

Mobility Management Discussion:

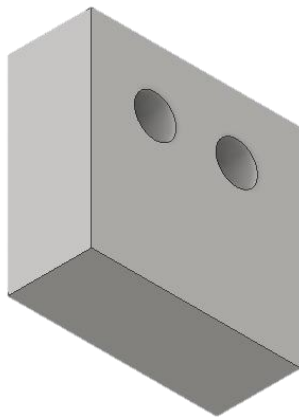
1-Motor Selection and Implementation

- DC Geared Motor for Movement:

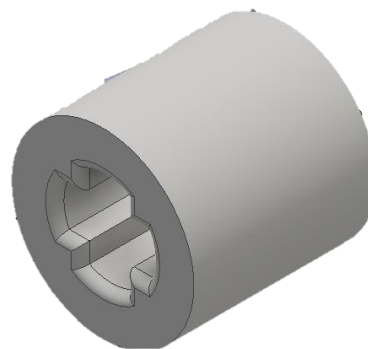
- Selection: The DC geared motor is selected for its high speed and efficiency, ideal for moving the vehicle forward and backward. Its built-in gearbox allows for effective speed regulation.
- Implementation: The motor's built-in gearbox connects directly to a differential mechanism, facilitating smooth power distribution to the wheels and enabling independent wheel speeds during turns. To help fix the motor in place and ensure proper alignment with the differential mechanism, we printed a spacer using 3D printing technology. Additionally, we printed a cabler that acts as a connector between the motor and the differential mechanism shaft, ensuring the motion of gears when the motor rotates.



3D Printed Models :

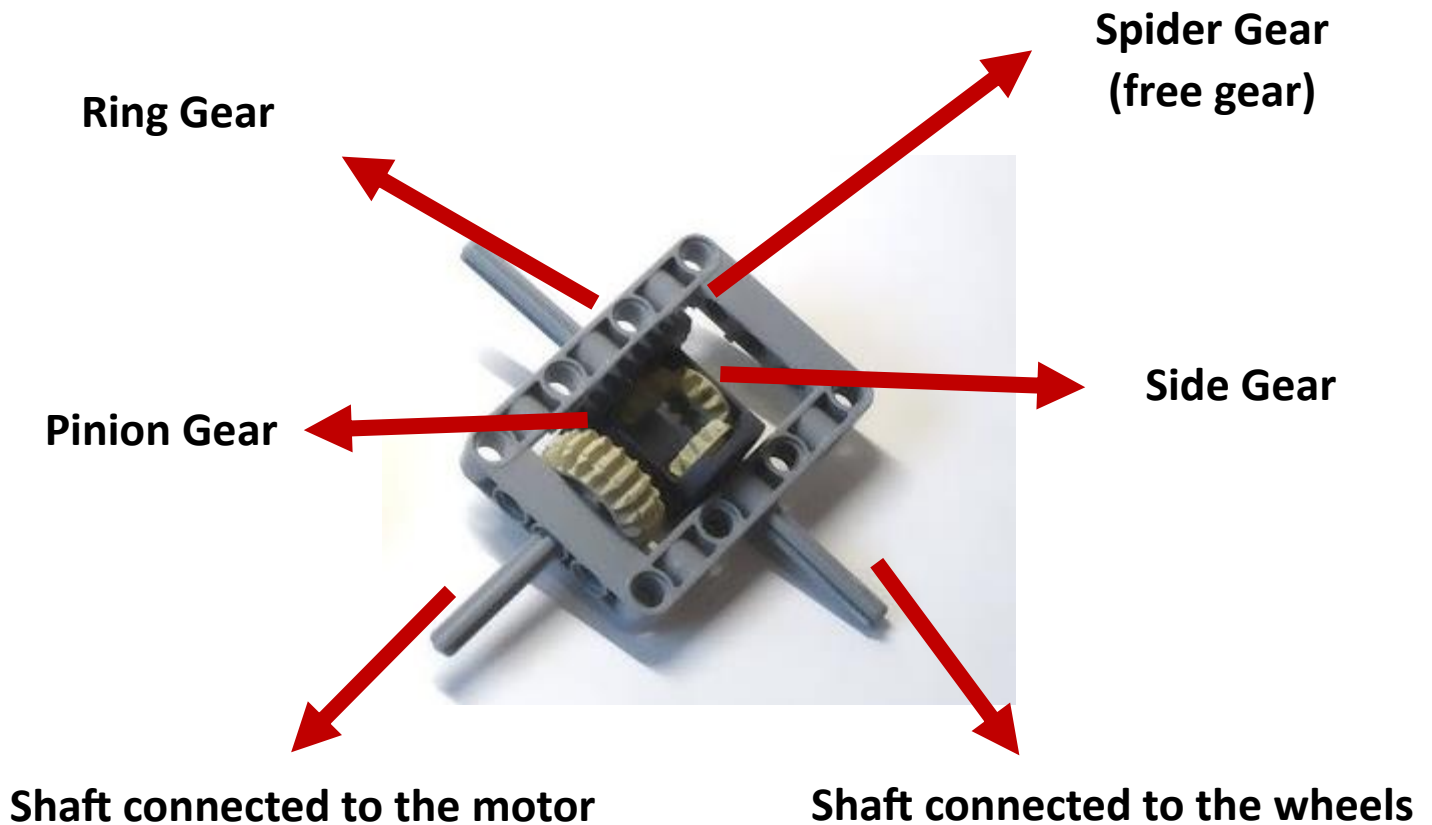


Spacer



Cabler

Differential mechanism (Built from LEGO EV3):



- The differential mechanism enables the wheels to rotate at different speeds, providing smooth turns and enhancing overall stability.

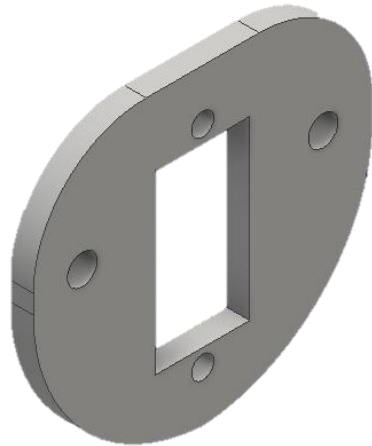
- Servo Motor for Steering:

- o Selection: A 180° servo motor is utilized for steering, providing precise control necessary for effective navigation.
- o Implementation: The servo is mounted at the front of the vehicle and connected to the steering mechanism, which is constructed using LEGO EV3 beams and shafts. To securely hold the servo in the chassis, we used a 3D printed model, ensuring stability during operation. Additionally, we designed a 3D printed connector that links the servo to the LEGO beams, allowing for effective rotation of the wheels when the servo rotates. This

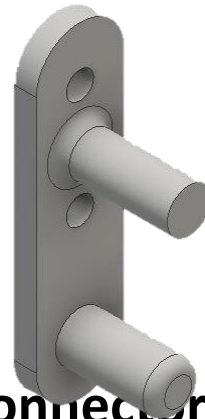


configuration enhances the vehicle's maneuverability and responsiveness.

3D Printed Models :

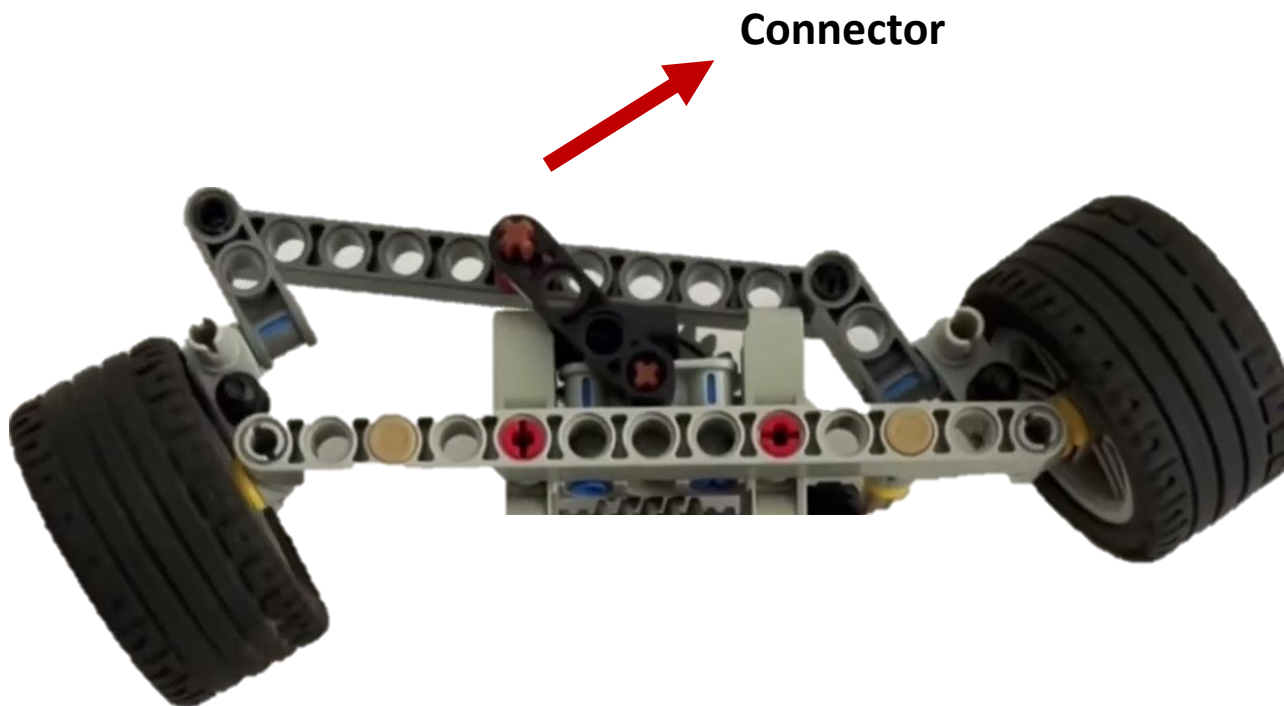


Servo holder



Connector

Steering mechanism:



Vehicle Chassis Design and Selection

- For our vehicle, we opted for a handmade wooden chassis due to its low weight, which contributes to improved maneuverability and efficiency. The wooden structure provides a sturdy base while remaining lightweight, allowing for easier handling and faster speeds.
- To build the movement mechanism, we utilized LEGO components. This modular approach not only simplifies assembly but also allows for easy modifications and repairs as needed.

Motors Mounting:

- o The yellow DC geared motor is securely mounted at the rear of the chassis. This position is chosen for ease of access, making it straightforward to replace or maintain the motor if needed.
- o The built-in gearbox of the motor connects directly to the differential, which then distributes power to the wheel axles, allowing for independent movement during turns.
- o The servo motor is mounted at the front and directly linked to the steering assembly for precise wheel control.

Used Wheels (4x):

- We selected tank LEGO wheels for the vehicle, as they offer high friction and a larger diameter (8 CM). The increased friction enhances traction, which is essential for navigating various surfaces. Additionally, the larger diameter of the wheels contributes to a higher speed, enabling the vehicle to traverse obstacles more effectively while maintaining stability.



In the qualifying challenge, our strategy involves the following steps to ensure effective navigation:

1. Forward Movement:

- **The vehicle will continuously move forward until the front ultrasonic sensor detects a wall.**

2. Distance Comparison:

- **Upon detecting an obstacle, the vehicle will compare the distances measured by the right and left ultrasonic sensors to determine which side has more space available.**

3. Turning Decision:

- **The vehicle will turn towards the side with the larger distance, allowing it to navigate around the obstacle efficiently.**

4. Centering algorithm:

- **While moving, the robot will utilize four ultrasonic sensors (two on the right and two on the left) to help maintain its central position on the course. This ensures balanced navigation and helps avoid straying from the intended path.**

In the final challenge, our strategy involves the following steps to ensure effective navigation:

1- Forward Movement:

- **The vehicle will continuously move forward until the color sensor detects either a red or green obstacle.**

2- Color Detection:

- **Upon detecting a color, the vehicle will determine its position relative to the obstacle.**

3- Turning Decision:

- Based on the position of the detected color (left or right), the vehicle will make a decision to turn right or left to avoid the obstacle.

4- Continuing Forward:

- After successfully navigating around the obstacle, the vehicle will continue moving forward while monitoring for additional obstacles.

Pseudo Code

```
1. DEFINE pin constants for ultrasonic sensors and motors
2
3. INITIALIZE global variables for distances and speeds
4
5 FUNCTION setup:
6   SET pin modes for ultrasonic sensors and motors
7   ATTACH servo to the designated pin
8   INITIALIZE serial communication
9   WAIT until button is pressed
10
11 FUNCTION loop:
12   CALL ultra to get distances from sensors
13
14   IF right rasper sensor is triggered:
15     RECORD start time
16     WHILE right rasper sensor is still triggered:
17       CALL Move to move the robot left
18     RECORD end time
19     CALL Move to stop and adjust speed
20     CALL Move to continue in the original direction
21
22   ELSE IF left rasper sensor is triggered:
23     RECORD start time
24     WHILE left rasper sensor is still triggered:
25       CALL Move to move the robot right
26     RECORD end time
27     CALL Move to stop and adjust speed
28     CALL Move to continue in the original direction
29
30   ELSE IF center front distance is less than threshold:
31     CALL Move to stop the robot
32     CALL ultra to update distances
33     IF right front distance is greater than left front distance:
34       CALL Move to turn left
35
36   ELSE:
37     SET speed to a default value
38     CALL ultra to update distances
39     IF right front distance is less than threshold:
40       CALL Move to steer left
41     ELSE IF left front distance is less than threshold:
42       CALL Move to steer right
43     ELSE:
44       CALL Move to continue straight
45
46 FUNCTION ultra(front, back):
47   IF front sensors are active:
48     FOR each front sensor:
49       TRIGGER sensor
50       WAIT for echo response
51       CALCULATE distance using response time
52
53   IF back sensors are active:
54     FOR each back sensor:
55       TRIGGER sensor
56       WAIT for echo response
57       CALCULATE distance using response time
58
59   REPLACE zero distances with a large value
60
61 FUNCTION Move(speed, direction, duration):
62   IF speed is positive:
63     SET motor direction to forward
64     SET motor speed based on speed
65   ELSE IF speed is negative:
66     SET motor direction to backward
67     SET motor speed based on speed
68   ELSE:
69     STOP motors
70
71   CALCULATE steering based on direction
72   MOVE servo to the calculated position
73   WAIT for the specified duration
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
```

```
FUNCTION center(duration):
  RECORD start time
  WHILE the elapsed time is less than duration:
    CALL ultra to update distances
    IF right front distance is less than threshold:
      CALL Move to steer left
    ELSE IF left front distance is less than threshold:
      CALL Move to steer right
    ELSE:
      CALL Move to go straight
```