

ZCHEDULER SIDE OF TRIP PLAN

Updated 9/9/2024

COMBINED MOTIONS

The “simple” process of driving from one location to another is a complex sequence of steps, some not so obvious, that must be performed in the correct order. Transitioning through the different types of intersections requires different rules for each type of intersection.

Each general task of driving requires multiple steps to complete in the proper order. The Trip Plan generator is always creating a motion command for the current road section, but also has to create all the extra moves that are required to get the Zoomie at the starting point of the next road.

BACKING OUT OF THE DRIVEWAY

To back out of a driveway, the following steps must be performed and are all created within the BOD_2 function.

1. Add_Instruction_2("BEG",0,"0",0,0,0,0,DW_X, DW_Y)
2. Add_Instruction_2("STP",Dist_2_Travel,"---",0,0,0,0,0,DW_X, DW_Y)
3. Add_Instruction_2("BOD",Dist_2_Travel,"---",0,0,0,0,BU_Heading,X_EP,Y_EP)
4. Add_Instruction_2(Piv_Turn_Dir,0,"REV",0,0,0,0,C_Heading,X_EP, Y_EP)

DRIVING FROM ONE BYPASS ROAD TO ANOTHER

To continue from one bypass road section to another, a sweeping turn needs to be performed at the end of the first road section.

1. Add_Instruction_2("FLS",C_Abs_End,"---",Pre_Set,0,0,C_Road_No,C_Heading,C_XE,C_YE)
2. Add_Instruction_2("FLM",Dist,"---",0,0,0,0,N_Heading,N_XS,N_YS)

MIDROAD ENTRY FOLLOWED WITH A TURN INTO THE START OF THE NEXT ROAD

When a road is entered somewhere in the middle, a preset position action is required. Additionally the road is traversed with FLS, and STP is performed before making the turn towards the next road, and finally FLD finishes the drive to get to the start of the next road.

1. Add_Instruction_2("FLS",Turn_Pos,"-w-",Pre_Set,0,0,C_Road_No,C_Heading,G_X_Pos,G_Y_Pos)
2. Add_Instruction_2("STP",0,"-w-",0,0,0,0,C_Heading,G_X_Pos, G_Y_Pos)
3. Add_Instruction_2(Turn_Dir,0,"FWD",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)
4. Add_Instruction_2("FLD",Extend,"-w-",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)

DRIVE TO END OF ROAD, TURN INTO THE MIDDLE OF THE NEXT ROAD

The current road is driven to the end, then a STP is performed before proceeding across the traffic lanes with the FLD command and finally, a PTL/PTR is performed to get the Zoomie heading in the correct direction.

1. Add_Instruction_2("FLS",Stop_Pos,"A--",Pre_Set,0,0,C_Road_No,C_Heading,G_X_Pos,G_Y_Pos)
2. Add_Instruction_2("STP",0,"---",0,0,0,0,C_Heading,G_X_Pos, G_X_Pos)
3. Add_Instruction_2("FLD",Extend,"A--",0,0,0,0,C_Heading,G_X_Pos,G_Y_Pos)
4. Add_Instruction_2(Turn_Dir,0,"FWD",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)

DRIVE TO END OF ONE ROAD, THEN STRAIGHT ACROSS INTERSECTION TO THE NEXT

The current road is driven to the end and a STP is performed before driving across the traffic lanes using the FLD command.

1. Add_Instruction_2("FLS",Stop_Pos,"---",Pre_Set,0,0,C_Road_No,C_Heading,G_X_Pos,G_Y_Pos)
2. Add_Instruction_2("STP",0,"---",0,0,0,0,C_Heading,G_X_Pos, G_X_Pos)
3. Add_Instruction_2("FLD",Cross_Intersection_Dist,"---",0,0,0,0,C_Heading,G_X_Pos,G_Y_Pos)

DRIVE TO END OF CURRENT ROAD THEN TURN INTO THE START OF NEXT

This requires a turn at the end of the current road, then a pivot turn and finally a FLD to move to the start of the next road.

1. Add_Instruction_2("FLD",Extend,"---",0,0,0,0,C_Heading,G_X_Pos,G_Y_Pos)
2. Add_Instruction_2(Turn_Dir,0,"FWD",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)
3. Add_Instruction_2("FLD",Extend_2,"---",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)

PARK THE ZOOMIE AND END THE TRIP

Drive along the current road to the location of the driveway, STP to make sure traffic is clear before making the PTL/PTR and finally drive to within the correct distance of the barrier.

1. Add_Instruction_2("FLS",Stop_Pos,"---",Pre_Set,0,0,C_Road_No,C_Heading,G_X_Pos,G_Y_Pos)
2. Add_Instruction_2("STP",0,"---",0,0,0,0,C_Heading,G_X_Pos, G_Y_Pos)
3. Add_Instruction_2(Turn_Dir,0,"FWD",0,0,0,0,P_Heading,G_X_Pos, G_Y_Pos)
4. Add_Instruction_2("PID",P_Dist,"---",0,0,0,0,P_Heading,G_X_Pos,G_Y_Pos)

TRIP PLAN COMMANDS / FUNCTION CALLS

These are function calls used to create a Trip Plan motion command for the Zoomie. Each instruction will have 10 parameters, with each parameter having different meanings depending on the command. Not all parameters are used at this time. The Zcheduler creates a list of these commands which can then be sent to the Zoomie for it to execute.

NON-MOTION COMMANDS

Add_Instruction_2("BEG",0,"0",0,0,0,0,0,DW_X, DW_Y)

"BEG": **B**EGinning of Trip Plan
"0" Default string value
DW_X Global X Pos
DW_Y Global Y Pos

Used as an identifier in the Trip Plan that this is the beginning and to establish global position at the start of the trip.

Add_Instruction_2("STP",0,"---",0,0,0,0,0,DW_X, DW_Y)

"STP" **S**T**O**P command, requires permission to move
"---" Default string value
DW_X Global X Pos
DW_Y Global Y Pos

Causes the Zoomie to stop if moving. Before proceeding, the Zoomie must obtain permission to proceed which is based on traffic being clear.

Add_Instruction_2("END",0,"0",0,0,0,0,0,DW_X, DW_Y)

"END": **E**ND of Trip Plan
"0" Default string value
DW_X Global X Pos
DW_Y Global Y Pos

This is the END marker within the Trip Plan. This is used by the Zommie's interpreter to reset everything and clear out the current Trip Plan.

PARKING COMMANDS

Add_Instruction_2("BOD",Dist_2_Travel,"---",0,0,0,0,BU_Heading,X_EP,Y_EP)

| | |
|---------------|---|
| "BOD" | Back Out Driveway command |
| Dist_2_Travel | Distance to move or local position to move to |
| "---" | Default string value |
| BU_Heading | Heading (N,S,E,W)while moving |
| X_EP Global | Global X Pos (at end of move) |
| Y_EP Global | Global Y Pos (at end of move) |

This is used solely for the purpose of backing out of the driveway. This will use both the Encoder feedback and the Distance sensor (CAS) to monitor position during the motion. This motion requires permission to proceed as it will be moving into traffic therefore it is preceded with a STP command.

Add_Instruction_2("PID",P_Dist,"---",0,0,0,0,P_Heading,G_X_Pos,G_Y_Pos)

| | |
|-----------|--|
| "PID" | Pull In Driveway command to go forward specified distance Also uses distance sensor for localization |
| P_Dist | Distance to travel |
| "---" | Default string value |
| P_Heading | Heading while traveling on current road |
| G_X_Pos | Global X Pos (at end of move) |
| G_Y_Pos | Global Y Pos (at end of move) |

This is used solely for the purpose of pulling into the driveway. This will use both the Encoder feedback and the Distance sensor (CAS) to monitor position during the motion. It requires that the Zoomie be positioned at the correct heading, roughly aligned on the guideline and at the approximate distance away from the barrier.

LINE FOLLOWING - STRAIGHT MOTION COMMANDS

Add_Instruction_2("FLS",C_Abs_End,"---",Pre_Set,0,0,C_Road_No,C_Heading,C_XE,C_YE)

| | |
|-----------|--|
| "FLS" | Follow Line S teering command for line following motions |
| C_Abs_End | Local position at end point |
| "---" | Default string value |
| Pre_Set | Presets local position on road when encountering the next magnet on right side |
| C_Road_No | Road number traveling on |
| C_Heading | Heading while traveling on current road |
| C_XE | Global X Pos (at end of move) |
| C_YE | Global Y Pos (at end of move) |

This is the primary line following command and is used primarily on straight roads. This motion may or may not include a Pre_Set command to specify the Local position upon encountering a magnet on the right side. Localization is also updated by sensing marker tape at 1 inch intervals. The road number is provided to the Zoomie so that it can communicate back what road it is traveling on.

Add_Instruction_2("FLD",Extend,"---",0,0,0,0,N_Heading,G_X_Pos,G_Y_Pos)

| | |
|-----------|---|
| "FLS" | Follow Line D istance command for line following motion a specified distance |
| Extend | Distance to travel |
| "---" | Default string value |
| N_Heading | Heading while traveling on current road |
| G_X_Pos | Global X Pos (at end of move) |
| G_Y_Pos | Global Y Pos (at end of move) |

This command is used to drive from one road to another where they may not be connected such as right-angle intersections. Encoder feedback is the only position tracking used in this move.

TURNING COMMANDS

Add_Instruction_2(Piv_Turn_Dir,0,"REV",0,0,0,0,C_Heading,X_EP, Y_EP)

| | |
|--------------|---|
| Piv_Turn_Dir | PTL or PTR, Pivot Turn Left / Right command |
| "REV" | Traverse direction (REV or FWD) |
| C_Heading | Heading (N,S,E,W) at end of turn |
| X_EP Global | Global X Pos (at end of move) |
| Y_EP Global | Global Y Pos (at end of move) |

The Pivot Turn is utilized everywhere except on Bypass road to Bypass road transitions. There are actually two commands, one is LEFT turn the other is Right turn, but otherwise they are identical. The pivot turn can be performed while traversing FORWARD or in REVERSE.

Add_Instruction_2("FLM",Dist,"---",0,0,0,0,N_Heading,N_XS,N_YS)

| | |
|-----------|---|
| "FLM" | Follow Line Motion command for line following motion a specified distance |
| Dist | Distance to travel |
| "---" | Default string value |
| N_Heading | Heading while traveling on current road |
| N_XS | Global X Pos (at end of move) |
| N_YS | Global Y Pos (at end of move) |

This command is used on the Sweeping turns that connect Bypass road sections. The distance specified is approximate as it also uses a marker tape to detect the ending position of the motion. The Marker tape overrides the distance, however the distance must be great enough to get to that marker tape.

Trip Plan as created on the Zscheduler computer.

```
[ 0 ]> ['BEG', 0, '0', 0, 0, 0, 0, 0, 85.5, 31]
[ 1 ]> ['STP', 8.2, '---', 0, 0, 0, 0, 0, 85.5, 31]
[ 2 ]> ['BOD', 8.2, '---', 0, 0, 0, 0, 'E', 94, 31]
[ 3 ]> ['PTL', 0, 'REV', 0, 0, 0, 0, 'N', 94, 31]
[ 4 ]> ['FLS', 16.98, '-w-', 10.0, 0, 0, 21, 'N', 94, 22]
[ 5 ]> ['STP', 0, '-w-', 0, 0, 0, 0, 'N', 94, 22]
[ 6 ]> ['PTL', 0, 'FWD', 0, 0, 0, 0, 'W', 94, 22]
[ 7 ]> ['FLD', 4.5, '-w-', 0, 0, 0, 0, 'W', 87.5, 22]
[ 8 ]> ['FLS', 9, 'A--', 0, 0, 0, 4, 'W', 78.5, 22]
[ 9 ]> ['STP', 0, '---', 0, 0, 0, 0, 'W', 78.5, 22]
[ 10 ]> ['FLD', 1.95, 'A--', 0, 0, 0, 0, 'W', 76, 22]
[ 11 ]> ['PTR', 0, 'FWD', 0, 0, 0, 0, 'N', 76, 22]
[ 12 ]> ['FLS', 23.98, '-w-', 21.0, 0, 0, 19, 'N', 76, 15]
[ 13 ]> ['STP', 0, '-w-', 0, 0, 0, 0, 'N', 76, 15]
[ 14 ]> ['PTL', 0, 'FWD', 0, 0, 0, 0, 'W', 76, 15]
[ 15 ]> ['FLD', 4.5, '-w-', 0, 0, 0, 0, 'W', 69.5, 15]
[ 16 ]> ['FLS', 19.355, '---', 0, 0, 0, 6, 'W', 49.625, 15]
[ 17 ]> ['PTR', 0, 'FWD', 0, 0, 0, 0, 'N', 49.625, 15]
[ 18 ]> ['PID', 4.5, '---', 0, 0, 0, 0, 'N', 49.625, 10.25]
[ 19 ]> ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0]
```

This is the Trip Plan being transmitted to the Zoomie

Transmitting 0 ['BEG', 0, '0', 0, 0, 0, 0, 0, 85.5, 31] Z%/TP_SEG
Transmitting 1 ['STP', 8.2, '---', 0, 0, 0, 0, 0, 85.5, 31] Z%/TP_SEG
Transmitting 2 ['BOD', 8.2, '---', 0, 0, 0, 0, 'E', 94, 31] Z%/TP_SEG
Transmitting 3 ['PTL', 0, 'REV', 0, 0, 0, 0, 'N', 94, 31] Z%/TP_SEG
Transmitting 4 ['FLS', 16.98, '-w-', 10.0, 0, 0, 21, 'N', 94, 22] Z%/TP_SEG
Transmitting 5 ['STP', 0, '-w-', 0, 0, 0, 0, 'N', 94, 22] Z%/TP_SEG
Transmitting 6 ['PTL', 0, 'FWD', 0, 0, 0, 0, 'W', 94, 22] Z%/TP_SEG
Transmitting 7 ['FLD', 4.5, '-w-', 0, 0, 0, 0, 'W', 87.5, 22] Z%/TP_SEG
Transmitting 8 ['FLS', 9, 'A--', 0, 0, 0, 4, 'W', 78.5, 22] Z%/TP_SEG
Transmitting 9 ['STP', 0, '---', 0, 0, 0, 0, 'W', 78.5, 22] Z%/TP_SEG
Transmitting 10 ['FLD', 1.95, 'A--', 0, 0, 0, 0, 'W', 76, 22] Z%/TP_SEG
Transmitting 11 ['PTR', 0, 'FWD', 0, 0, 0, 0, 'N', 76, 22] Z%/TP_SEG
Transmitting 12 ['FLS', 23.98, '-w-', 21.0, 0, 0, 19, 'N', 76, 15] Z%/TP_SEG
Transmitting 13 ['STP', 0, '-w-', 0, 0, 0, 0, 'N', 76, 15] Z%/TP_SEG
Transmitting 14 ['PTL', 0, 'FWD', 0, 0, 0, 0, 'W', 76, 15] Z%/TP_SEG
Transmitting 15 ['FLD', 4.5, '-w-', 0, 0, 0, 0, 'W', 69.5, 15] Z%/TP_SEG
Transmitting 16 ['FLS', 19.355, '---', 0, 0, 0, 6, 'W', 49.625, 15] Z%/TP_SEG
Transmitting 17 ['PTR', 0, 'FWD', 0, 0, 0, 0, 'N', 49.625, 15] Z%/TP_SEG
Transmitting 18 ['PID', 4.5, '---', 0, 0, 0, 0, 'N', 49.625, 10.25] Z%/TP_SEG
Transmitting 19 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG
Transmitting 20 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG
Transmitting 21 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG
Transmitting 22 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG
Transmitting 23 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG
Transmitting 24 ['END', 0, '---', 0, 0, 0, 0, 0, 0, 0] Z%/TP_SEG