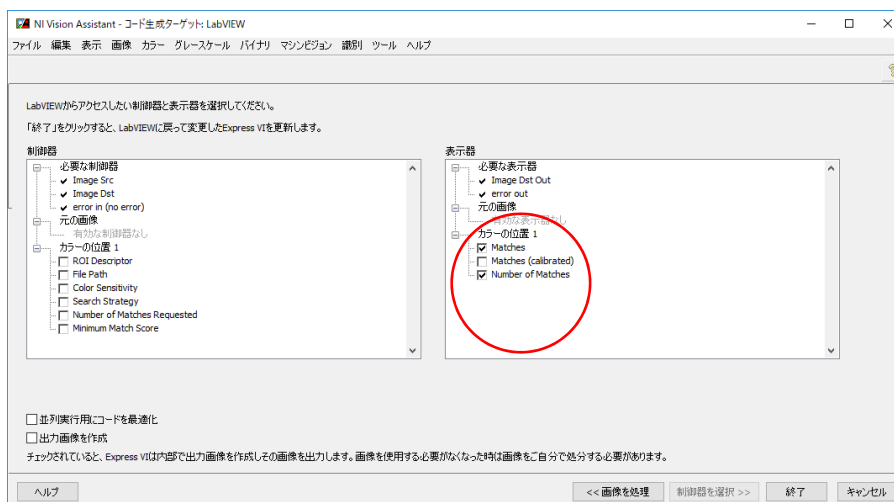


図 9 VissionAssistant2 の制御器の選択画面

(5) Authenticate ウィンドウが開く場合、何も入力せずに Cancel ボタンをクリックする。

### ③ アーム動作設定

図 3 の③のアイコンをダブルクリックで開き、ブロックダイアグラムを開くと図 9 のようなプログラムが開かれる。

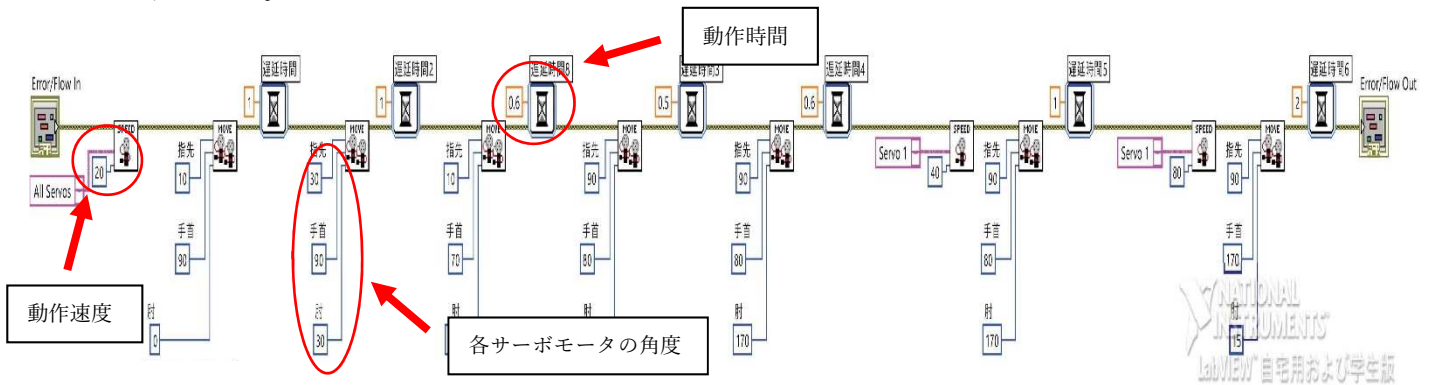


図 9 アーム動作のブロックダイアグラム

上記のプログラムを変更してブロックを持ち上げられるようにする。各マシンによって適切な値が異なるためサーボモータの角度や動作時間、動作速度の調整を行い、必要であればプログラムの追加や削除などを行い、図 10 のアーム動作イメージ図 (①～⑤) を再現できるようにプログラムの調整を行う。もし、マシンの可動域が適切ではない場合は、p.10 以降のサーボモータの可動域調整を参考に調整を行うこと。

#### ・アーム動作イメージ図

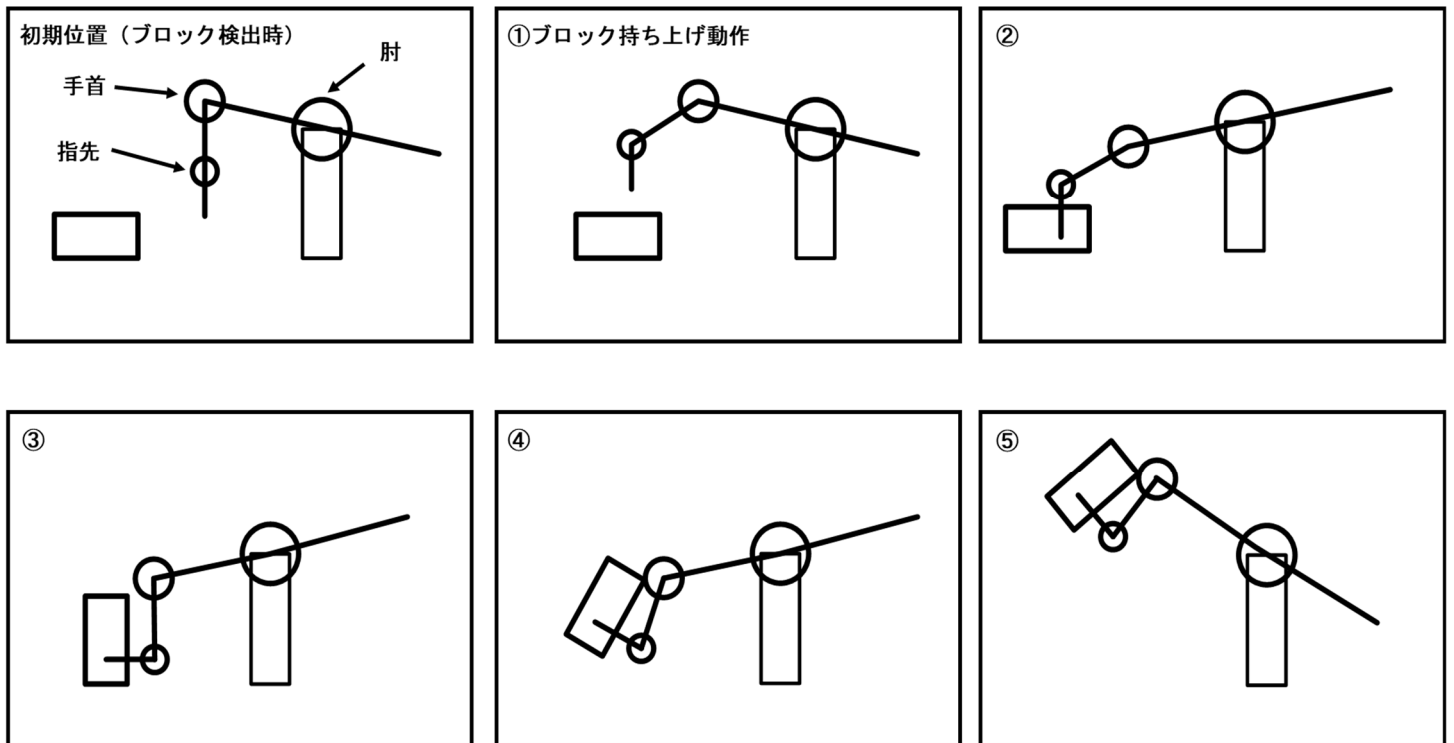


図 10 アーム動作イメージ図



## 2 プログラムの実行

- (1) プログラムを実行するとフロントパネルの画面に図 11 のような画面が表示される。WEB カメラの映像や認識した穴の映像や座標、個数が表示される。

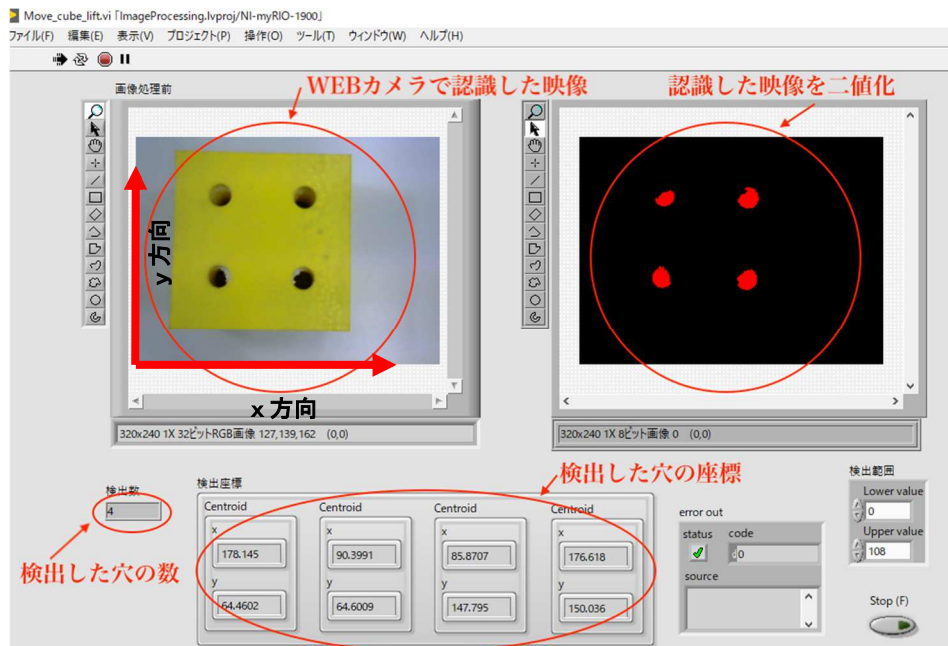


図 11 プログラム実行中のフロントパネルの画面

うまく認識しない場合は「①穴の検出設定」の調整を行う。

- (2) 図 12 のようにブロックを置き、プログラムを実行する。

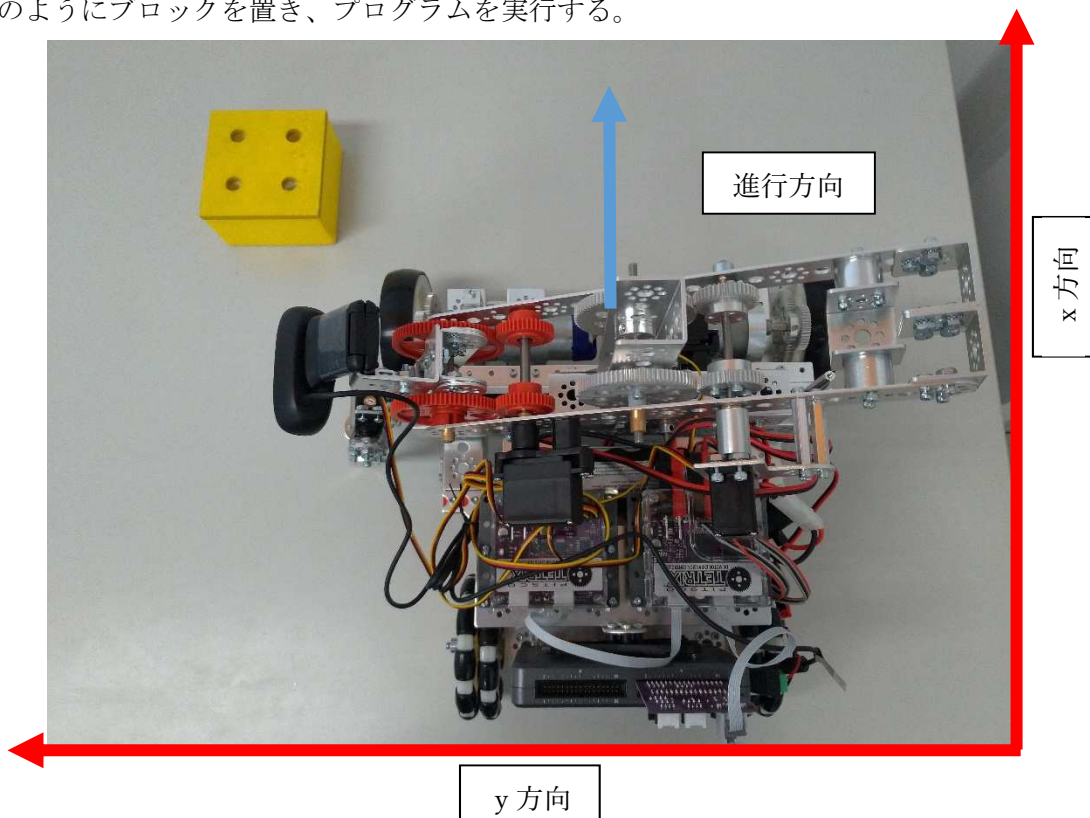


図 12 実験環境

図 13 にマシンの前進方法を示す。移動スピードの値を増やしていくとマシンが前進を始め、穴を検知するまで進み続ける。LabVIEW 上でプログラムを止めても myRIO 内で制御しているため止まらないので挙動が変であったり、うまく静止できない場合はすぐにマシンの電源を切ること。その際は、再び電源を入れ、myRIO とパソコンが正常に接続されてからプログラムを実行すること。また、ギア部分などに指を挟まないように十分注意して行うこと。

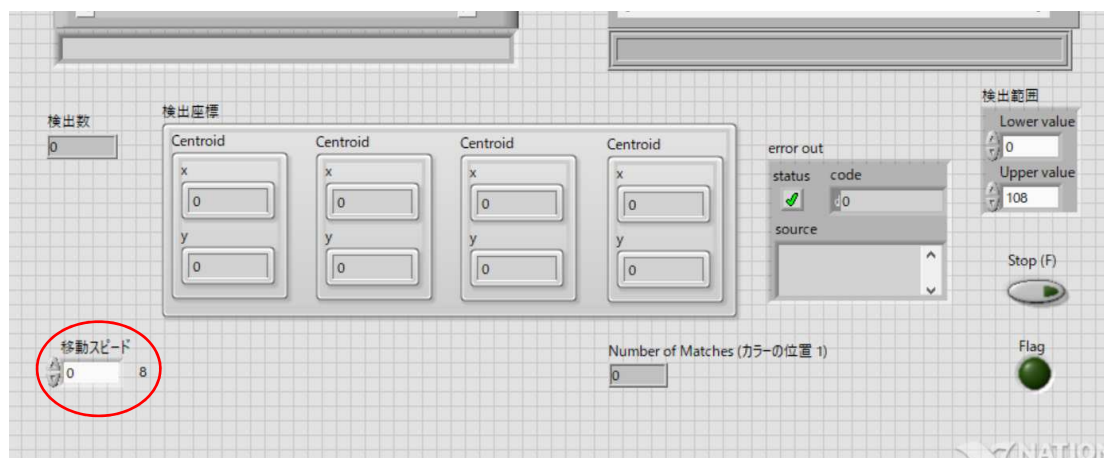


図 13 マシンの前進

(3)持ち上げられない場合はパラメータの調整を行い、再度繰り返す。

### ※注意

サーボモータはプログラム停止後も状態を保持し続けているので、プログラム実行後は TETRIX 本体の電源を一旦、OFF にすること。電源を OFF にすることでサーボモータの状態が解除され、負荷がかからなくなるので忘れずに行うこと。

### 3 サーボモータの可動域調整

◎アーム部のサーボモータは0～180度までが可動域であるため適切な位置で設置をしていないと動作に支障が出るので注意すること。

○肘と手首の位置を調整する方法は以下を参考にすること。

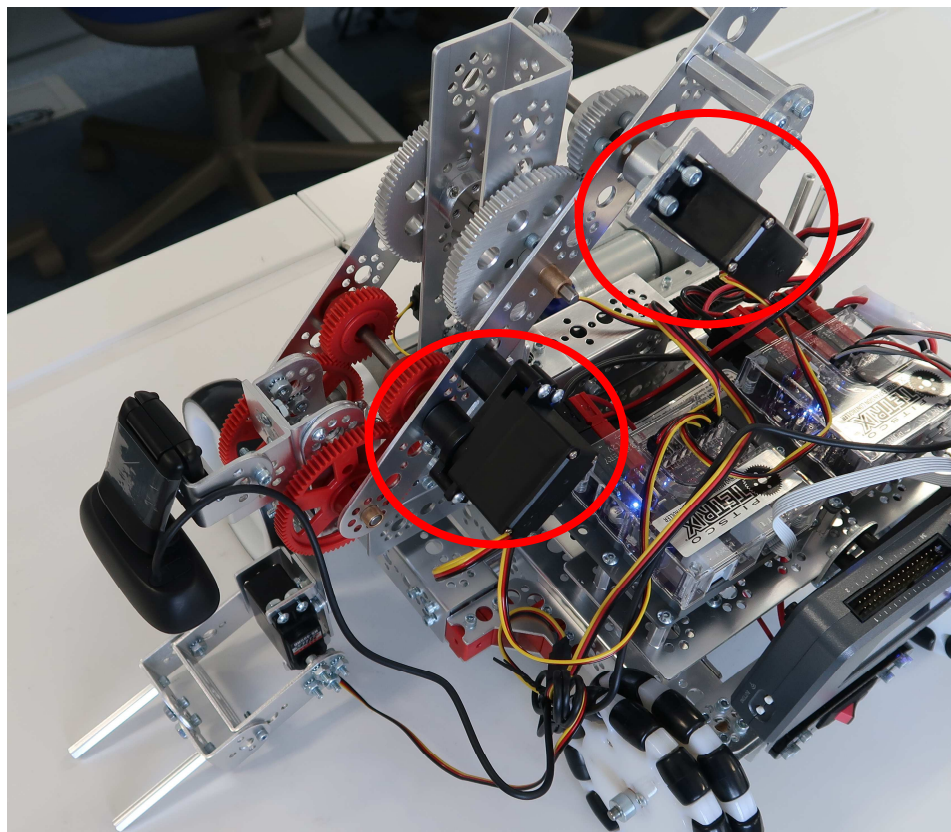


図 14 サーボモータの調整

(1)図 14 の赤丸部分のサーボモータを調整する場合（両方とも方法は同じ）は図 15 のように小さい方のギアのネジを緩める。

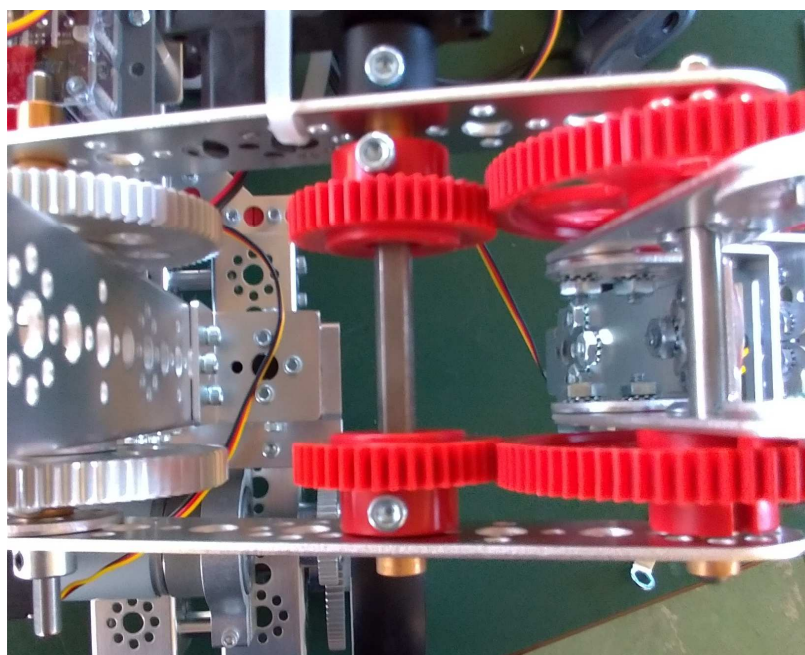


図 15 ギアのネジ

(2)図 16 のようにギアを干渉しない位置までずらし位置を調整後、ギアを元に戻す。

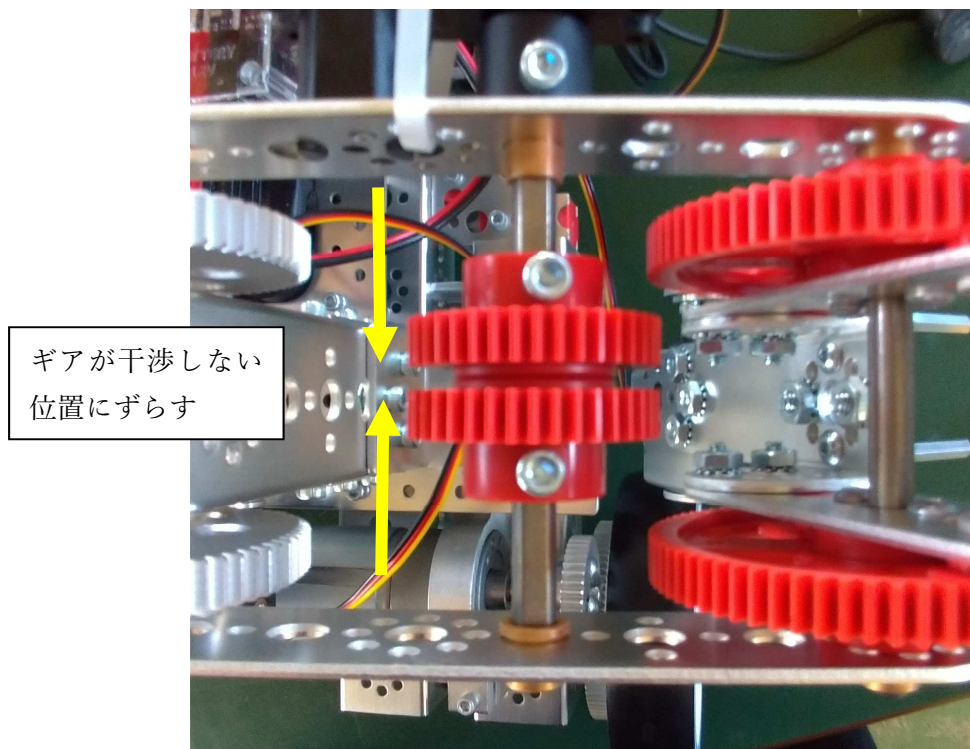


図 16 ギアをずらす

(3)アームの可動域を確認後、問題なければギアを固定する。ギアを固定する場合は図 17 のようにシャフトの平らな面がギアのネジと同じ位置に来るようにしてからネジを締める。

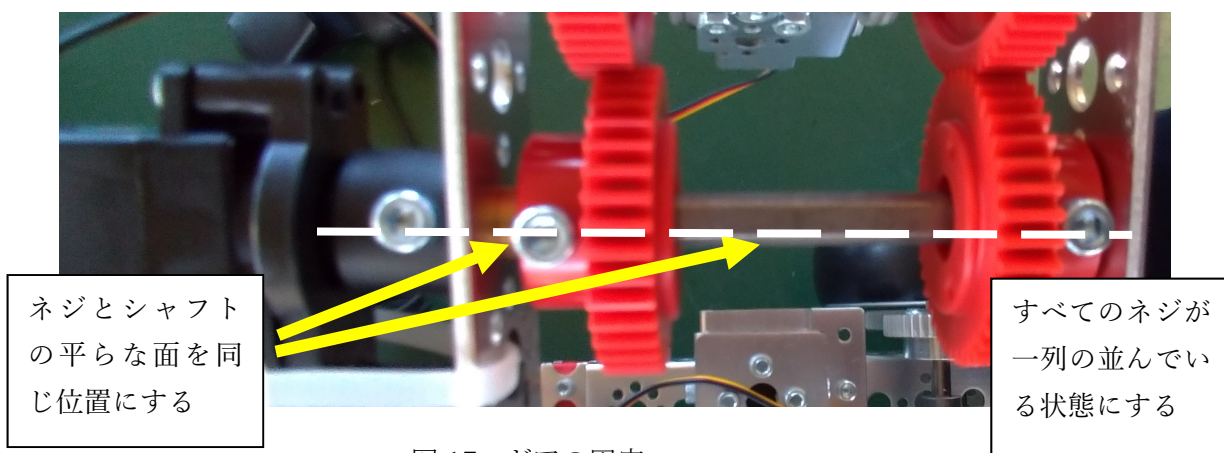


図 17 ギアの固定

○指先の可動域の調整

(1)図 18 のようにサーボモータと金具が接続されている所を矢印の方向へ引っ張る。  
接続部を破壊しないように注意すること。

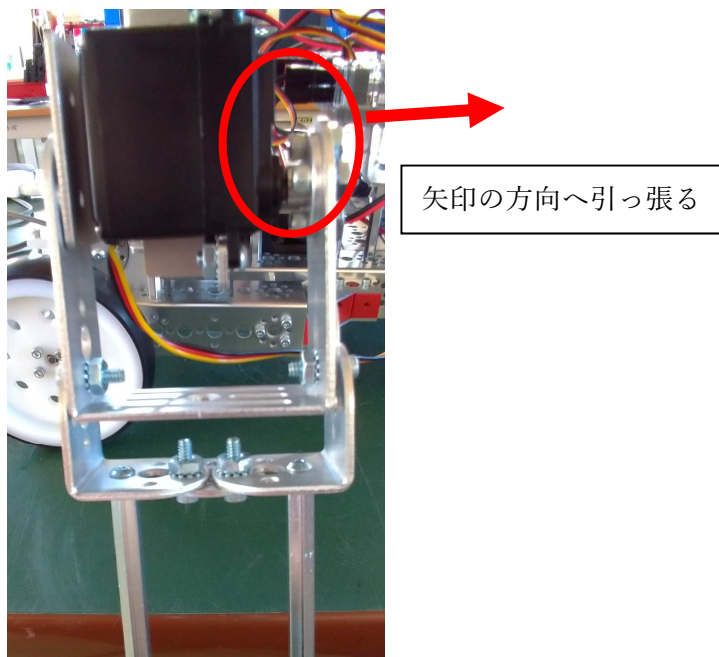


図 18 指先の調整

(2)調整後，金具を接続する。接続する金具は幅がモータ側と同じくらいのため少し外側に広げてから  
はめる。力を入れすぎて金具を変形させないように注意する。

