

EV3Control 資料

ポート名	vel	データ型	TimedVelocity2D									
入力内容	<p>他のコンポーネントから送られた速度の値を受け取ります。 EV3 に与える速度には上限が決められています。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 40%;"></th> <th style="width: 30%;">最小値</th> <th style="width: 30%;">最大値</th> </tr> </thead> <tbody> <tr> <td>vx (前後移動)</td> <td>-0.5 (後退)</td> <td>0.5 (前進)</td> </tr> <tr> <td>va (旋回運動)</td> <td>-8.5 (右旋回)</td> <td>8.5 (左旋回)</td> </tr> </tbody> </table> <p>この範囲以外の値を入れてもモーターに反映されません。</p> <p>※データ上はロボットを横移動させるための値 vy も定義されていますが、EV3 は横移動に対応していないため使用できません。</p>				最小値	最大値	vx (前後移動)	-0.5 (後退)	0.5 (前進)	va (旋回運動)	-8.5 (右旋回)	8.5 (左旋回)
	最小値	最大値										
vx (前後移動)	-0.5 (後退)	0.5 (前進)										
va (旋回運動)	-8.5 (右旋回)	8.5 (左旋回)										
入力の仕方	<p>TimedVelocity2D 型の OutPort 変数に任意の値を入力し、出力します。 入力できる値の最大値、最小値は上記の表の通りです。</p> <p>TimedVelocity2D 型のメンバ変数には、</p> <ul style="list-style-type: none"> • m_ “OutPort 変数名” .data.vx : 前後移動 • m_ “OutPort 変数名” .data.vy : 横移動 • m_ “OutPort 変数名” .data.va : 旋回運動 <p>の3つが定義されています。この変数に値を入れます。 ※Python 環境では、“m_”が”self.d_”になります。</p> <p>例) OutPort 名・変数名 vel のコンポーネントから EV3Control へ前進の値を出力する場合、以下になります。</p> <ul style="list-style-type: none"> ●C++ <pre> m_vel.data.vx = 0.5; m_vel.data.vy = 0; m_vel.data.va = 0; m_velOut.write(); </pre> ●Python <pre> self.d_vel.data.vx = 0.5 self.d_vel.data.vy = 0.0 self.d_vel.data.va = 0.0 self._velOut.write() </pre> 											

ポート名	ultrasonic	データ型	RangeData																		
出力内容	<p>超音波センサーから受け取った値を出力します。</p> <p>EV3用の超音波センサーの仕様は以下になります。</p> <table border="1"> <tr> <td>計測可能距離</td> <td>3cm～250cm</td> </tr> <tr> <td>計測半径</td> <td>約 20 度</td> </tr> <tr> <td>計測精度</td> <td>+/- 1cm</td> </tr> </table> <p>センサーからの値は cm 単位で返されますが、EV3Control では m 単位に変換して出力します。</p> <p>例) センサーが返す値 : 3.0 ~ 250.0 → EV3Control が出力する値 : 0.03 ~ 2.5</p>			計測可能距離	3cm～250cm	計測半径	約 20 度	計測精度	+/- 1cm												
計測可能距離	3cm～250cm																				
計測半径	約 20 度																				
計測精度	+/- 1cm																				
受け取り方	<p>受け取ったデータのメンバ変数 (配列) [ranges] の 0 番目に超音波センサーからの値が受信されます。設定した閾値と比較して実行する処理を作成してください。</p> <p>詳しい利用方法は下記の「EV3Control_Sample_超音波センサー」を参考にしてください。</p>																				
EV3Control_Sample_超音波センサー																					
C++用 : https://rtc-fukushima.jp/wp/wp-content/uploads/2017/03/1707-1co_files.zip																					
Python 用 : https://rtc-fukushima.jp/wp/wp-content/uploads/2018/10/PSub1.zip																					
ポート名	color	データ型	TimedString																		
出力内容	<p>カラーセンサーから受け取った値を出力します。</p> <p>カラーセンサーは認識した色を値に変換して返します。</p> <p>EV3に搭載されているカラーセンサーの色と値の対応表は以下になります。</p> <table border="1"> <thead> <tr> <th>色</th> <th>値</th> </tr> </thead> <tbody> <tr> <td>なし</td> <td>0</td> </tr> <tr> <td>黒</td> <td>1</td> </tr> <tr> <td>青</td> <td>2</td> </tr> <tr> <td>緑</td> <td>3</td> </tr> <tr> <td>黄色</td> <td>4</td> </tr> <tr> <td>赤</td> <td>5</td> </tr> <tr> <td>白</td> <td>6</td> </tr> <tr> <td>茶</td> <td>7</td> </tr> </tbody> </table> <p>※値は int 型ではなく string 型で取得されます。</p>			色	値	なし	0	黒	1	青	2	緑	3	黄色	4	赤	5	白	6	茶	7
色	値																				
なし	0																				
黒	1																				
青	2																				
緑	3																				
黄色	4																				
赤	5																				
白	6																				
茶	7																				

受け取り方	<p>値を受け取る際、受信先のコンポーネントで受け取ったデータを(const char*)でキャストする必要があります。これは、CORBA を使用して C++の string 型データを送信する際に char*型に変換する必要があるためです。</p> <p>値は受信したデータのメンバ変数[data]に保持されています。</p> <p>※Python 環境で使用する際は、(const char*)でキャストする必要はありません。</p> <p>詳しい利用方法は下記の「EV3Control_Sample_カラーセンサー」を参考にしてください。</p>
-------	--

EV3Control_Sample_カラーセンサー

C++用 : https://rtc-fukushima.jp/wp/wp-content/uploads/2017/03/1715-1co_files.zip

Python 用 : <https://rtc-fukushima.jp/wp/wp-content/uploads/2018/10/PSub3.zip>

ポート名	touch	データ型	TimedBooleanSeq
------	-------	------	-----------------

出力内容	<p>タッチセンサーから受け取った値を出力します。</p> <p>EV3 に搭載されているタッチセンサーの返す値は以下になります。</p> <table border="1" style="margin-left: 20px;"> <tr> <th style="background-color: #cccccc;">センサー状態</th> <th style="background-color: #cccccc;">値</th> </tr> <tr> <td>Released</td> <td>0</td> </tr> <tr> <td>Press</td> <td>1</td> </tr> </table>	センサー状態	値	Released	0	Press	1
センサー状態	値						
Released	0						
Press	1						

受け取り方	<p>受け取ったデータのメンバ変数（長さが 2 の配列） [data]として送られてきます。</p> <p>data[0]には左側タッチセンサーの値、data[1]には右側タッチセンサーの値が保持されています。</p> <p>詳しい利用方法は下記の「EV3Control_Sample_タッチセンサー」を参考にしてください。</p>
-------	--

EV3Control_Sample_タッチセンサー

C++用 : https://rtc-fukushima.jp/wp/wp-content/uploads/2017/03/1711-1co_files.zip

Python 用 : <https://rtc-fukushima.jp/wp/wp-content/uploads/2018/10/PSub2.zip>