関数

オリジナルの関数について以下に示す。

int main()　　各関数を呼び出し、ゴールする。

path_create()を呼び出す。(first_touch=1)

ループ　path_create()を呼び出す。

number に final_number()の値を入れる。

number= -1 なら、再画像認識ループへ

ループ　cameraNO_create()を呼び出す。

number に final_number()の値を入れる。

number= -1 でないならループを抜ける。

指定回数認識しても number= -1 なら number=ERR とし、ループを抜ける。

cameraOK_create()を呼び出す。

switch(number)　[0]なし、[1]key_create()の呼び出し、[2]なし

[3]roomcheck_create()の呼び出しと room_create()の呼び出し

[4]rotarycheck_create()の呼び出しと rotary_create()の呼び出し

[5]goal_create()の呼び出し、[6]error_create()の呼び出し、[7]なし

---

void *cameraOK_motor_thread(void *arg)　画像認識に成功した後に呼び出され、camera_end()を呼び出す。

camera_end()を呼び出す。

motor_thread_end を 1 にして、スレッドを終了する。

---

void *cameraNO_motor_thread(void *arg)　画像認識に失敗した後に呼び出され、後退と通路走行をする。

後退する。

ループ　白線センサ　　白線検知したら seitai()を呼び出し、正対が完了したら(seitai_finish=1 なら)ループを抜ける。

タッチセンサ　・正面タッチしたら後退し、左右の壁で WALL_LR1 以上の方向へ 90°旋回する。

・左右タッチしたら後退し、曲線後退する。

PD 走行　　左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。

motor_thread_end を 1 にして、スレッドを終了する。

---

void *error_motor_thread(void *arg)画像認識に指定回数失敗したときに呼び出され、壁づたい走行をする。

ループ　白線センサ　　白線検知したら後退し、ループを抜ける。

タッチセンサ　・正面タッチしたら後退し、左右の壁が WALL_LR1 以下なら 180°旋回をし、左右の壁で WALL_LR1 以上ならその方向へ 90°旋回をする。

・左右タッチしたら後退し、曲線後退する。

片側 PD 走行　ERROR_WALL による一方の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる

motor_thread_end を 1 にして、スレッドを終了する。

---

void camera_end()　　画像認識終了後の走行をする。

前進する。

ループ　タッチセンサ　前進し、正面タッチしたらループを抜ける。

後退する。

90°旋回をする。

前進する。

左右どちらかの白線を検知したら、number=8,camera=-1 にする。

int goal_performance()　　白線を検知し、ゴールエリア内で停止し、パフォーマンスをする。

ループ　白線センサ　　・左右検知してないときは前進する。
　　・片方検知したら曲線後退する。
　　・両方検知したら停止し、ループを抜ける。

ループ　白線センサ　　両方検知している間、前進する。

後退する。

パフォーマンスをする。

---

void io_check()　　I/O エラーをチェックする。

s1 か s2 が -1 ならば、　io_error へ 1 を代入する。
　　motor_thread_end へ 1 を代入する。(ts_ws_thread と uss_thread の終了)
　　motor_thread を強制終了する。
　　スレッドの終了を待つ sleep

---

void io_restart()　　./motor_restart を実行する。

io_error が 1 ならば、　motor_restart を実行する。
　　s1 と s2 へ 0 を代入する。
　　sleep

---

void motor_rotate(int arg)　　引数の角度だけその場で回転走行をする。

どーなってんの？

---

int seitai(void)　　白線センサと前の二つの超音波センサで正対補正走行をする。

ループ　白線センサ　　・左右検知してないときは後退し、ループを抜ける。
　　・片方検知したら、検知していない方のタイヤを前進する。
　　・両方検知したら停止し、ループを抜ける。

白線を両方検知してないときは終了

ループ　超音波センサ　　・前のどちらかが WALL_F2 以上なら、run_back へ 1 を代入してループを抜ける。
　　・前の値の差が 1cm 以内なら、停止し、ループを抜ける。
　　・前の値の大きい方のタイヤを前進する。

run_back が 1 なら　　前進し，左右どちらかの白線を検知したら、number=8,camera=-1 にして終了する。
　　前進,回転,回転

---

void *ts_ws_thread(void)　　タッチセンサと白線センサの値を読み取る。

ループ　白線センサの値を line_judge[0.1]へ入れる。
　　タッチセンサの値を touch_judge[0.1.2.3]へ入れる。
　　first_touch=1 で、正面がタッチされた場合、motor_thread_end を 1 にしてスレッドを終了する。
　　motor_thread_end=1 なら、ループを抜ける。

スレッドを終了する。

---

void *uss_ thread(void *a)　　超音波センサの値を読み取る。

ループ　first_touch=1 のとき、待機ループへ
　　ループ　motor_thread_end=1 なら、スレッドを終了する。
　　超音波センサの値を uss_judge[0.1.2.3]へ入れる。
　　motor_thread_end=1 なら、ループを抜ける。

スレッドを終了する。

void *goal_motor_thread(void *arg)認識値が 5 のときに呼び出され、goal_performance()を呼び出す。
ループ　白線センサ　　白線検知したらループを抜ける。
　　タッチセンサ　　・正面タッチしたら後退し、左右の壁で WALL_LR1 以上の方向へ 90°旋回する。
　　・左右タッチしたら後退し、曲線後退する。
　　PD 走行　　左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。
goal_performance()を呼び出す。
motor_thread_end を 1 にして、スレッドを終了する。

void *key_motor_thread(void *arg)認識値が 1 のときに呼び出され、鍵と壁づたい走行をする。
ループ　PD 走行(key_push=0)　　超音波センサとエンコーダより、モーターへの入力を変化させる。
　　片 PD 走行(key_push=1)　direction による一方の超音波センサとエンコーダより、モーターへの入力を変化させる
　　白線センサ　　白線検知したら後退し、ループを抜ける。
　　タッチセンサ(key_push=0)　　正面タッチしたら後退、180°旋回し、key_push に 1 をいれる。
　　タッチセンサ　　・key_push=1 で、正面タッチしたら後退し、左右の壁が WALL_LR1 以下なら 180°旋回をし、左右の壁で WALL_LR1 以上ならその方向へ 90°旋回をする。
　　・左右タッチしたら後退し、曲線後退する。
motor_thread_end を 1 にして、スレッドを終了する。

int final_number(void) 画像認識をするときに呼び出され、数値の読み取りと曲がる方向の設定をする。
switch(num_l)　　読み取った値から優先度を設定する。
switch(num_r)　　読み取った値から優先度を設定する。
優先度を比較し、曲がる方向を direction へ入れる。
曲がる方向の数値を返し、終了する。

void *path_motor_thread(void *arg)通路走行をするときに呼び出され、通路走行をし seitai()を呼び出す。
first_touch=1 のとき、待機ループへ
　　ループ　motor_thread_end=1 なら、スレッドを終了する。
ループ　白線センサ　　白線検知したら seitai()を呼び出し、正対が完了したら(seitai_finish=1 なら)ループを抜ける。
　　タッチセンサ　　・正面タッチしたら後退し、左右の壁で WALL_LR1 以上の方向へ 90°旋回する。
　　・左右タッチしたら後退し、曲線後退する。
　　PD 走行　　左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。
motor_thread_end を 1 にして、スレッドを終了する。

void *room_motor_hread(void *arg)小部屋攻略するときに呼び出され、出口を探しながら走行する。
CLOCKWISE_ROOM の値と room_in の値によって、90°旋回する。
ループ　白線センサ　　白線を検知したら後退し、motor_thread_end を 1 にして、スレッドを終了する。
　　片側 PD 走行　　CLOCKWISE_ROOM による一方の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる
　　超音波センサ　　出口を見つけるか、通路にいると判断したら、ループを抜ける。
　　タッチセンサ　　・正面タッチしたら後退し、部屋の隅なら、90°旋回する。出口がある場合は 90°旋回し前進してループを抜ける。
　　・左右タッチしたら後退し、曲線後退する。
motor_thread_end を 1 にして、スレッドを終了する。

void *roomcheck_motor_hread(void *arg) 認識値が 3 のときに呼び出され、小部屋を探しながら走行する。		
ループ	白線センサ	白線を検知したら後退し、motor_thread_end を 1 に、room_in を-1 にして スレッドを終了する。
	タッチセンサ	・正面タッチしたら後退し、左右の壁で WALL_LR1 以上の方向へ 90°旋回する。 ・左右タッチしたら後退し、曲線後退する。
	PD 走行	左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。
	超音波センサ	小部屋かどうか、小部屋のどこから入ったかを判断し room_in へ値を入れる。 小部屋に入った場合(room_in=1.2.3)ループを抜ける。
motor_thread_end を 1 にして、スレッドを終了する。		
void *rotary_motor_hread(void *arg)ロータリー攻略するときに呼び出され、出口を探しながら走行する。 CLOCKWISE_ROTARY の値によって、曲線前進する。		
ループ	白線センサ	白線を検知したら後退し、motor_thread_end を 1 にして、スレッドを終了する。
	タッチセンサ	・正面タッチしたら後退する。 ・左右タッチしたら後退し、曲線後退する。
	PD 走行	左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。
	超音波センサ	左右の壁で、PATH_LR2 以上の方向へ 90°旋回し、前進してループを抜ける。
motor_thread_end を 1 にして、スレッドを終了する。		
void *rotarycheck_motor_hread(void *arg) 認識値が 4 のときに呼び出されロータリーを探しながら走行。		
ループ	白線センサ	白線を検知したら後退し、motor_thread_end を 1 に、rotary_in を-1 にして スレッドを終了する。
	タッチセンサ	・正面タッチしたら後退し、左右両方の壁が SIDE_LR 以上ならループを抜ける。 左右の壁で WALL_LR1 以上の方向へ 90°旋回する。 ・左右タッチしたら後退し、曲線後退する。
	PD 走行	左右の超音波センサの値とエンコーダの値より、モーターへの入力を変化させる。
motor_thread_end を 1 にして、スレッドを終了する。		