

```
1 //-----
2 /*! ¥file
3 ¥brief robot simulator protocol
4 ¥author Akihiko Yamaguchi
5 ¥date Mar.13 2007 */
6 //-----
7
8 const char *SOCK_NAME = "/tmp/octrobot-socket"; //!< 通信に使うソケットファイル
9
10 // サーバ側が受け付けるコマンド
11 const int ORS_START_SIM (0); //!< シミュレーションを開始する
12 const int ORS_STOP_SIM (1); //!< シミュレーションを終了する
13 const int ORS_STEP_SIM (2); //!< シミュレーションを 1 ステップ進める (dvalue=時間ステップ)
14 const int ORS_RESET_SIM (3); //!< シミュレーションリセットする
15 const int ORS_DRAW_WORLD (4); //!< 世界を描画
16 const int ORS_SET_TORQUE (5); //!< トルク入力を設定 (step=0... 関節数-1, dvalue)
17 const int ORS_SET_WINDOWSIZE (6); //!< 画面サイズを変更. デフォルト 400x400 (step=0:x, step=1:y, ivalue). ORS_START_SIM よりも前に使わ
    ないと意味がない.
18 const int ORS_GET_JOINT_NUM (7); //!< 関節数を返す. (int*1)
19 const int ORS_GET_JSTATE_DIM (8); //!< 関節状態の次元を返す. (int*1)
20 const int ORS_GET_BSTATE_DIM (9); //!< ベース状態の次元を返す. (int*1)
21 const int ORS_GET_JOINT_STATE (10); //!< 関節状態を取得 (double*関節状態ベクトル次元のデータが返される)
22 const int ORS_GET_BASE_STATE (11); //!< ベース状態を取得 (double*ベース状態ベクトル次元のデータが返される)
23
24 struct TXData //! 通信に使うデータ
25 {
26     int command;
27     int step;
28     union
29     {
30         int ivalue;
31         double dvalue;
32     };
33 };
34
35
36
37
```