



# LEGO BB-8

Learn how to automate a LEGO BB-8 for motion, light, and sound using Crazy Circuits.

Written By: Joshua



## INTRODUCTION

We absolutely LOVE the new LEGO Star Wars sets that have come out over the last couple of years. They're well designed, fun to build, and look great. What would make them even *more* fun is if they also moved on their own!

We took an off the shelf LEGO BB-8 set and automated it so the head spins around! Even better, we added sound effects and lighting effects! But if that isn't enough we also added a motion sensor so that it would activate when someone walks by.

Overall this project isn't overly difficult to do, but does take awhile and requires a healthy amount of random Technic LEGO parts to built the gear box area.



### TOOLS:

- [Scissors](#) (1)



### PARTS:

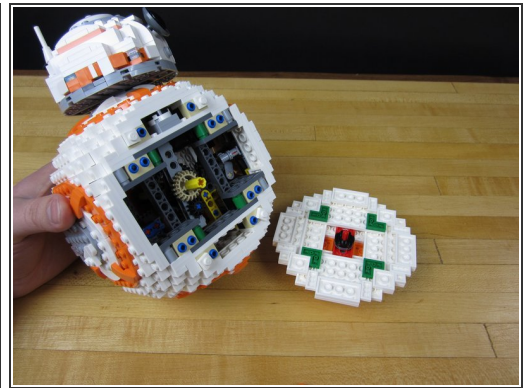
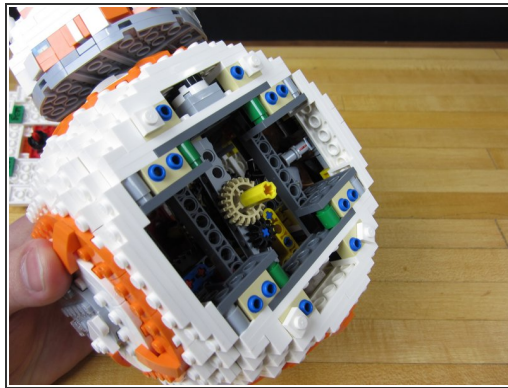
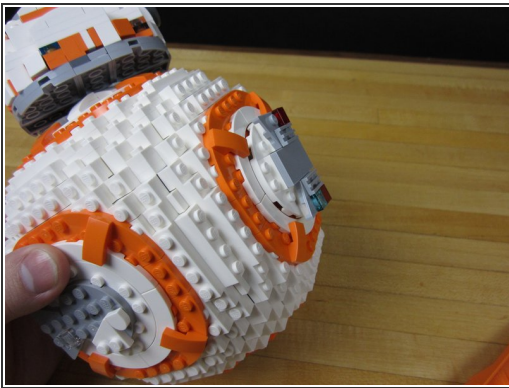
- [Crazy Circuits Robotics Board](#) (1)
- [Crazy Circuits LED Chip](#) (1)
- [Continuous Rotation Servo](#) (1)
- [YX5300 MP3 Player Module](#) (1)
- [Ultrasonic Distance Sensor](#) (1)
- [Female to Female Jumper Cables](#) (1)
- [LEGO Technic Worm Gear Box](#) (1)
- [LEGO Technic Large Gear](#) (1)
- [LEGO BB-8 Kit](#) (1)

## Step 1 — Build Your BB-8



- Build your LEGO BB-8 as normal.

## Step 2 — Remove the Head Control

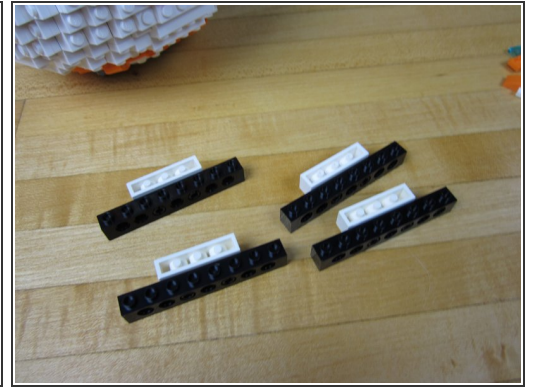
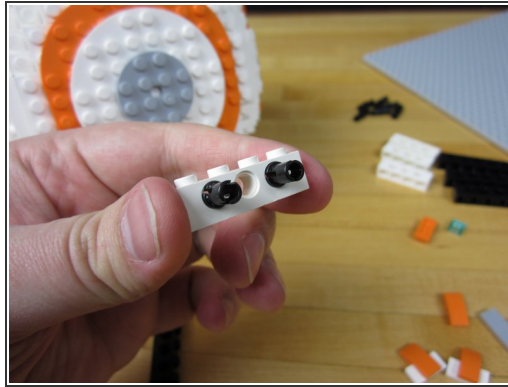
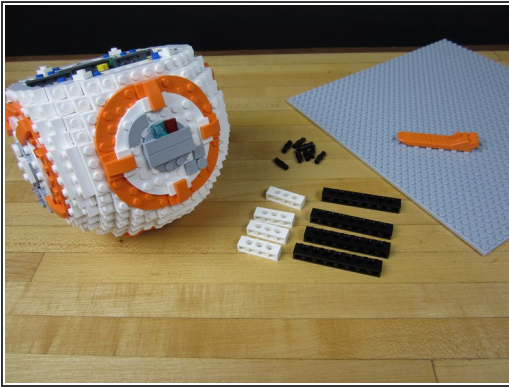


- Using a pry tool, remove the side of BB-8 that controls the head.

⚠ Put inside half of the axel back in if it comes out.

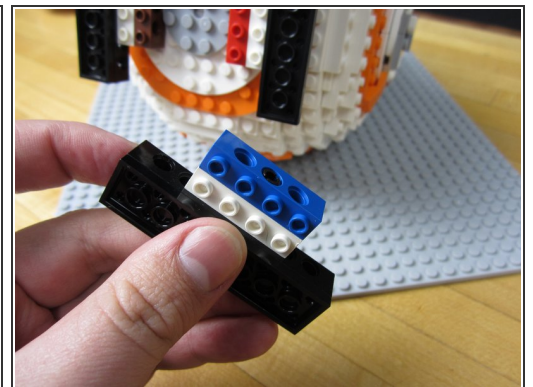
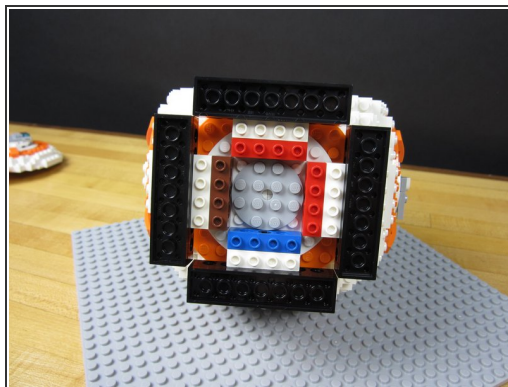
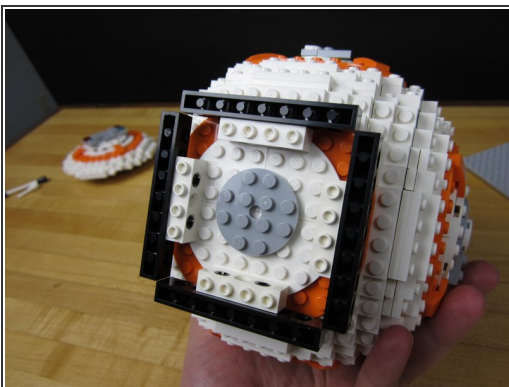


### Step 3 — Construct Mount Adaptor



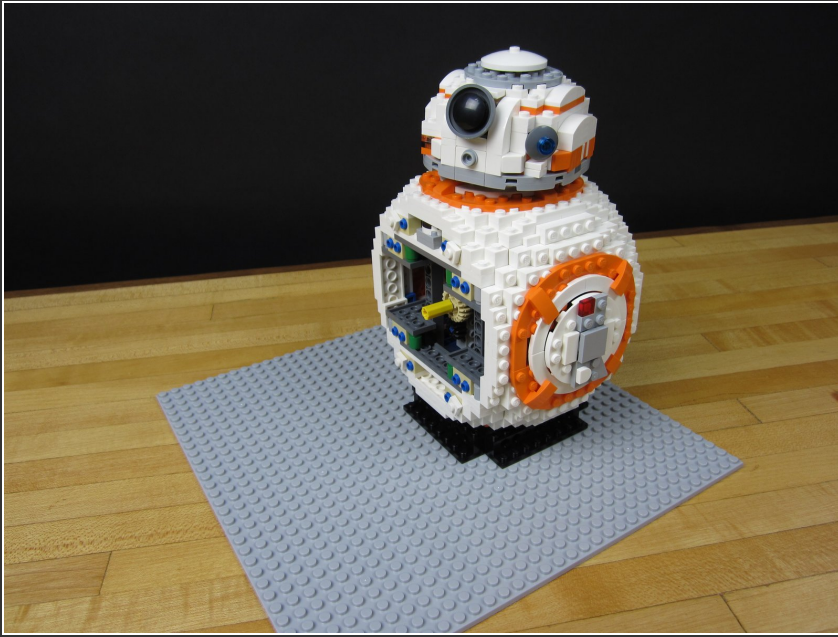
- ❗ Using 1x4 and 1x8 Technic bricks, construct an adaptor to mount the BB-8 onto your large Base Plate.
- Remove all the extra parts on the bottom of your BB-8. You want to leave the bottom white area completely flat.
- Connect your white bricks UPSIDE DOWN to your black technic bricks.
- ❗ We ended up using two of the 1x4 bricks per adaptor for extra strength, but it's probably not necessary.

### Step 4 — Connect Adaptors and Plates



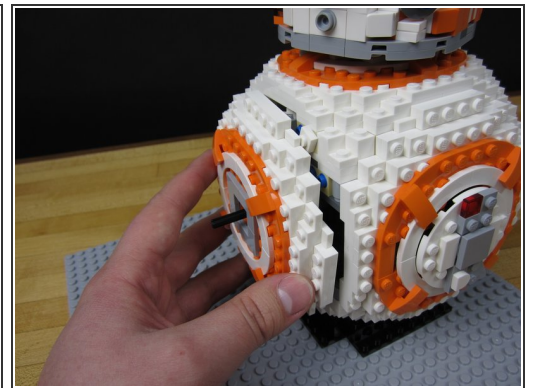
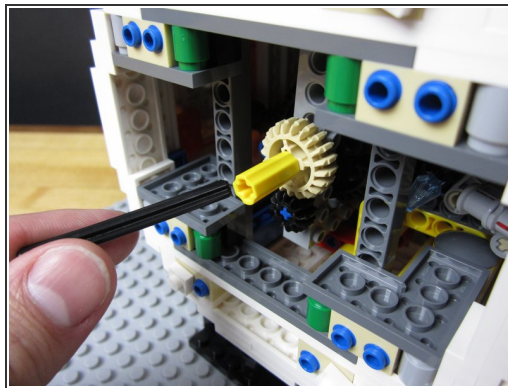
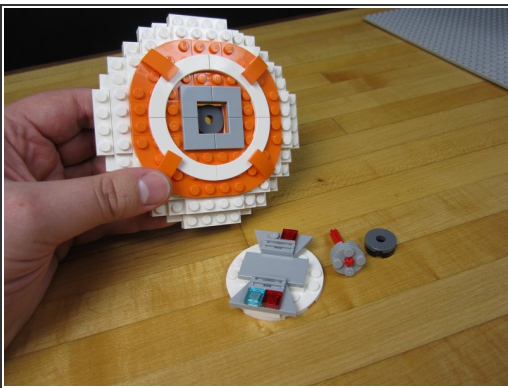
- Attach your adapters to your BB-8.
- If possible, use some 2x8 plates to increase the footprint and connection area.
- As mentioned before, we also added in a second 1x4 brick to be on the safe side.

## Step 5 — Attach to Base Plate



- We connected everything to a large base plate.
- Give yourself some space in the front and back in order to add the rest of your parts.
- Make sure the empty "gear area" is pointed towards the BACK of your project. We need space for the gearing and electronics.

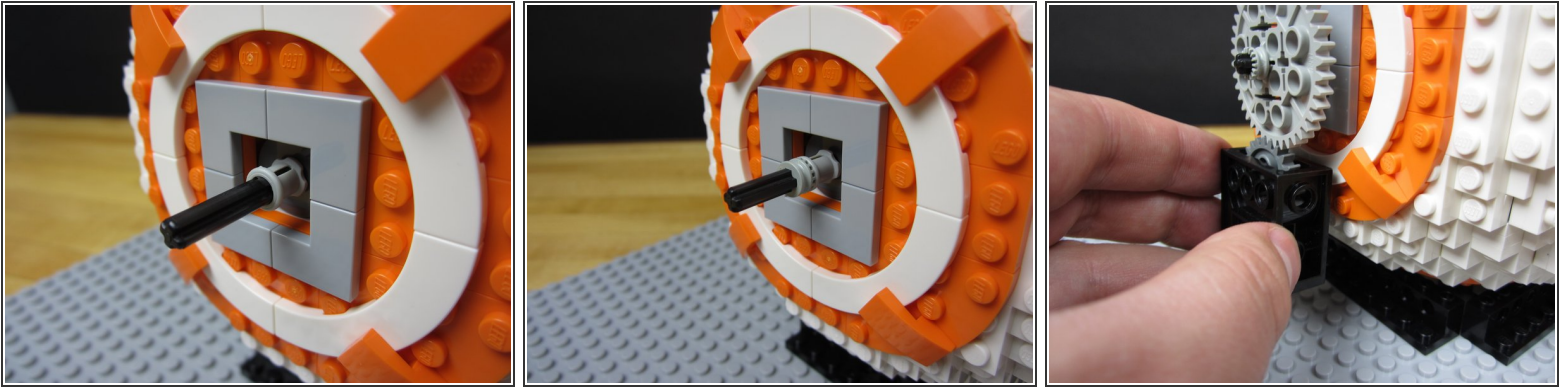
## Step 6 — Remove Spinner and Reattach



- Remove the spinning section from your head control area.
- Grab a long size 12 or better axle and attach it to the connector inside the BB-8.
- Reattach the entire side.

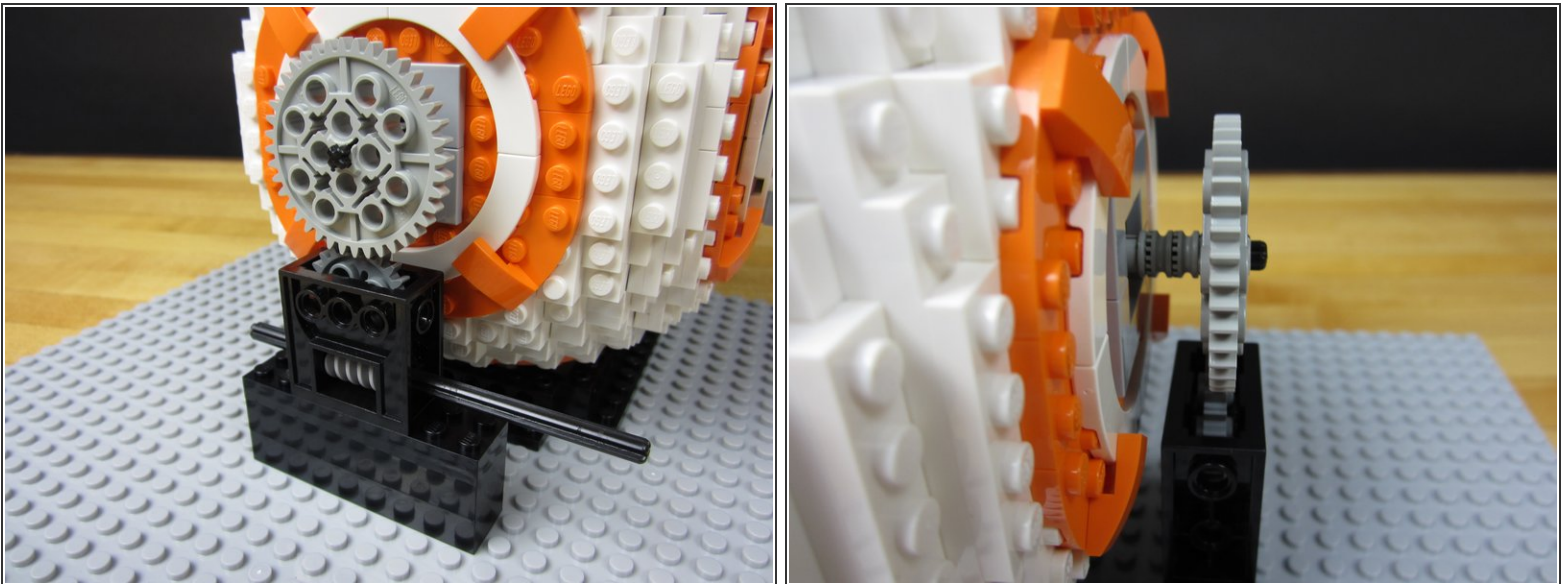


## Step 7 — Add Spacers and a Large Gear



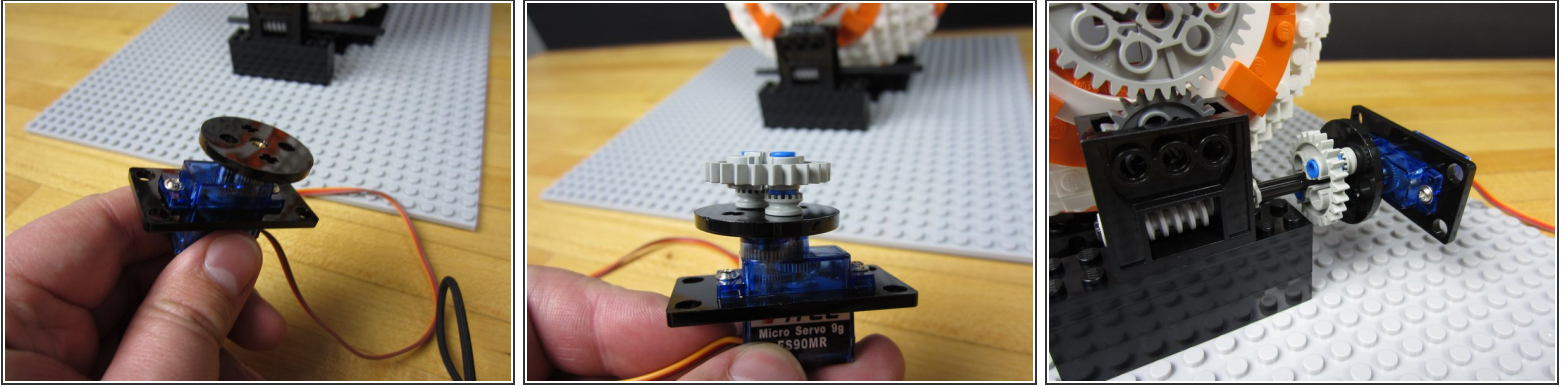
- You'll need to attach a couple of different sized spacers before attaching your large Technic Gear.
- We also added a small bushing to the end of our axle to hold everything tightly in place.

## Step 8 — Create Worm Drive Stand



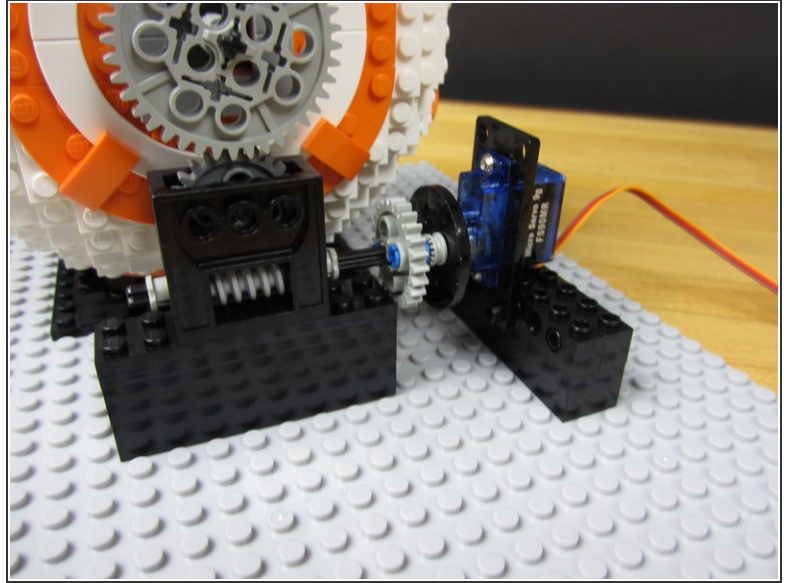
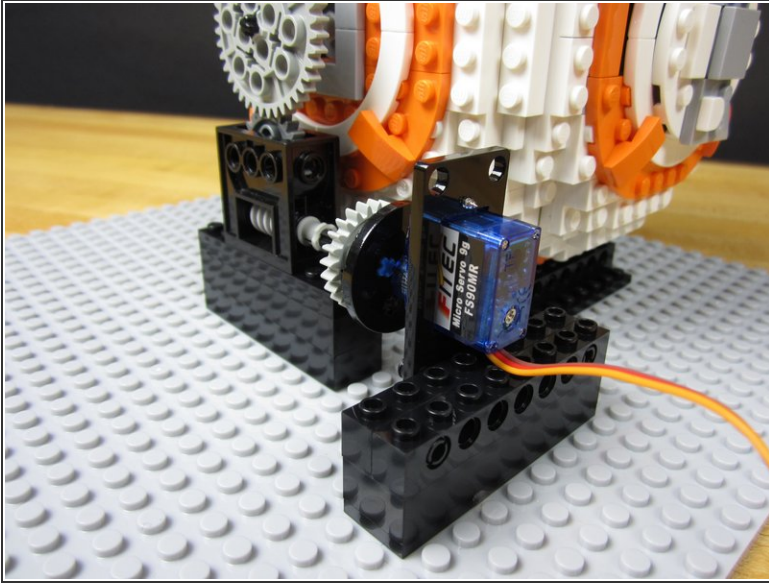
- Use a couple of standard LEGO 2x8 sized bricks along with 2 2x8 sized plates to create a platform for your worm drive.
- Connect everything below your large technic Gear.
- Adjust parts as needed to make sure everything connects and fits well. Give the axle a spin to make sure they work.

## Step 9 — Prepare the Servo & Horn



- i** We're using a 9G sized Continuous Rotation Servo with metal gearing, connected to LEGO using our Crazy Circuits adaptors. (Laser cut and open source!)
- Attach the laser cut Crazy Circuits adaptors to the 9G sized servo.
  - Connect the round Servo Horn (laser cut disc) to the Servo. Using a couple of technic pieces and a gear, build out an adaptor on the end.
  - This allows your servo to directly interface with the axle on the worm drive.

## Step 10 — Connect the Gearbox to the X-Wing

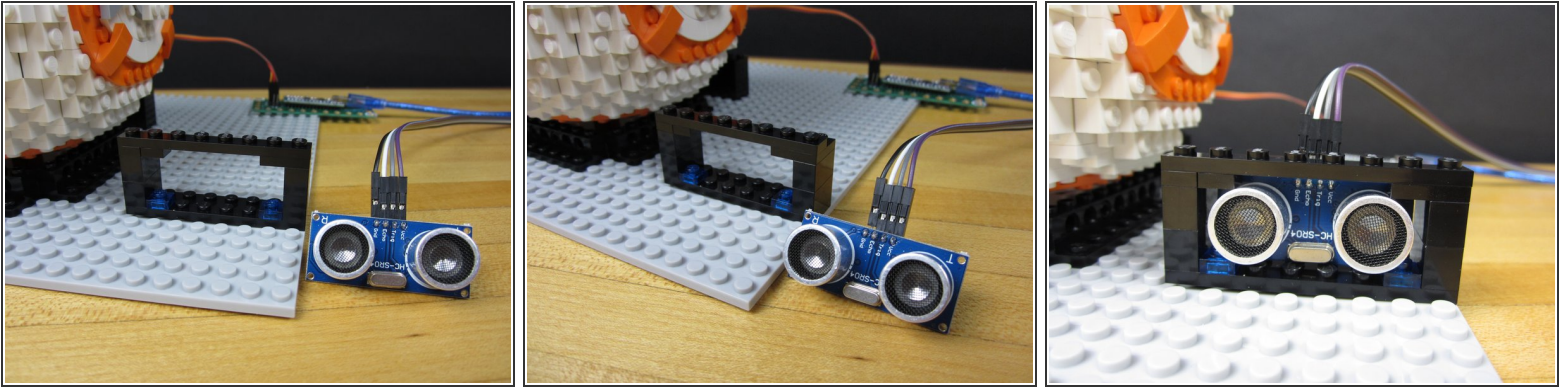


- We connected two 1x8 Technic bricks together and then attached the servo to those pieces.
- Below them is a single 2x8 brick.

**⚠ The main concern here is to keep everything secured and tightly together. The servo moves around a lot and the last thing you want is for your pieces to start popping off the large base plate.**

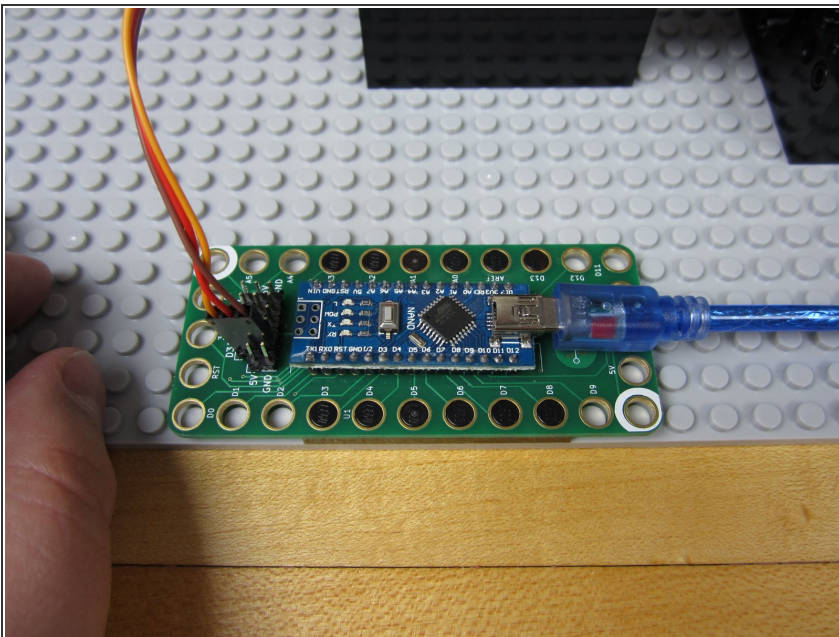


## Step 11 — Build a Sensor Mount



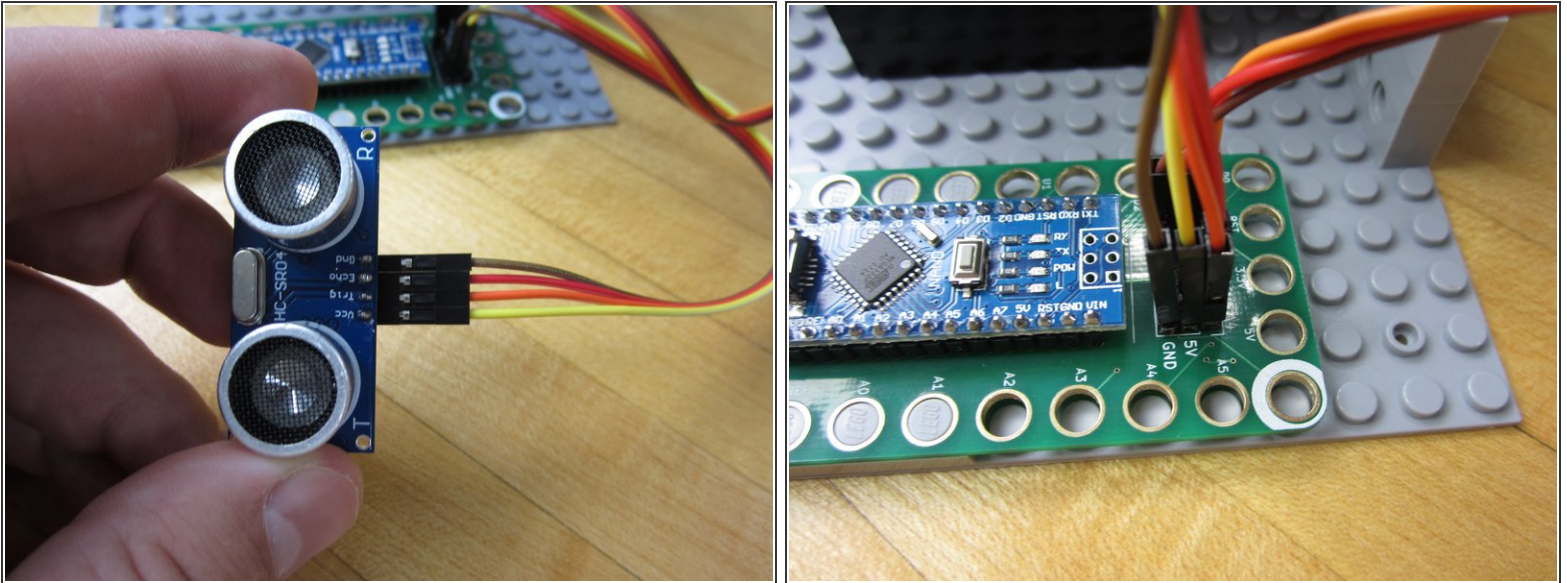
- i** We went all fancy and included an Ultrasonic Motion Sensor. To make it look nice we build a small LEGO based holder for it to hang out in. We can't take credit for this design, we randomly found it in a google image search.
- Start with a 2x8 plate, attach some 1x2 bricks onto the side, two L shaped plates on the top edge, and a 1x8 across the top. Use a couple of 1x1 plated on the bottom (blue in our design).

## Step 12 — Build the Robotics Board Platform



- Use two 1x6 or 1x8 plates to create a small platform for the Robotics Board to sit on.
- You can also plug the Servo into the D3 Row Header Set at this time.

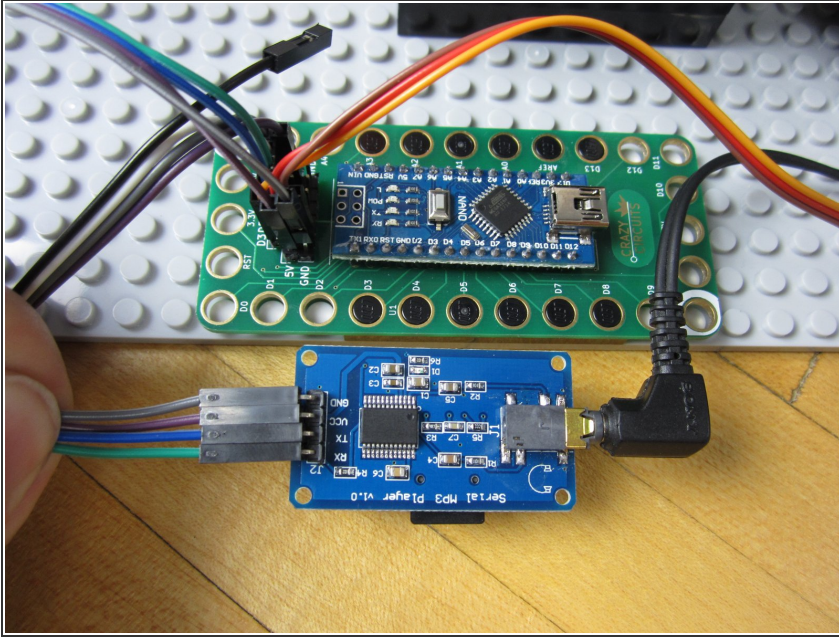
## Step 13 — Connect the Range Finder



**i** The Diagram at the start of the guide is a good reference to show you how to wire up the Range Finder.

- Connect VCC to a 5V Pin.
- Connect Trig to A4.
- Connect Echo to A5.
- Connect GND to a GND Pin.

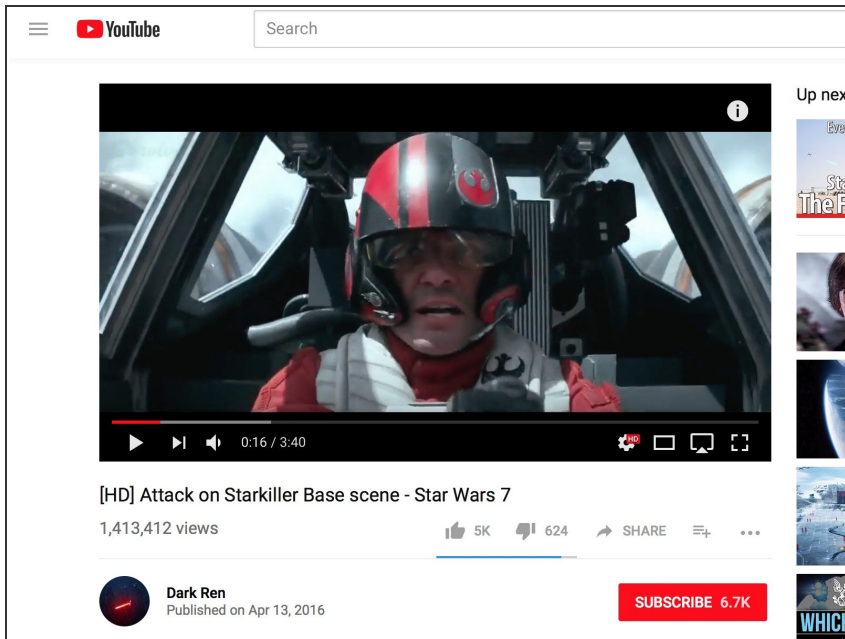
## Step 14 — Connect the MP3 Player



- ① Use the diagram again to help you out.
- Connect GND to a GND Pin.
- Connect VCC to a 5V Pin.
- Connect TX to 5.
- Connect RX to 6.
- ① Online documentation for this board is weird. Trust us, this is the correct wiring.



## Step 15 — Find a Sound Clip





- We grabbed our sound clip via a YouTube Video. You can use any sound file as long as it's a .WAV or .MP3. (We can't give you a sound clip for Copyright reasons.)
  - You'll want to use just one sound clip for our code. Once you find one put it on a micro SD card formatted in FAT.
  - Put the Micro SD Card into the MP3 Player.
- ⓘ Make sure you take note as to how long your clip is, as it will be quite helpful when modifying the code.
- ⓘ BB-8 sounds are great as well as some classic Star Wars music.

## Step 16 — Modify Your Code

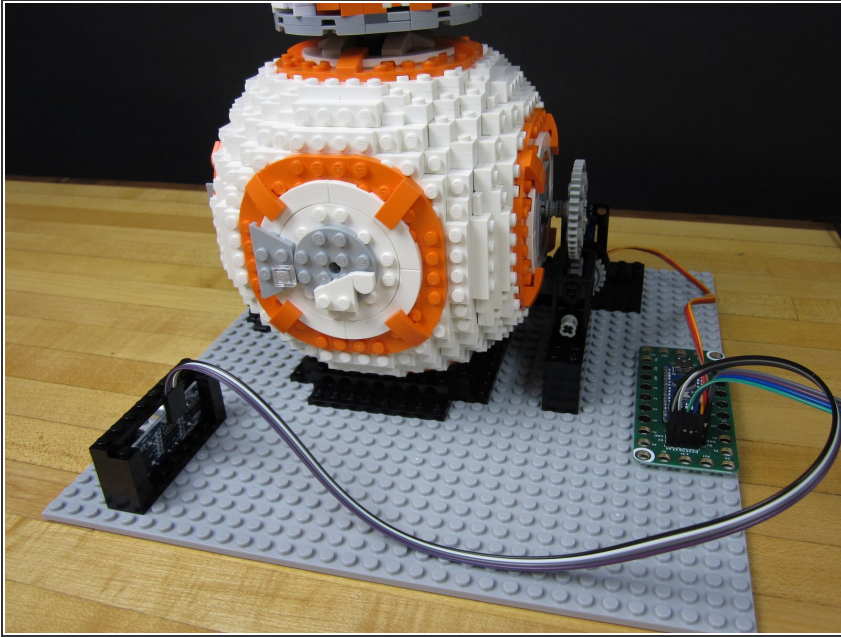
```

XWingCode
19 NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE); // NewPing setup of pins and maximum distance.
20
21 int theDistance = 0;
22 int theDistanceTemp = 0;
23 int theDistanceTotal = 0;
24 int triggerDistance = 30; // set the distance in centimeters to trigger the action
25 int sampleVal = 5;
26
27 Servo myServo;
28
29 int servoPIN = 3;
30 int servoTurnTimeForward = 20000; // time servo will run this many milliseconds
31 int servoTurnTimeReverse = 20000; // time servo will run this many milliseconds
32 int servoForward = 1200;
33 int servoStop = 1500;
34 int servoReverse = 1800;
35
36 int LEDpin = 11;
37
38 static int8_t Send_buf[8] = {0};
39
40 #define CMD_PLAY_N_INDEX 0x03
41 #define CMD_SET_VOLUME 0x06
42 #define CMD_SEL_DEV 0x09
43 #define DEV_TF 0x02
44 #define CMD_PLAY 0x00
45 #define CMD_PAUSE 0x0E
46 #define CMD_SINGLE_CYCLE 0x19
47 #define SINGLE_CYCLE_ON 0x00
48 #define SINGLE_CYCLE_OFF 0x01
49 #define CMD_PLAY_N_VOL 0x22
50
51 void setup() {
52   Serial.begin(9600);
53   Serial.println("Starting...");
54   mySerial.begin(9600);
55   pinMode(LEDpin, OUTPUT);
56   analogWrite(LEDpin, 10);
57   myServo.attach(servoPIN);
58
59   // reset
60   theDistance = 0;
61   theDistanceTotal = 0;
62
63   // LED
64   Serial.println("LED up");
65   for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
66     analogWrite(LEDpin, fadeValue);
67     delay(50);
68   }
69
70   // reset
71   theDistance = 0;
72   theDistanceTotal = 0;
73
74   // LED
75   Serial.println("LED up");
76   for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
77     analogWrite(LEDpin, fadeValue);
78     delay(50);
79   }
80
81   // audio
82   Serial.println("play audio");
83   sendCommand(CMD_PLAY_N_VOL, 0x1E01); // play the first file with volume 15 class
84   delay(25000); // delay for audio to finish playing
85
86   // servo
87   Serial.println("servo reverse");
88   myServo.writeMicroseconds(servoReverse); // start turning
89   delay(servoTurnTimeReverse); // allow servo to turn for X number of milliseconds
90   Serial.println("servo stop");
91   myServo.writeMicroseconds(servoStop); // stop turning
92
93   // LED
94   Serial.println("LED up");
95   for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
96     analogWrite(LEDpin, fadeValue);
97     delay(50);
98   }
99
100  // reset
101  theDistance = 0;
102  theDistanceTotal = 0;
103
104  // LED
105  Serial.println("LED up");
106  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
107    analogWrite(LEDpin, fadeValue);
108    delay(50);
109  }
110
111  // else {
112  //   Serial.println("Waiting...");
113  // }
114
115  // LED
116  Serial.println("LED up");
117  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
118    analogWrite(LEDpin, fadeValue);
119    delay(50);
120  }
121
122  // reset
123  theDistance = 0;
124  theDistanceTotal = 0;
125
126  // LED
127  Serial.println("LED up");
128  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
129    analogWrite(LEDpin, fadeValue);
130    delay(50);
131  }
132
133  // reset
134  theDistance = 0;
135  theDistanceTotal = 0;
136
137  // LED
138  Serial.println("LED up");
139  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
140    analogWrite(LEDpin, fadeValue);
141    delay(50);
142  }
143
144  // reset
145  theDistance = 0;
146  theDistanceTotal = 0;
147
148  // LED
149  Serial.println("LED up");
150  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
151    analogWrite(LEDpin, fadeValue);
152    delay(50);
153  }
154
155  // reset
156  theDistance = 0;
157  theDistanceTotal = 0;
158
159  // LED
160  Serial.println("LED up");
161  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
162    analogWrite(LEDpin, fadeValue);
163    delay(50);
164  }
165
166  // reset
167  theDistance = 0;
168  theDistanceTotal = 0;
169
170  // LED
171  Serial.println("LED up");
172  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
173    analogWrite(LEDpin, fadeValue);
174    delay(50);
175  }
176
177  // reset
178  theDistance = 0;
179  theDistanceTotal = 0;
180
181  // LED
182  Serial.println("LED up");
183  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
184    analogWrite(LEDpin, fadeValue);
185    delay(50);
186  }
187
188  // reset
189  theDistance = 0;
190  theDistanceTotal = 0;
191
192  // LED
193  Serial.println("LED up");
194  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
195    analogWrite(LEDpin, fadeValue);
196    delay(50);
197  }
198
199  // reset
200  theDistance = 0;
201  theDistanceTotal = 0;
202
203  // LED
204  Serial.println("LED up");
205  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
206    analogWrite(LEDpin, fadeValue);
207    delay(50);
208  }
209
210  // reset
211  theDistance = 0;
212  theDistanceTotal = 0;
213
214  // LED
215  Serial.println("LED up");
216  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
217    analogWrite(LEDpin, fadeValue);
218    delay(50);
219  }
220
221  // reset
222  theDistance = 0;
223  theDistanceTotal = 0;
224
225  // LED
226  Serial.println("LED up");
227  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
228    analogWrite(LEDpin, fadeValue);
229    delay(50);
230  }
231
232  // reset
233  theDistance = 0;
234  theDistanceTotal = 0;
235
236  // LED
237  Serial.println("LED up");
238  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
239    analogWrite(LEDpin, fadeValue);
240    delay(50);
241  }
242
243  // reset
244  theDistance = 0;
245  theDistanceTotal = 0;
246
247  // LED
248  Serial.println("LED up");
249  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
250    analogWrite(LEDpin, fadeValue);
251    delay(50);
252  }
253
254  // reset
255  theDistance = 0;
256  theDistanceTotal = 0;
257
258  // LED
259  Serial.println("LED up");
260  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
261    analogWrite(LEDpin, fadeValue);
262    delay(50);
263  }
264
265  // reset
266  theDistance = 0;
267  theDistanceTotal = 0;
268
269  // LED
270  Serial.println("LED up");
271  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
272    analogWrite(LEDpin, fadeValue);
273    delay(50);
274  }
275
276  // reset
277  theDistance = 0;
278  theDistanceTotal = 0;
279
280  // LED
281  Serial.println("LED up");
282  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
283    analogWrite(LEDpin, fadeValue);
284    delay(50);
285  }
286
287  // reset
288  theDistance = 0;
289  theDistanceTotal = 0;
290
291  // LED
292  Serial.println("LED up");
293  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
294    analogWrite(LEDpin, fadeValue);
295    delay(50);
296  }
297
298  // reset
299  theDistance = 0;
300  theDistanceTotal = 0;
301
302  // LED
303  Serial.println("LED up");
304  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
305    analogWrite(LEDpin, fadeValue);
306    delay(50);
307  }
308
309  // reset
310  theDistance = 0;
311  theDistanceTotal = 0;
312
313  // LED
314  Serial.println("LED up");
315  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
316    analogWrite(LEDpin, fadeValue);
317    delay(50);
318  }
319
320  // reset
321  theDistance = 0;
322  theDistanceTotal = 0;
323
324  // LED
325  Serial.println("LED up");
326  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
327    analogWrite(LEDpin, fadeValue);
328    delay(50);
329  }
330
331  // reset
332  theDistance = 0;
333  theDistanceTotal = 0;
334
335  // LED
336  Serial.println("LED up");
337  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
338    analogWrite(LEDpin, fadeValue);
339    delay(50);
340  }
341
342  // reset
343  theDistance = 0;
344  theDistanceTotal = 0;
345
346  // LED
347  Serial.println("LED up");
348  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
349    analogWrite(LEDpin, fadeValue);
350    delay(50);
351  }
352
353  // reset
354  theDistance = 0;
355  theDistanceTotal = 0;
356
357  // LED
358  Serial.println("LED up");
359  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
360    analogWrite(LEDpin, fadeValue);
361    delay(50);
362  }
363
364  // reset
365  theDistance = 0;
366  theDistanceTotal = 0;
367
368  // LED
369  Serial.println("LED up");
370  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
371    analogWrite(LEDpin, fadeValue);
372    delay(50);
373  }
374
375  // reset
376  theDistance = 0;
377  theDistanceTotal = 0;
378
379  // LED
380  Serial.println("LED up");
381  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
382    analogWrite(LEDpin, fadeValue);
383    delay(50);
384  }
385
386  // reset
387  theDistance = 0;
388  theDistanceTotal = 0;
389
390  // LED
391  Serial.println("LED up");
392  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
393    analogWrite(LEDpin, fadeValue);
394    delay(50);
395  }
396
397  // reset
398  theDistance = 0;
399  theDistanceTotal = 0;
400
401  // LED
402  Serial.println("LED up");
403  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
404    analogWrite(LEDpin, fadeValue);
405    delay(50);
406  }
407
408  // reset
409  theDistance = 0;
410  theDistanceTotal = 0;
411
412  // LED
413  Serial.println("LED up");
414  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
415    analogWrite(LEDpin, fadeValue);
416    delay(50);
417  }
418
419  // reset
420  theDistance = 0;
421  theDistanceTotal = 0;
422
423  // LED
424  Serial.println("LED up");
425  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
426    analogWrite(LEDpin, fadeValue);
427    delay(50);
428  }
429
430  // reset
431  theDistance = 0;
432  theDistanceTotal = 0;
433
434  // LED
435  Serial.println("LED up");
436  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
437    analogWrite(LEDpin, fadeValue);
438    delay(50);
439  }
440
441  // reset
442  theDistance = 0;
443  theDistanceTotal = 0;
444
445  // LED
446  Serial.println("LED up");
447  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
448    analogWrite(LEDpin, fadeValue);
449    delay(50);
450  }
451
452  // reset
453  theDistance = 0;
454  theDistanceTotal = 0;
455
456  // LED
457  Serial.println("LED up");
458  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
459    analogWrite(LEDpin, fadeValue);
460    delay(50);
461  }
462
463  // reset
464  theDistance = 0;
465  theDistanceTotal = 0;
466
467  // LED
468  Serial.println("LED up");
469  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
470    analogWrite(LEDpin, fadeValue);
471    delay(50);
472  }
473
474  // reset
475  theDistance = 0;
476  theDistanceTotal = 0;
477
478  // LED
479  Serial.println("LED up");
480  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
481    analogWrite(LEDpin, fadeValue);
482    delay(50);
483  }
484
485  // reset
486  theDistance = 0;
487  theDistanceTotal = 0;
488
489  // LED
490  Serial.println("LED up");
491  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
492    analogWrite(LEDpin, fadeValue);
493    delay(50);
494  }
495
496  // reset
497  theDistance = 0;
498  theDistanceTotal = 0;
499
500  // LED
501  Serial.println("LED up");
502  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
503    analogWrite(LEDpin, fadeValue);
504    delay(50);
505  }
506
507  // reset
508  theDistance = 0;
509  theDistanceTotal = 0;
510
511  // LED
512  Serial.println("LED up");
513  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
514    analogWrite(LEDpin, fadeValue);
515    delay(50);
516  }
517
518  // reset
519  theDistance = 0;
520  theDistanceTotal = 0;
521
522  // LED
523  Serial.println("LED up");
524  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
525    analogWrite(LEDpin, fadeValue);
526    delay(50);
527  }
528
529  // reset
530  theDistance = 0;
531  theDistanceTotal = 0;
532
533  // LED
534  Serial.println("LED up");
535  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
536    analogWrite(LEDpin, fadeValue);
537    delay(50);
538  }
539
540  // reset
541  theDistance = 0;
542  theDistanceTotal = 0;
543
544  // LED
545  Serial.println("LED up");
546  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
547    analogWrite(LEDpin, fadeValue);
548    delay(50);
549  }
550
551  // reset
552  theDistance = 0;
553  theDistanceTotal = 0;
554
555  // LED
556  Serial.println("LED up");
557  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
558    analogWrite(LEDpin, fadeValue);
559    delay(50);
560  }
561
562  // reset
563  theDistance = 0;
564  theDistanceTotal = 0;
565
566  // LED
567  Serial.println("LED up");
568  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
569    analogWrite(LEDpin, fadeValue);
570    delay(50);
571  }
572
573  // reset
574  theDistance = 0;
575  theDistanceTotal = 0;
576
577  // LED
578  Serial.println("LED up");
579  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
580    analogWrite(LEDpin, fadeValue);
581    delay(50);
582  }
583
584  // reset
585  theDistance = 0;
586  theDistanceTotal = 0;
587
588  // LED
589  Serial.println("LED up");
590  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
591    analogWrite(LEDpin, fadeValue);
592    delay(50);
593  }
594
595  // reset
596  theDistance = 0;
597  theDistanceTotal = 0;
598
599  // LED
600  Serial.println("LED up");
601  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
602    analogWrite(LEDpin, fadeValue);
603    delay(50);
604  }
605
606  // reset
607  theDistance = 0;
608  theDistanceTotal = 0;
609
610  // LED
611  Serial.println("LED up");
612  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
613    analogWrite(LEDpin, fadeValue);
614    delay(50);
615  }
616
617  // reset
618  theDistance = 0;
619  theDistanceTotal = 0;
620
621  // LED
622  Serial.println("LED up");
623  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
624    analogWrite(LEDpin, fadeValue);
625    delay(50);
626  }
627
628  // reset
629  theDistance = 0;
630  theDistanceTotal = 0;
631
632  // LED
633  Serial.println("LED up");
634  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
635    analogWrite(LEDpin, fadeValue);
636    delay(50);
637  }
638
639  // reset
640  theDistance = 0;
641  theDistanceTotal = 0;
642
643  // LED
644  Serial.println("LED up");
645  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
646    analogWrite(LEDpin, fadeValue);
647    delay(50);
648  }
649
650  // reset
651  theDistance = 0;
652  theDistanceTotal = 0;
653
654  // LED
655  Serial.println("LED up");
656  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
657    analogWrite(LEDpin, fadeValue);
658    delay(50);
659  }
660
661  // reset
662  theDistance = 0;
663  theDistanceTotal = 0;
664
665  // LED
666  Serial.println("LED up");
667  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
668    analogWrite(LEDpin, fadeValue);
669    delay(50);
670  }
671
672  // reset
673  theDistance = 0;
674  theDistanceTotal = 0;
675
676  // LED
677  Serial.println("LED up");
678  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
679    analogWrite(LEDpin, fadeValue);
680    delay(50);
681  }
682
683  // reset
684  theDistance = 0;
685  theDistanceTotal = 0;
686
687  // LED
688  Serial.println("LED up");
689  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
690    analogWrite(LEDpin, fadeValue);
691    delay(50);
692  }
693
694  // reset
695  theDistance = 0;
696  theDistanceTotal = 0;
697
698  // LED
699  Serial.println("LED up");
700  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
701    analogWrite(LEDpin, fadeValue);
702    delay(50);
703  }
704
705  // reset
706  theDistance = 0;
707  theDistanceTotal = 0;
708
709  // LED
710  Serial.println("LED up");
711  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
712    analogWrite(LEDpin, fadeValue);
713    delay(50);
714  }
715
716  // reset
717  theDistance = 0;
718  theDistanceTotal = 0;
719
720  // LED
721  Serial.println("LED up");
722  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
723    analogWrite(LEDpin, fadeValue);
724    delay(50);
725  }
726
727  // reset
728  theDistance = 0;
729  theDistanceTotal = 0;
730
731  // LED
732  Serial.println("LED up");
733  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
734    analogWrite(LEDpin, fadeValue);
735    delay(50);
736  }
737
738  // reset
739  theDistance = 0;
740  theDistanceTotal = 0;
741
742  // LED
743  Serial.println("LED up");
744  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
745    analogWrite(LEDpin, fadeValue);
746    delay(50);
747  }
748
749  // reset
750  theDistance = 0;
751  theDistanceTotal = 0;
752
753  // LED
754  Serial.println("LED up");
755  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
756    analogWrite(LEDpin, fadeValue);
757    delay(50);
758  }
759
760  // reset
761  theDistance = 0;
762  theDistanceTotal = 0;
763
764  // LED
765  Serial.println("LED up");
766  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
767    analogWrite(LEDpin, fadeValue);
768    delay(50);
769  }
770
771  // reset
772  theDistance = 0;
773  theDistanceTotal = 0;
774
775  // LED
776  Serial.println("LED up");
777  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
778    analogWrite(LEDpin, fadeValue);
779    delay(50);
780  }
781
782  // reset
783  theDistance = 0;
784  theDistanceTotal = 0;
785
786  // LED
787  Serial.println("LED up");
788  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
789    analogWrite(LEDpin, fadeValue);
790    delay(50);
791  }
792
793  // reset
794  theDistance = 0;
795  theDistanceTotal = 0;
796
797  // LED
798  Serial.println("LED up");
799  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
800    analogWrite(LEDpin, fadeValue);
801    delay(50);
802  }
803
804  // reset
805  theDistance = 0;
806  theDistanceTotal = 0;
807
808  // LED
809  Serial.println("LED up");
810  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
811    analogWrite(LEDpin, fadeValue);
812    delay(50);
813  }
814
815  // reset
816  theDistance = 0;
817  theDistanceTotal = 0;
818
819  // LED
820  Serial.println("LED up");
821  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
822    analogWrite(LEDpin, fadeValue);
823    delay(50);
824  }
825
826  // reset
827  theDistance = 0;
828  theDistanceTotal = 0;
829
830  // LED
831  Serial.println("LED up");
832  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
833    analogWrite(LEDpin, fadeValue);
834    delay(50);
835  }
836
837  // reset
838  theDistance = 0;
839  theDistanceTotal = 0;
840
841  // LED
842  Serial.println("LED up");
843  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
844    analogWrite(LEDpin, fadeValue);
845    delay(50);
846  }
847
848  // reset
849  theDistance = 0;
850  theDistanceTotal = 0;
851
852  // LED
853  Serial.println("LED up");
854  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
855    analogWrite(LEDpin, fadeValue);
856    delay(50);
857  }
858
859  // reset
860  theDistance = 0;
861  theDistanceTotal = 0;
862
863  // LED
864  Serial.println("LED up");
865  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
866    analogWrite(LEDpin, fadeValue);
867    delay(50);
868  }
869
870  // reset
871  theDistance = 0;
872  theDistanceTotal = 0;
873
874  // LED
875  Serial.println("LED up");
876  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
877    analogWrite(LEDpin, fadeValue);
878    delay(50);
879  }
880
881  // reset
882  theDistance = 0;
883  theDistanceTotal = 0;
884
885  // LED
886  Serial.println("LED up");
887  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
888    analogWrite(LEDpin, fadeValue);
889    delay(50);
890  }
891
892  // reset
893  theDistance = 0;
894  theDistanceTotal = 0;
895
896  // LED
897  Serial.println("LED up");
898  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
899    analogWrite(LEDpin, fadeValue);
900    delay(50);
901  }
902
903  // reset
904  theDistance = 0;
905  theDistanceTotal = 0;
906
907  // LED
908  Serial.println("LED up");
909  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
910    analogWrite(LEDpin, fadeValue);
911    delay(50);
912  }
913
914  // reset
915  theDistance = 0;
916  theDistanceTotal = 0;
917
918  // LED
919  Serial.println("LED up");
920  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
921    analogWrite(LEDpin, fadeValue);
922    delay(50);
923  }
924
925  // reset
926  theDistance = 0;
927  theDistanceTotal = 0;
928
929  // LED
930  Serial.println("LED up");
931  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
932    analogWrite(LEDpin, fadeValue);
933    delay(50);
934  }
935
936  // reset
937  theDistance = 0;
938  theDistanceTotal = 0;
939
940  // LED
941  Serial.println("LED up");
942  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
943    analogWrite(LEDpin, fadeValue);
944    delay(50);
945  }
946
947  // reset
948  theDistance = 0;
949  theDistanceTotal = 0;
950
951  // LED
952  Serial.println("LED up");
953  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
954    analogWrite(LEDpin, fadeValue);
955    delay(50);
956  }
957
958  // reset
959  theDistance = 0;
960  theDistanceTotal = 0;
961
962  // LED
963  Serial.println("LED up");
964  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
965    analogWrite(LEDpin, fadeValue);
966    delay(50);
967  }
968
969  // reset
970  theDistance = 0;
971  theDistanceTotal = 0;
972
973  // LED
974  Serial.println("LED up");
975  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
976    analogWrite(LEDpin, fadeValue);
977    delay(50);
978  }
979
980  // reset
981  theDistance = 0;
982  theDistanceTotal = 0;
983
984  // LED
985  Serial.println("LED up");
986  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
987    analogWrite(LEDpin, fadeValue);
988    delay(50);
989  }
990
991  // reset
992  theDistance = 0;
993  theDistanceTotal = 0;
994
995  // LED
996  Serial.println("LED up");
997  for (int fadeValue = 255; fadeValue > 10; fadeValue -- 10) {
998    analogWrite(LEDpin, fadeValue);
999    delay(50);
1000  }

```

-  If you've never used our Robotics Board before you'll need to read the user guide and install the right software and drivers.
-  You'll also need to grab and install the [NewPing library](#) as well.
- New line. Open up your Arduino software and [copy our code](#) into a new project window.
- Lines 30 & 31 control how long the Servo will move when opening and closing the wings. We find that 20000 ms is about right. You can change the time by changing those values.
- Line 91 controls how long the Servo pauses to wait for your audio clip. Since we want the head to spin constantly we just made this a big fat zero.

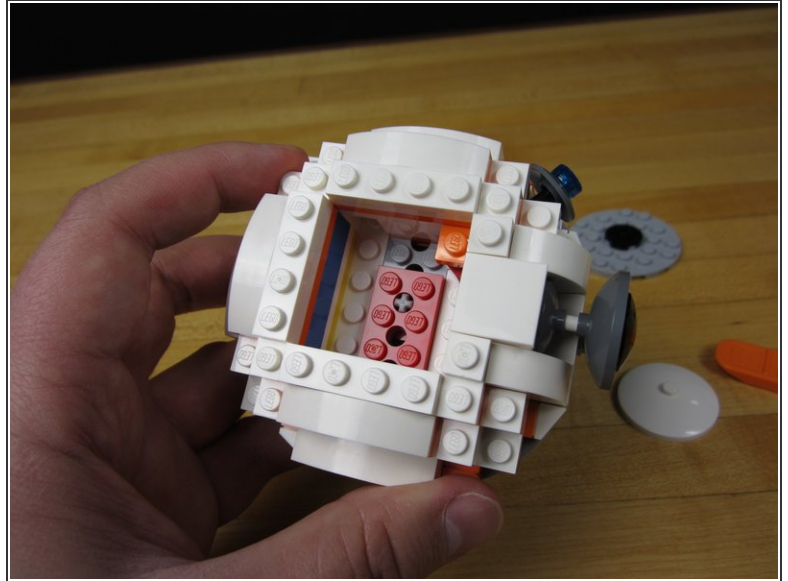
## Step 17 — Test Things Out



- At this point it doesn't hurt to just test everything out to make sure things are working properly.
- Attach some headphone or some small speakers to the MP3 player. If you have small desktop (computer) speakers that are wall powered, use them. They work best with this module.
- Plug your system into a USB power source (computer or wall) and see if everything works. Wave your hand in front of the distance sensor to start everything off.

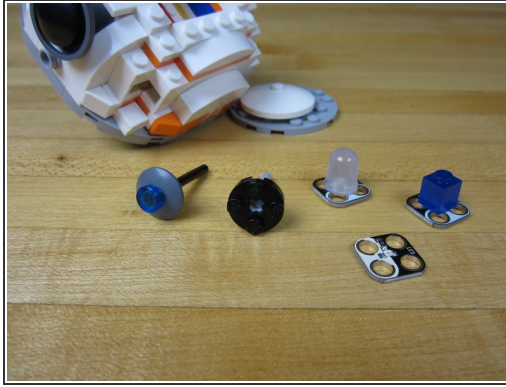
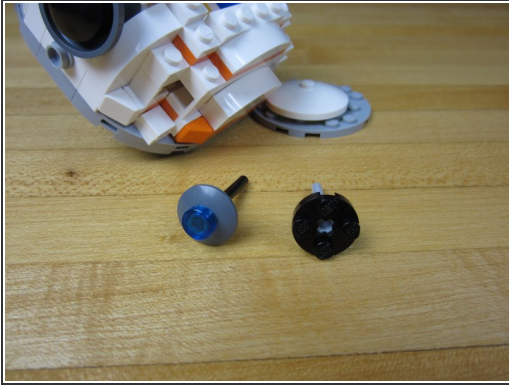


## Step 18 — Prepare the Head



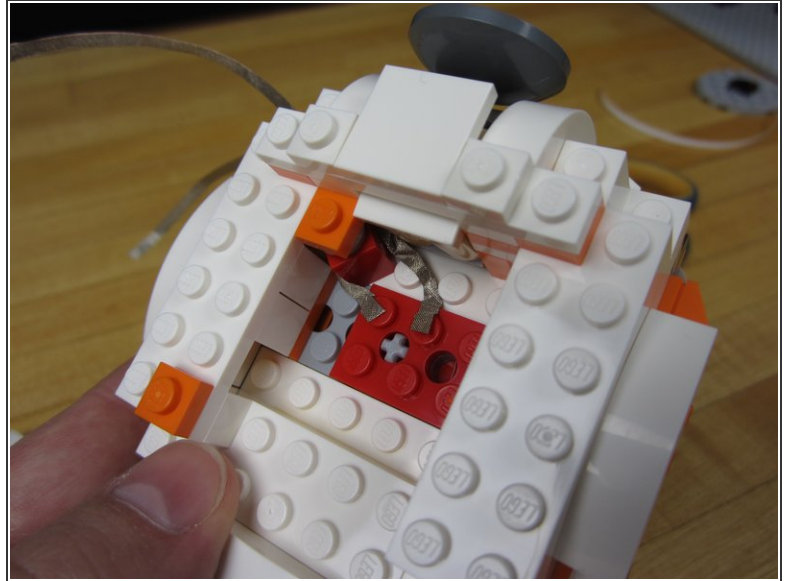
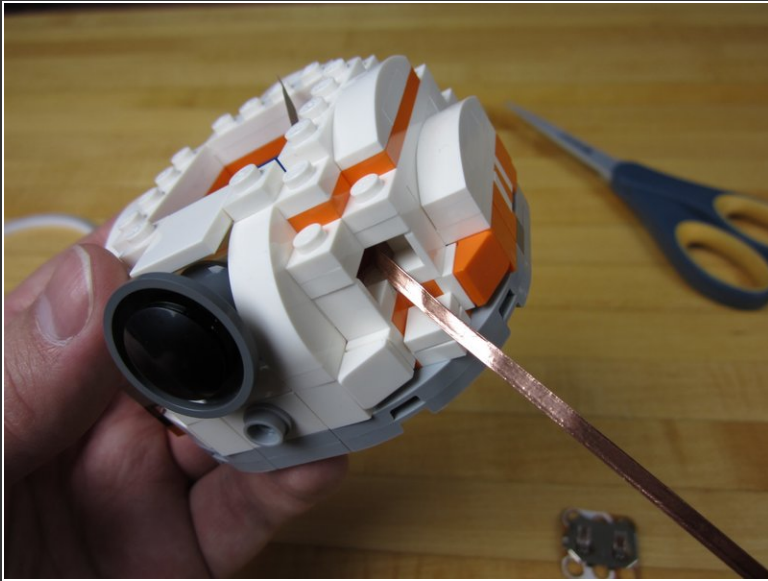
- i** To give our BB-8 a bit of "flair" we added a blue LED to his head. This was rather easy as there is plenty of room inside for a small battery.
- Using a prying tool, remove the top of his head.
  - Clear out the couple of spacer bricks on the inside, giving you plenty of room to work with.

## Step 19 — Make a LED Holder



- Remove the blue "holographic" emitter.
- Use a small axle and attach it to a 2x2 round plate.
- We chose to use a blue "novelty brick" Crazy Circuits LED, but a 10mm or SMT LED will work just fine.

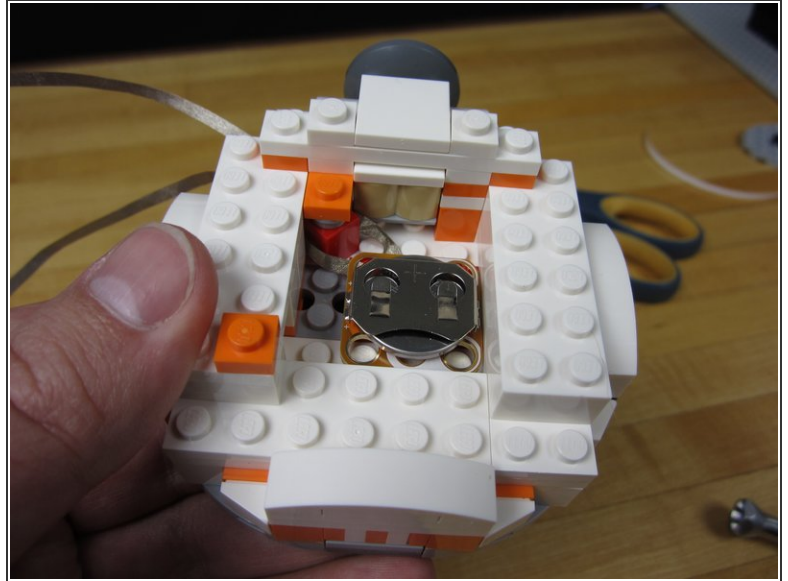
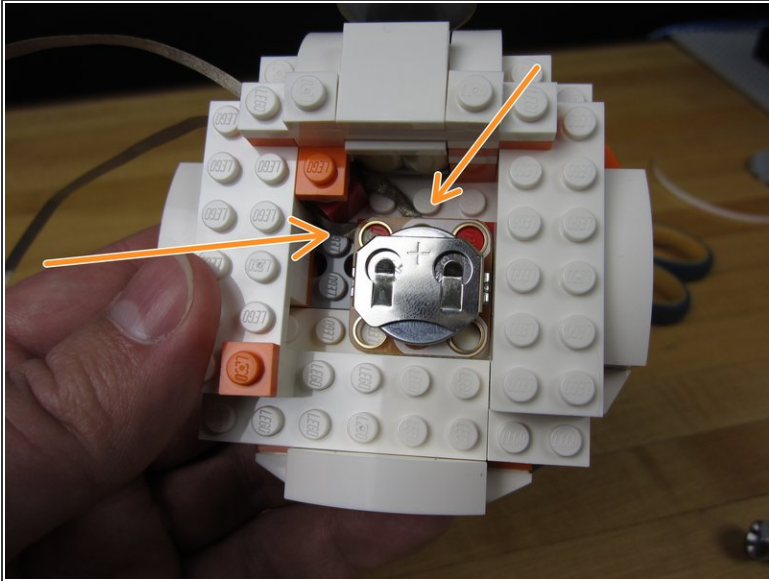
## Step 20 — Run Tape Inside



- Run two lines of Nylon Conductive Tape from the outside to the inside of the head.
- Press them down onto two studs.
- ⓘ This is a very annoying part of the build. Remove more parts of the head if your fingers are having issues.



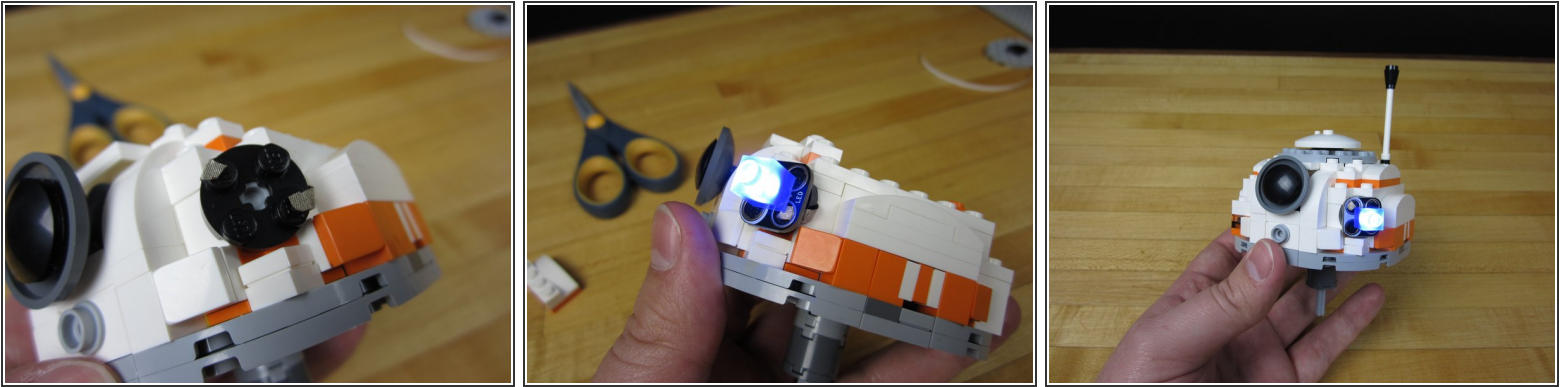
## Step 21 — Attach Battery





- Put a battery into the Crazy Circuits CR2032 Holder.
- Place the holder inside the head, on top of the studs.

**⚠ Make sure that one line of tape goes to the White (Negative) side of the battery holder and the other goes to the Orange (Positive) side.**

## Step 22 — Connect LED



- Push your LED holder (2x2 Round Plate) into place.
  - Trim your lines of tape and attach them to the studs.
  - Connect your LED. (If it doesn't turn on, rotate it. You probably have it put on backwards compared to you battery holder.)
  - Rebuild the head. Attach it to the body.
-  To easily turn your LED On and Off, just remove it.
-  When putting the head back on go very slowly. The connecting axle will push up through the head and disconnect your battery. Stop pushing when you feel a slight resistance.

## Step 23 — Enjoy!



- Your BB-8 is now complete! Maybe this IS the droid you're looking for!
- Apply this build to other LEGO projects. We more or less did the same build with our X-Wing and Clone ARC Fighter.