The Watt Governor

R. I. LEINE

2004

Figure 1: Flyball governor.

1 Description of the Watt governor

James Watt used the so-called ”flyball governor” to regulate the speed of his steam engines (1787). In the picture on the right you see such a flyball governor. It consists of two flyballs which are connected to the spindle by two flyball arms. The spindle of the governor is directly connected to the shaft of the steam engine. The flyballs move upward when the spindle speed increases due to the centrifugal forces. The flyball arms are connected to a throttle valve that regulates the steam input to the engine. If the spindle speed increases, then the flyballs arms move upward thereby closing the throttle valve, which reduces the steam input and the engine is therefore slowed down. The flyball governor therefore regulates the speed of the steam engine. It is in fact one of the first feed-backward mechanisms. The amount of feed-backward, i.e. the ”gain”, is determined by the kinematics between the flyball arms and the throttle valve. The stationary movement of the combined engine-governor system is an equilibrium of the system. If everything goes well, then this equilibrium is asymptotically stable, i.e. a disturbance in the load of the engine will die out and the system returns to stationary movement with a constant desired engine speed. However, if the gain is taken too large (if the flyballs influence the throttle valve too much), then the equilibrium becomes unstable and a stable limit cycle is created. This is what we call a Hopf bifurcation. The speed of the engine as well as the height of the flyballs will therefore not be constant for a large gain but will oscillate (which is undesirable).

2 The LEGO Watt Governor

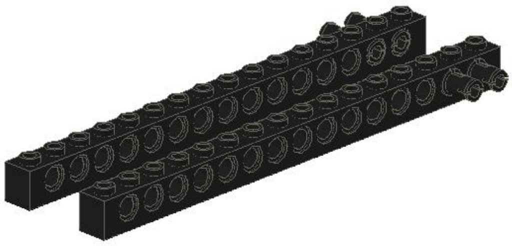
In order to show to students of my Nonlinear Dynamics course at the ETH Zrich what a Hopf bifurcation is, I designed a model of a Watt Governor using only ordinary LEGO bricks as well as the LEGO Mindstorms Robotic set. The flyball governor is made using LEGO TECHNIC bricks. The spindle is driven by an 9V LEGO electromotor. The height of the flyballs is measured using a rotation sensor which is connected to the RCX, being a LEGO programmable brick. On the RCX runs a computer program that determines the voltage to the electromotor. If the flyballs rise, then the electrical power to the electromotor is reduced by the RCX. The RCX together with the electromotor therefore represents the steam engine and throttle valve. In the next sections you will find the building instructions and part lists as well as the computer program for the RCX. The computer program has been written in NQC (Not Quite C), a programming language specially designed for the RCX by Dave Baum.

3 Building Instructions

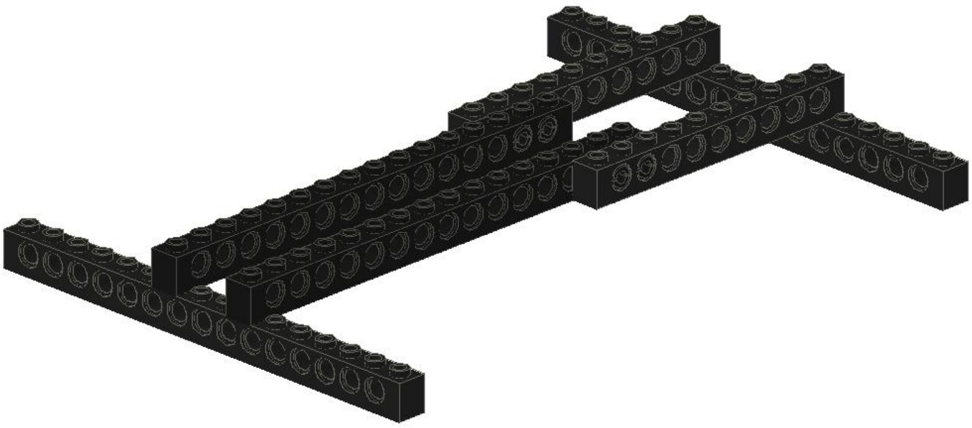
In the pages hereafter, you find the building instructions for the Watt governor explained in 57 steps. Steps 1 to 49 show how the submodels are built. Steps 50 to 57 explain the assembly of all the submodels.

Submodel: Base

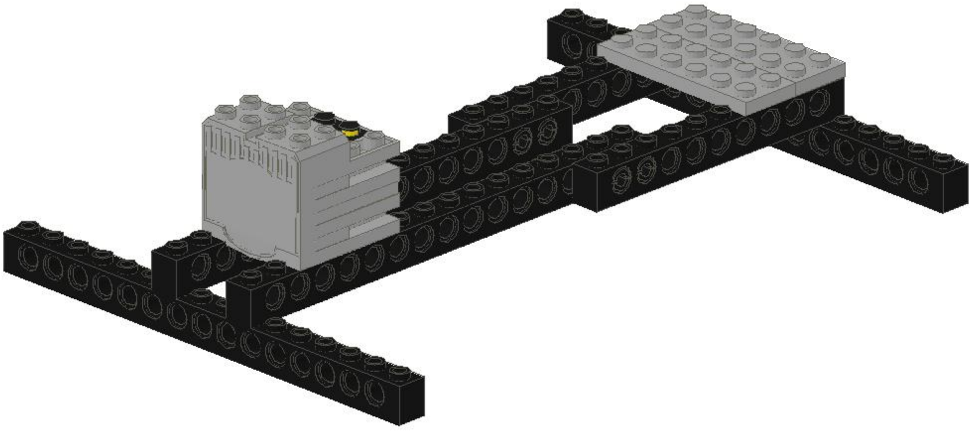
1



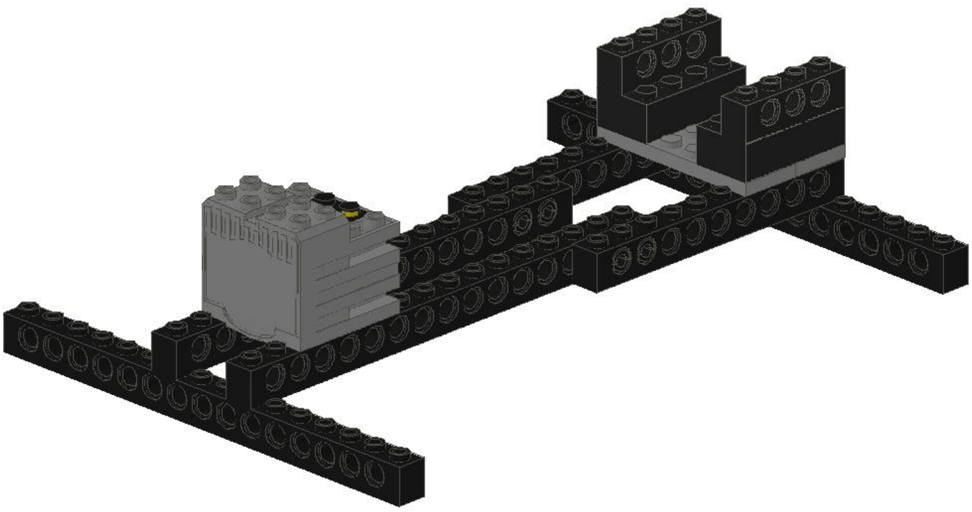
2



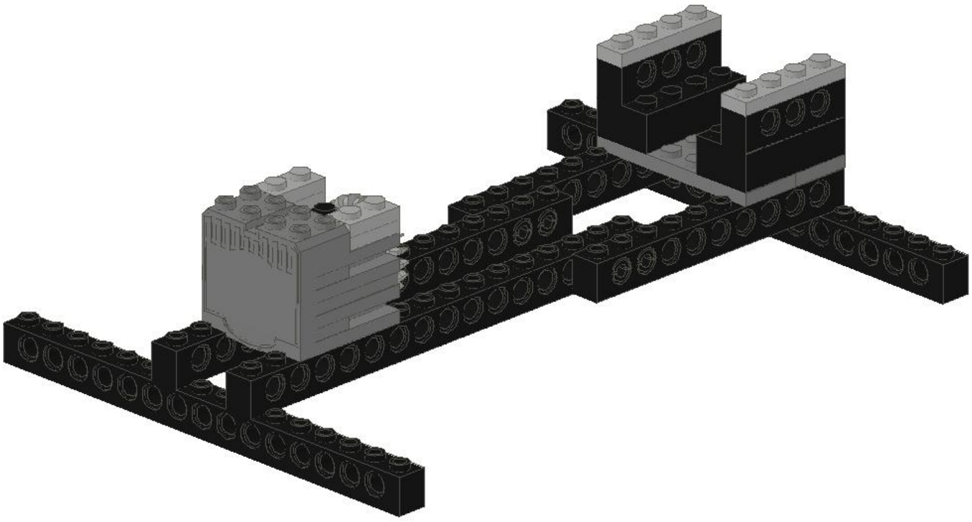
3



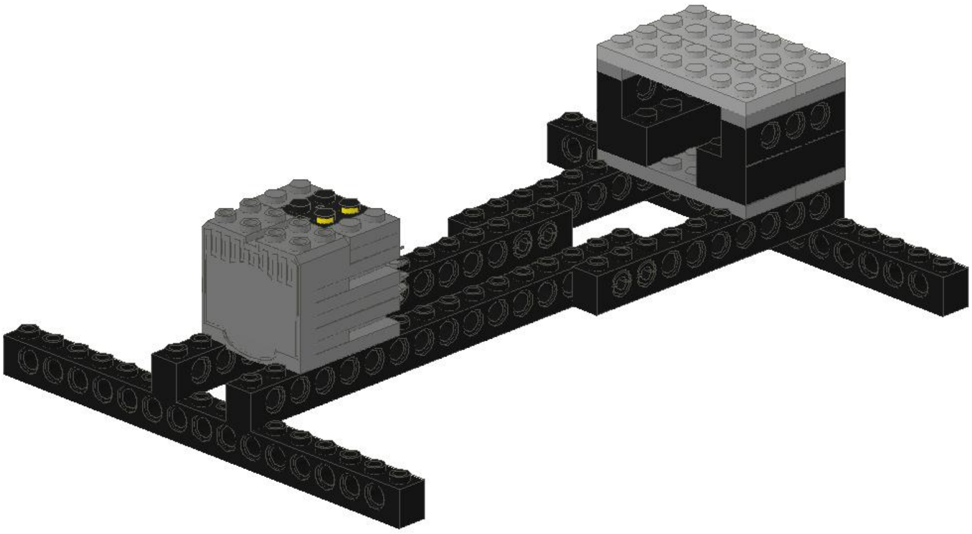
4



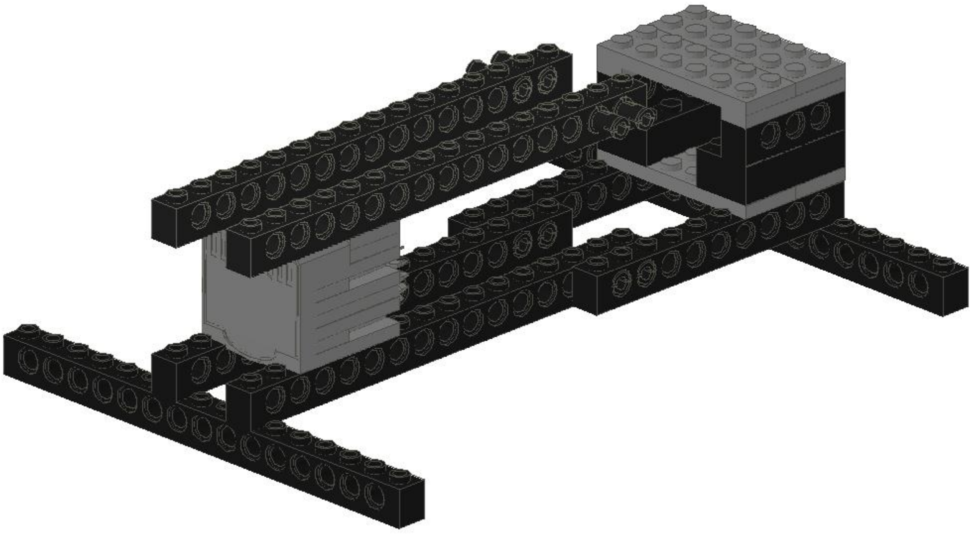
5



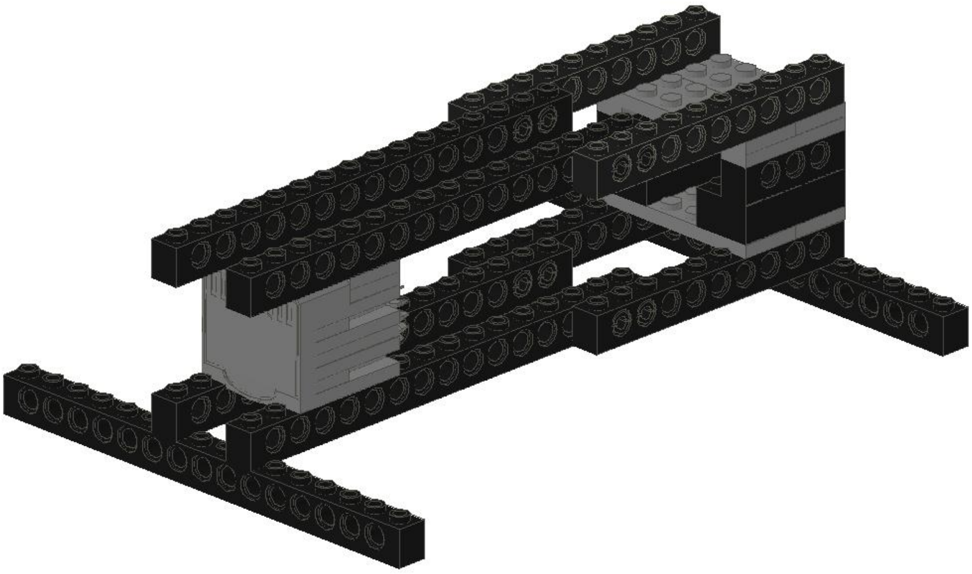
6



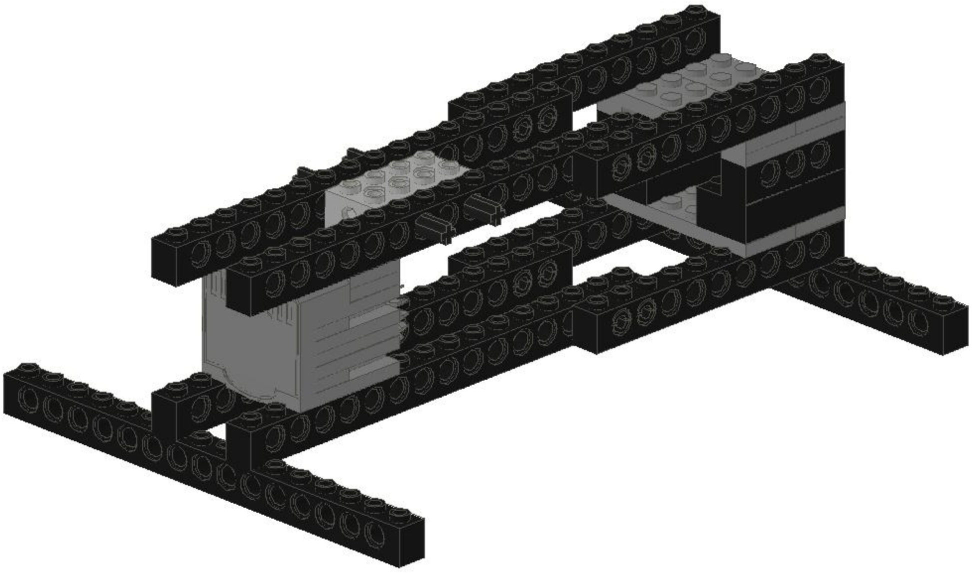
7



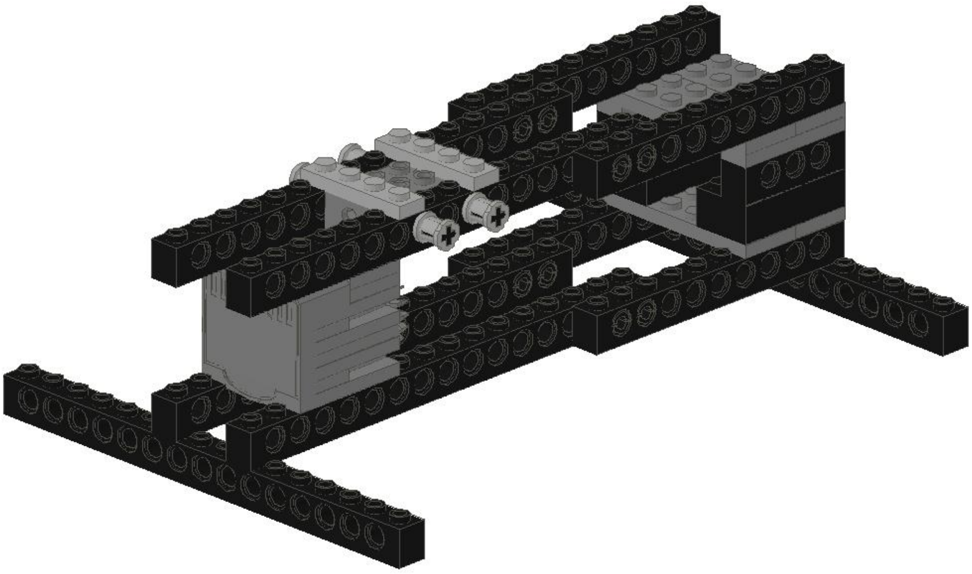
8



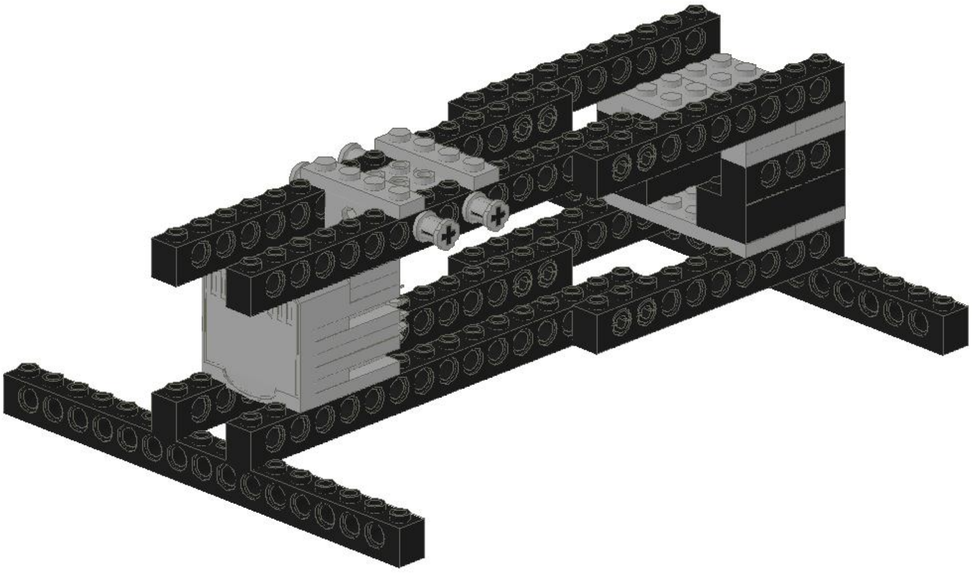
9



10



11

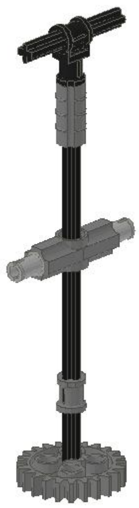


Submodel: Spindle

12

13

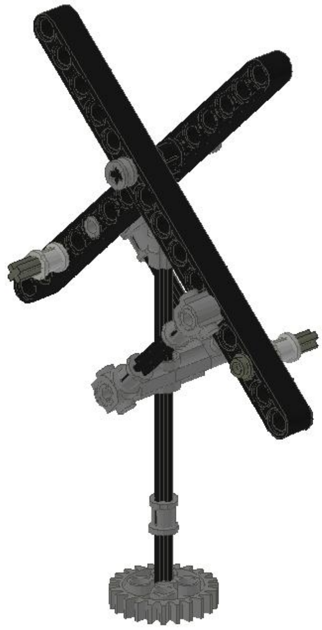
14



15

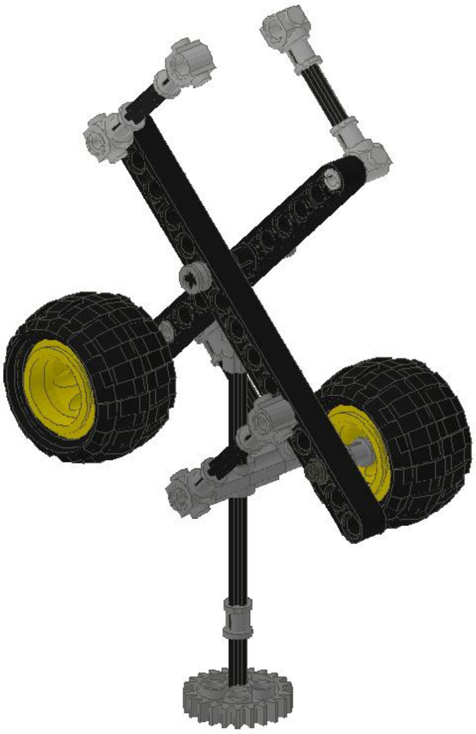
16

17

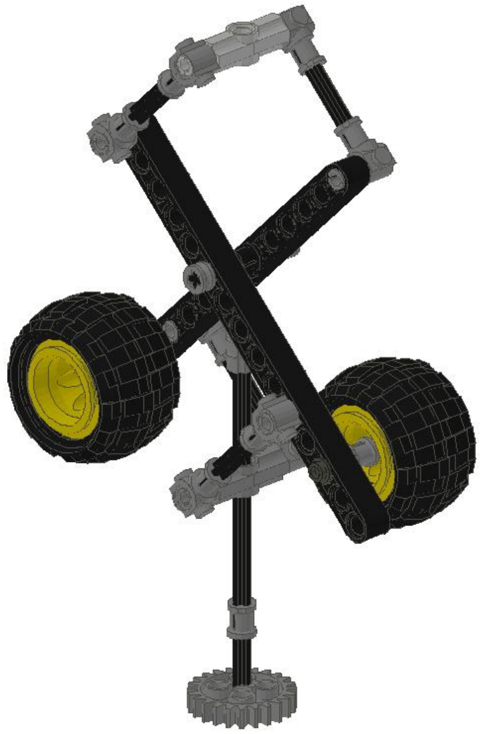


18

19



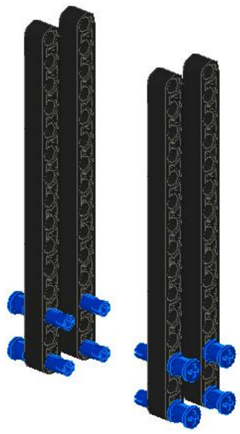
20



21

Submodel: Tower

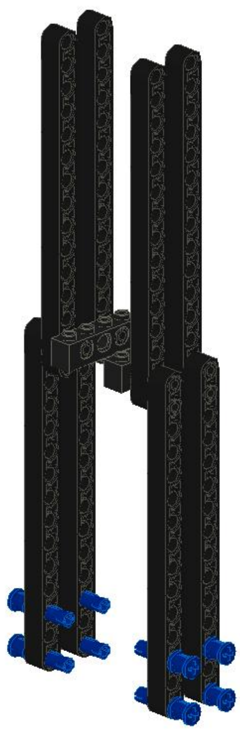
22



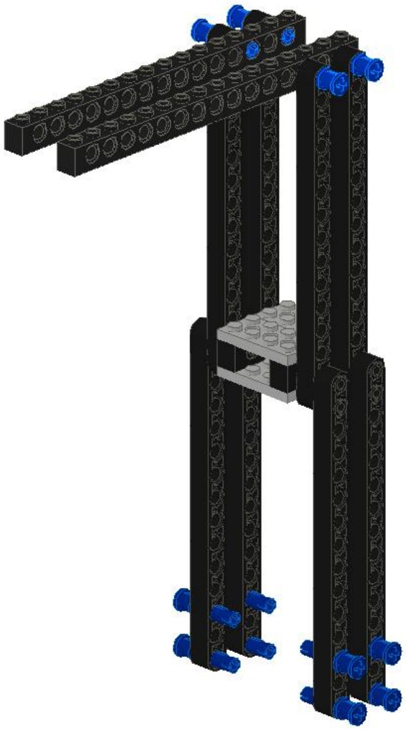
23

24

25



26

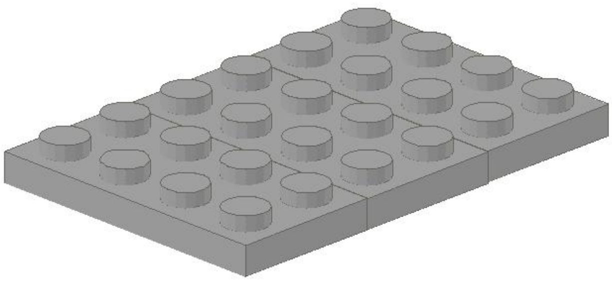


27

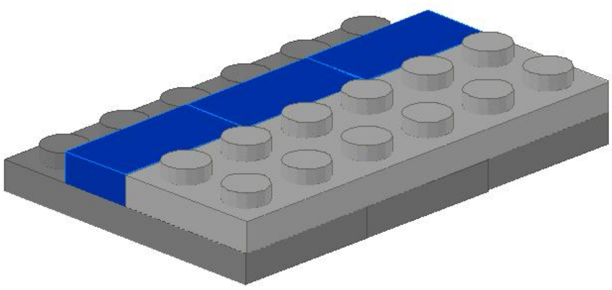
28

Submodel: Linear Displacement Sensor

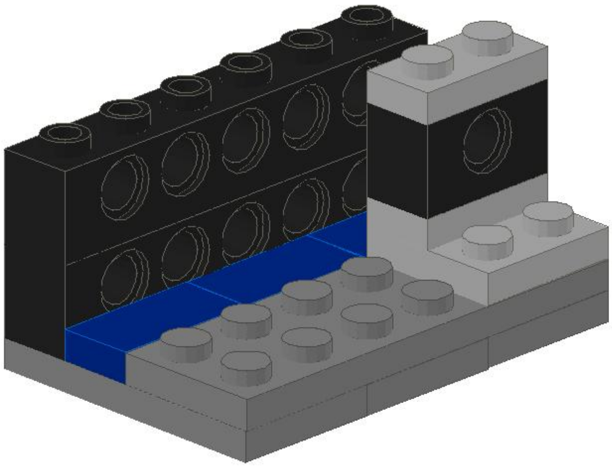
29



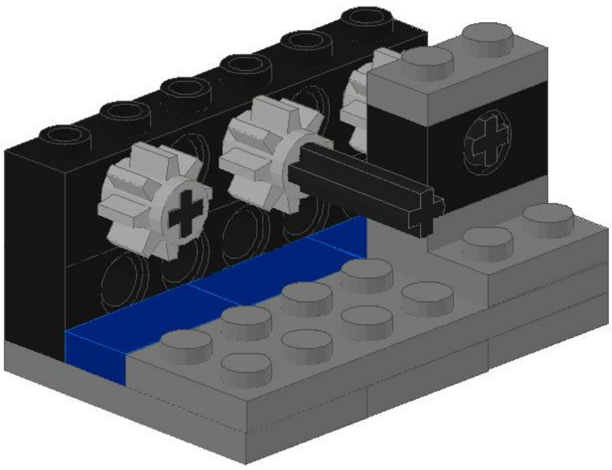
30



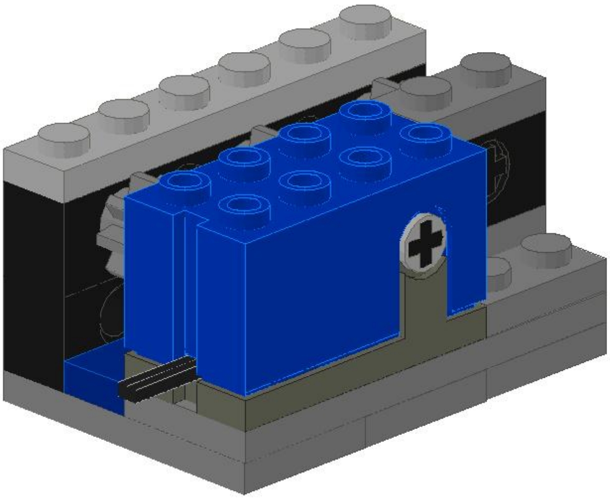
31



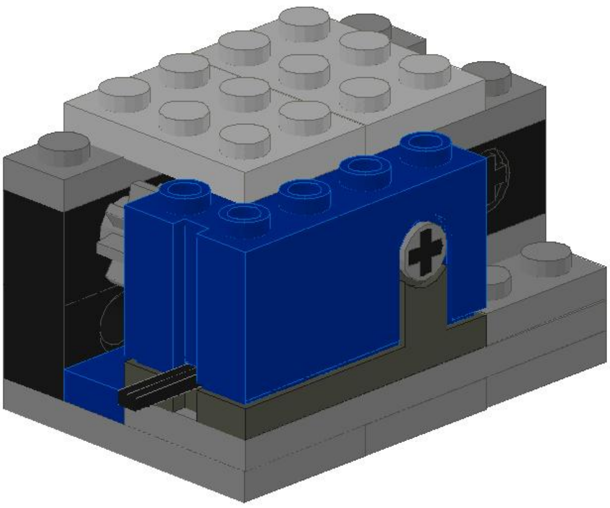
32



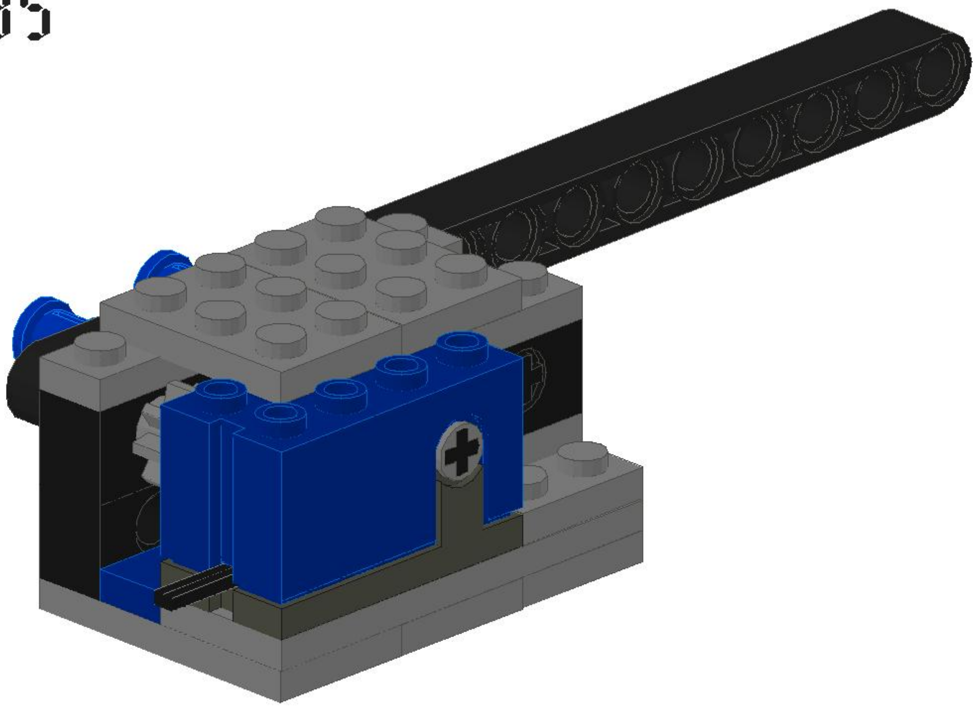
33



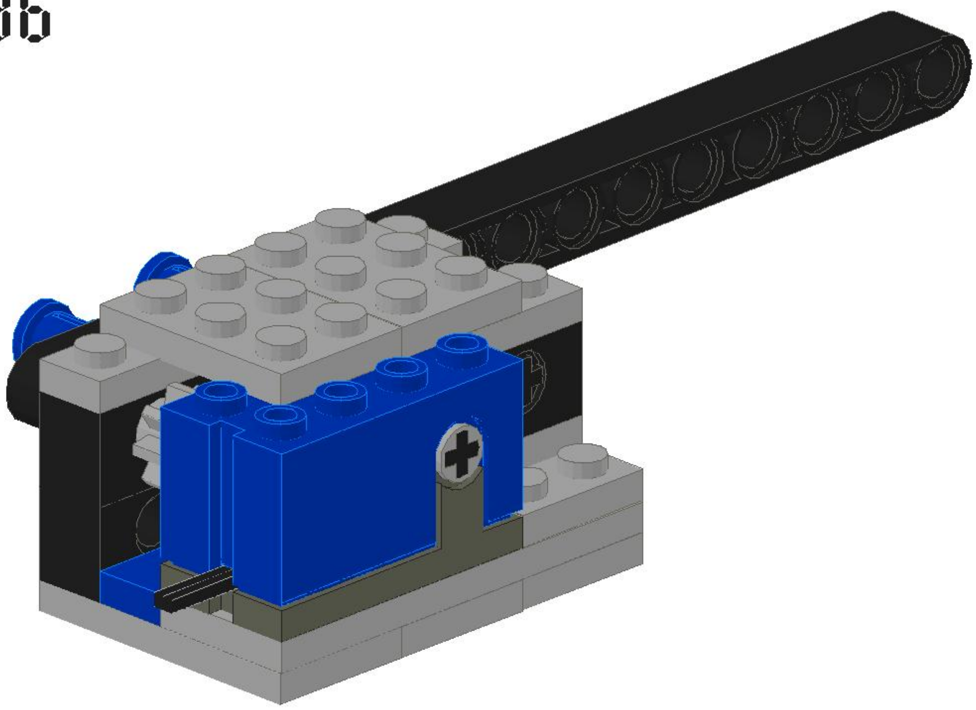
34



35

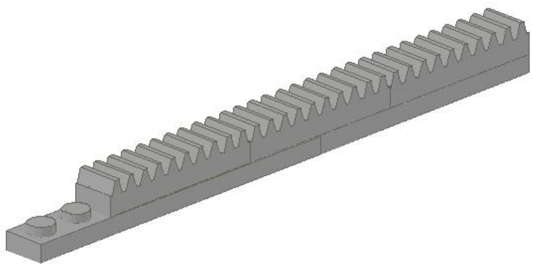


36

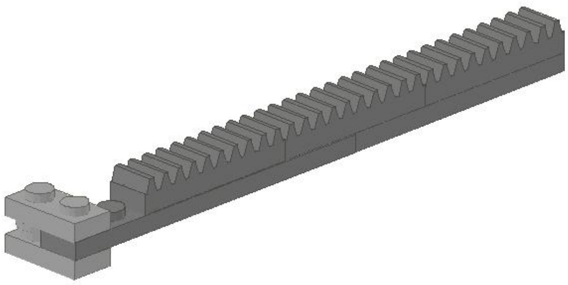


Submodel: Rack

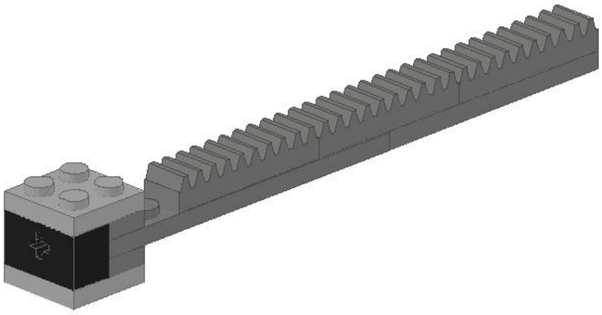
37



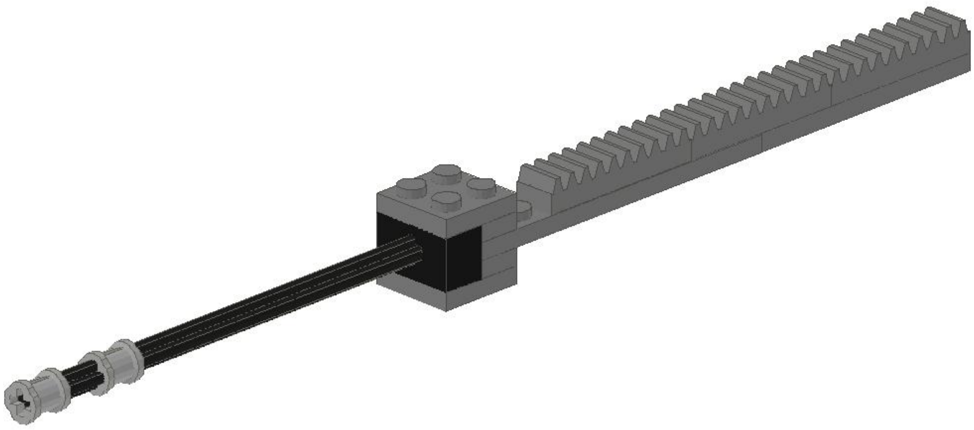
38



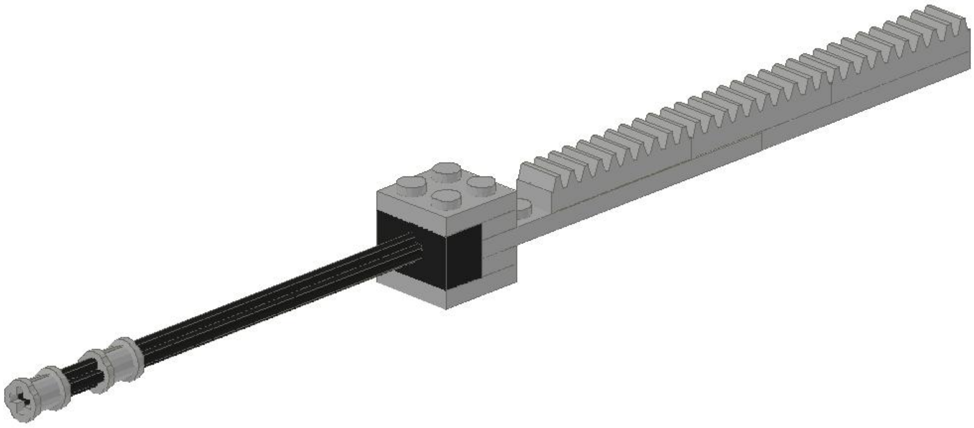
39



40

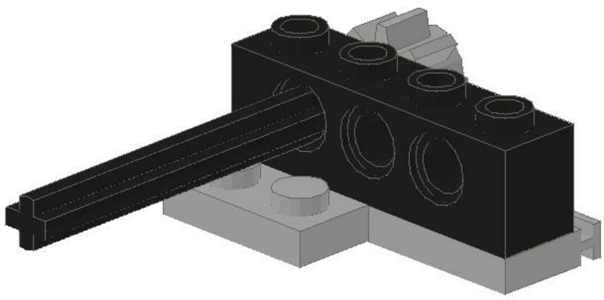


41

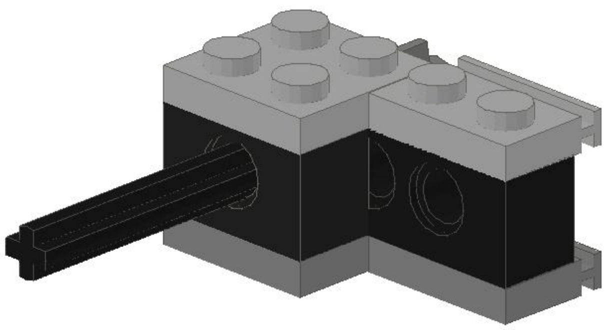


Submodel: Inertia Left

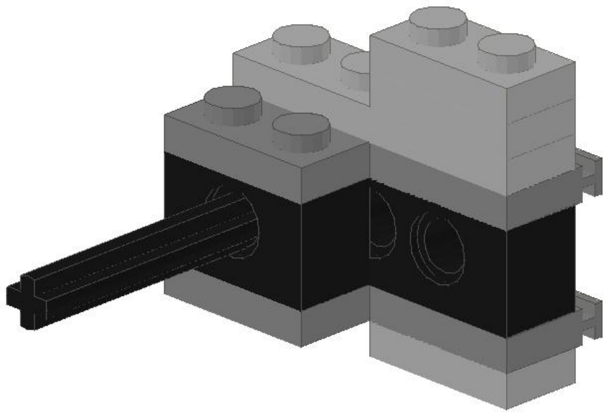
42



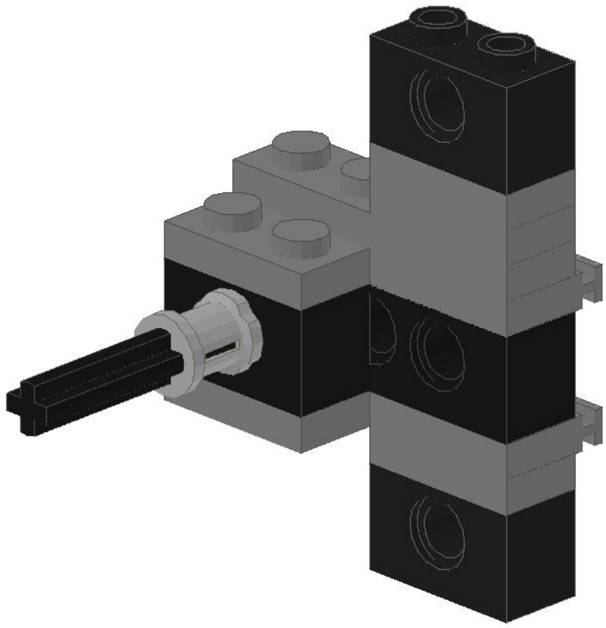
43



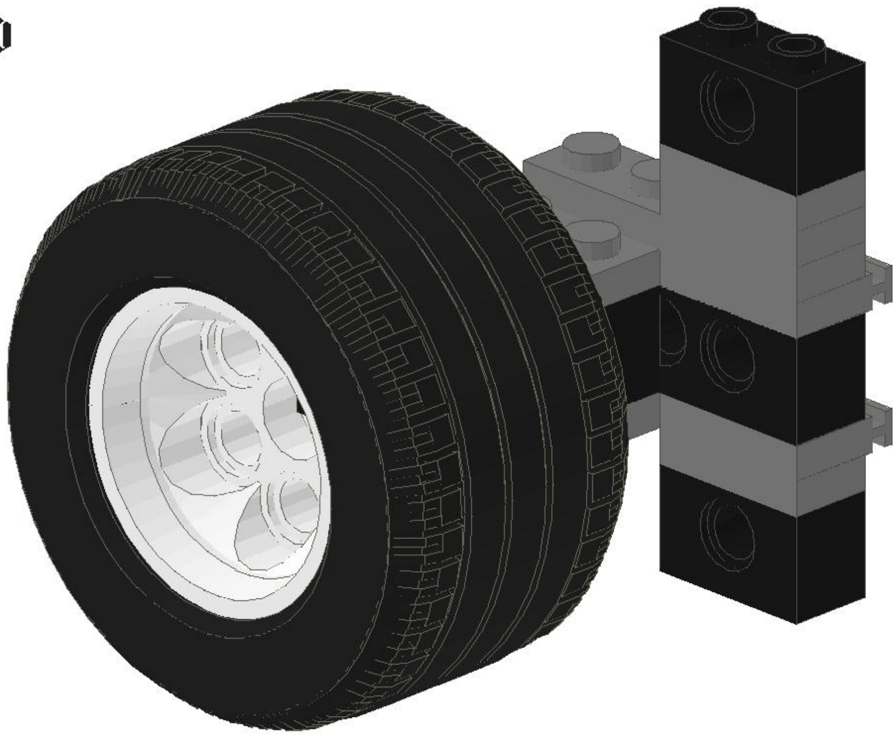
44



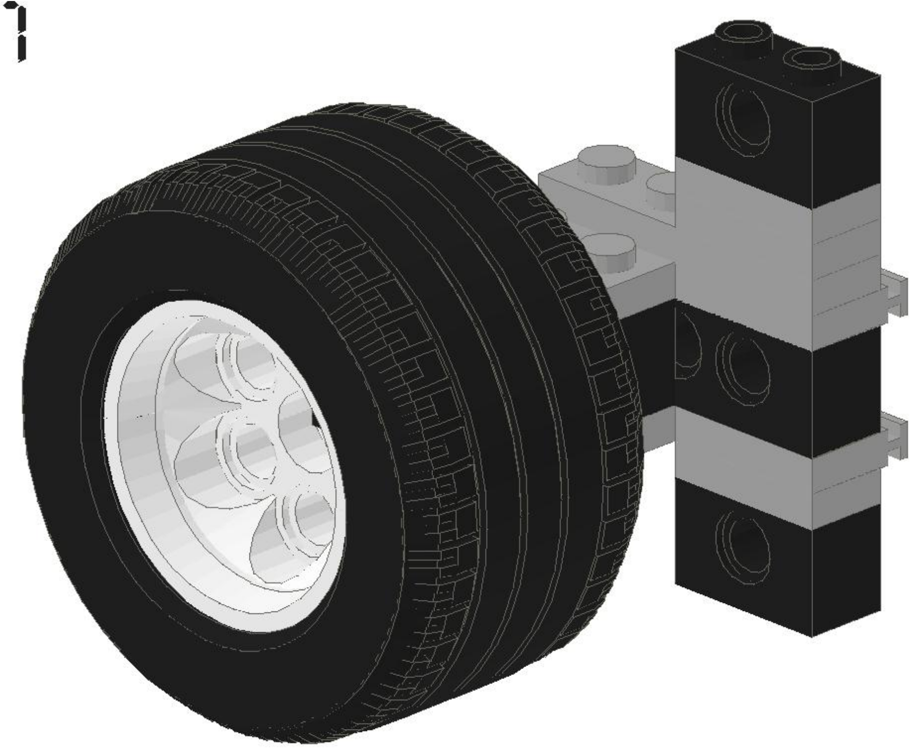
45



46

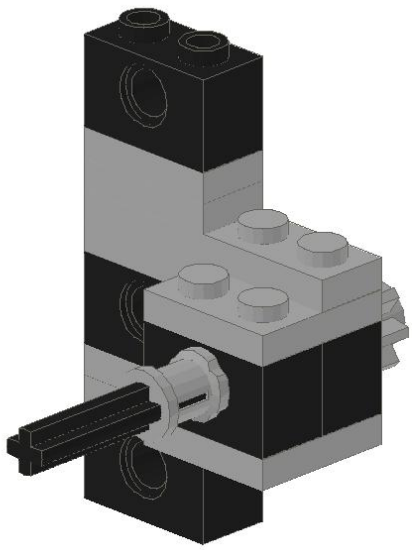


47

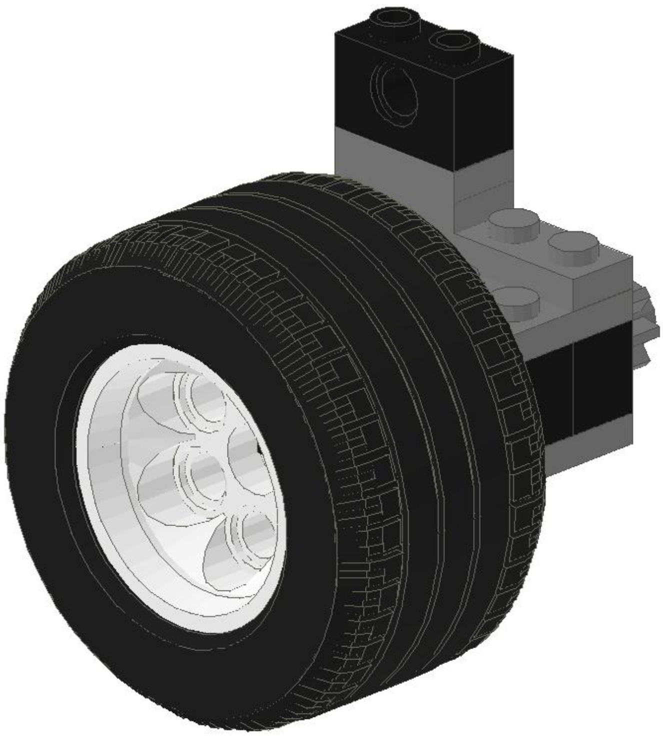


Submodel: Inertia Right

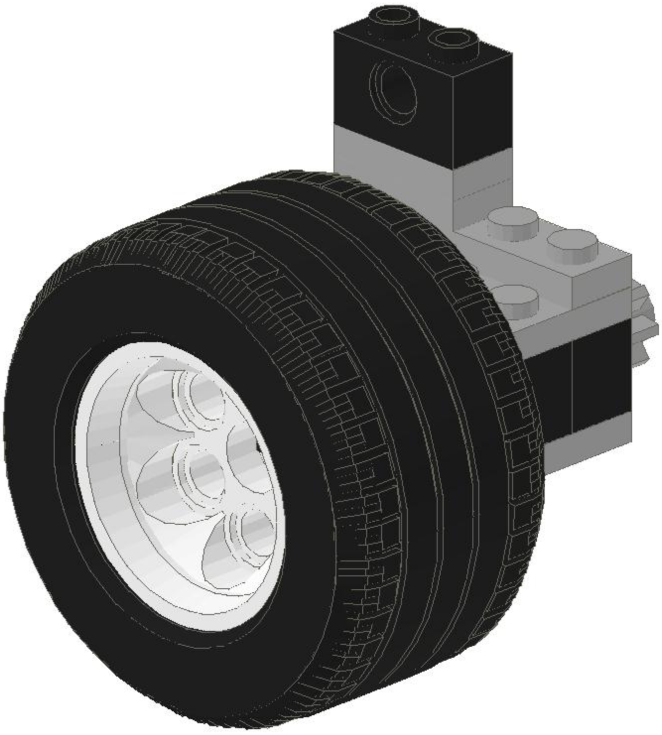
48



49

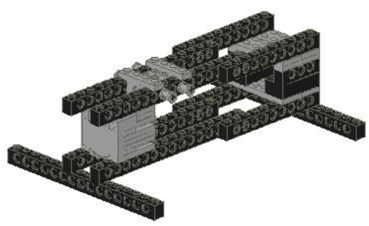


50



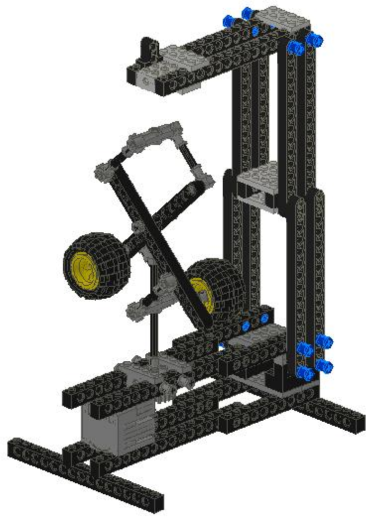
Total Assembly

51

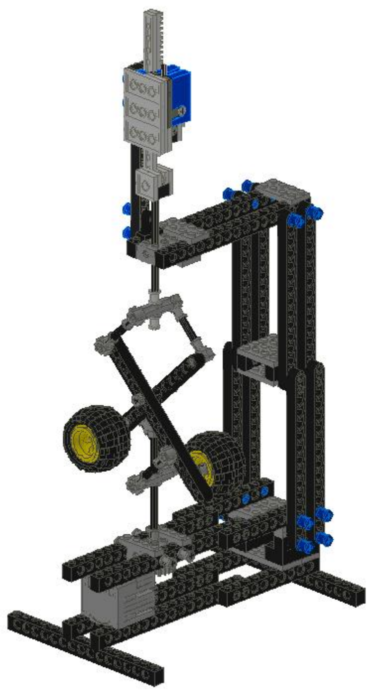


52

53



54



55

56

57

4 Part Lists

Below, you find the part lists of the submodels and the total assembly of the LEGO model. The part numbers are according to the definition of LEGO-CAD program *Ldraw* (http://www.ldraw.org/).

```
Submodel: Base
  No. Color      Part no.     Part name
------------------------------------------------------------------------------
    2 Black      3001.DAT     Brick  2 x  4
    1 Light-Gray 3709A.DAT    Brick  2 x  4 with Top/Side/End Holes
    1 Black      5306.DAT     Electric Brick  2 x  2 x  2/3 with Wire End
    1 Light-Gray 71427C01.DAT Electric Technic Mini-Motor  9v
    4 Light-Gray 3023.DAT     Plate  1 x  2
    4 Light-Gray 3710.DAT     Plate  1 x  4
    4 Light-Gray 3795.DAT     Plate  2 x  6
    2 Black      3706.DAT     Technic Axle  6
    2 Black      3701.DAT     Technic Brick  1 x  4 with Holes
    4 Black      2730.DAT     Technic Brick  1 x 10 with Holes
    6 Black      3703.DAT     Technic Brick  1 x 16 with Holes
    4 Light-Gray 3713.DAT     Technic Bush
    1 Light-Gray 3650A.DAT    Technic Gear 24 Tooth Crown
    8 Black      2780.dat     Technic Pin with Friction and Slots
```

```
Submodel: Spindle
  No. Color      Part no.     Part name
------------------------------------------------------------------------------
    2 Light-Gray 32034.DAT    Technic Angle Connector #2
    1 Black      3704.DAT     Technic Axle  2
    2 Dark-Gray  6587.DAT     Technic Axle  3 with Stud
    5 Black      3705.DAT     Technic Axle  4
    1 Black      3708.DAT     Technic Axle 12
    1 Light-Gray 6538A.DAT    Technic Axle Joiner
    4 Light-Gray 3749.DAT     Technic Axle Pin
    3 Light-Gray 3713.DAT     Technic Bush
    2 Light-Gray 4265C.DAT    Technic Bush 1/2 Smooth
    8 Light-Gray 3651.DAT     Technic Connector
    1 Black      32039.DAT    Technic Connector with Axlehole
    1 Light-Gray 3648.DAT     Technic Gear 24 Tooth
    2 Black      32278.DAT    Technic Liftarm  1 x 15 Straight
    2 Light-Gray 3673.DAT     Technic Pin
    2 Light-Gray 4459.DAT     Technic Pin with Friction
    2 Black      6579.DAT     Tyre 43.2 x 28 Balloon Small
    2 Yellow     6580.DAT     Wheel 43.2 x 28 Balloon Small
```

```
Submodel: Tower
  No. Color      Part no.     Part name
------------------------------------------------------------------------------
    1 Light-Gray 3709A.DAT    Brick  2 x  4 with Top/Side/End Holes
    5 Light-Gray 3020.DAT     Plate  2 x  4
    2 Black      3701.DAT     Technic Brick  1 x  4 with Holes
    2 Black      3703.DAT     Technic Brick  1 x 16 with Holes
    8 Black      32278.DAT    Technic Liftarm  1 x 15 Straight
    4 Black      6558.DAT     Technic Pin Long with Friction
   12 Blue       32054.DAT    Technic Pin Long with Stop Bush
    4 Black      4459.DAT     Technic Pin with Friction
    2 Light-Gray 3709B.DAT    Technic Plate  2 x  4 with Holes
    1 Black      32530.dat    Technic Pin Joiner Plate  1 x  2 x  1 & 2/3
```

Submodel: Linear Displacement Sensor

No.	Color	Part no.	Part name
1	Blue	2977C01.DAT	Electric Rotation Sensor
2	Light-Gray	3023.DAT	Plate 1 x 2
1	Light-Gray	3666.DAT	Plate 1 x 6
1	Light-Gray	3022.DAT	Plate 2 x 2
2	Light-Gray	3021.DAT	Plate 2 x 3
3	Light-Gray	3020.DAT	Plate 2 x 4
1	Light-Gray	3795.DAT	Plate 2 x 6
1	Black	3704.DAT	Technic Axle 2
1	Black	4519.DAT	Technic Axle 3
1	Black	3705.DAT	Technic Axle 4
1	Black	3700.DAT	Technic Brick 1 x 2 with Hole
2	Black	3894.DAT	Technic Brick 1 x 6 with Holes
3	Light-Gray	3647.DAT	Technic Gear 8 Tooth
1	Black	32278.DAT	Technic Liftarm 1 x 15 Straight
2	Blue	32054.DAT	Technic Pin Long with Stop Bush
3	Blue	3069A.DAT	Tile 1 x 2 without Groove

Submodel: Rack

No.	Color	Part no.	Part name
1	Light-Gray	4073.DAT	Plate 1 x 1 Round
2	Light-Gray	3023.DAT	Plate 1 x 2
1	Light-Gray	3666.DAT	Plate 1 x 6
1	Light-Gray	3460.DAT	Plate 1 x 8
2	Light-Gray	3022.DAT	Plate 2 x 2
1	Black	3708.DAT	Technic Axle 12
1	Black	32064.DAT	Technic Brick 1 x 2 with Axlehole
2	Light-Gray	3713.DAT	Technic Bush
3	Light-Gray	3743.DAT	Technic Gear Rack 1 x 4

Submodel: Inertia Left

No.	Color	Part no.	Part name
3	Light-Gray	3023.DAT	Plate 1 x 2
2	Light-Gray	32028.DAT	Plate 1 x 2 with Door Rail
1	Light-Gray	3710.DAT	Plate 1 x 4
2	Light-Gray	3022.DAT	Plate 2 x 2
1	Black	3706.DAT	Technic Axle 6
3	Black	3700.DAT	Technic Brick 1 x 2 with Hole
1	Black	3701.DAT	Technic Brick 1 x 4 with Holes
1	Light-Gray	3713.DAT	Technic Bush
1	Light-Gray	3647.DAT	Technic Gear 8 Tooth
1	Black	6594.DAT	Tyre 49.6 x 28 VR
1	White	6595.DAT	Wheel 49.6 x 28 VR

Submodel: Inertia Right

No.	Color	Part no.	Part name
3	Light-Gray	3023.DAT	Plate 1 x 2
2	Light-Gray	32028.DAT	Plate 1 x 2 with Door Rail
1	Light-Gray	3710.DAT	Plate 1 x 4
2	Light-Gray	3022.DAT	Plate 2 x 2
1	Black	3706.DAT	Technic Axle 6
3	Black	3700.DAT	Technic Brick 1 x 2 with Hole
1	Black	3701.DAT	Technic Brick 1 x 4 with Holes

1	Light-Gray	3713.DAT	Technic Bush
1	Light-Gray	3647.DAT	Technic Gear 8 Tooth
1	Black	6594.DAT	Tyre 49.6 x 28 VR
1	White	6595.DAT	Wheel 49.6 x 28 VR

Total Assembly of the Watt Regulator

No.	Color	Part no.	Part name
2	Black	5306.DAT	Electric Brick 2 x 2 x 2/3 with Wire End
1	Blue	884.DAT	Electric Mindstorms RCX
6	Blue	32054.DAT	Technic Pin Long with Stop Bush
1		submodel	Base
1		submodel	Spindle
1		submodel	Tower
1		submodel	Linear Displacement Sensor
1		submodel	Rack
1		submodel	Inertia Left
1		submodel	Inertia Right

5 RCX Program

The following program for the RCX is written in NQC (Not Quite C, see http://bricxcc.sourceforge.net/nqc/). The program reads in the angle of the rotation sensor and stores it in the variable ANGLE. Subsequently, the motor power M is determined by the difference of a constant ANGLEBAR and ANGLE. The gain is KU/KL. The program has to use two constant (KU and KL) to specify the gain, because the RCX processor can only calculate with integers. Find two parameter settings for the gain KU/KL such that the Watt Governor has a stable equilibrium for one setting and a limit cycle (oscillatory motion) for another setting and store these programs in two slots of the RCX.

```
// *******************************************
// Wattgovernor.nqc
// Remco Leine, September 2004
// *******************************************


// sensors
#define ANGLE SENSOR_1

// motors
#define MOTOR OUT_A

// other constants
#define KU 4
#define KL 1
#define ANGLEBAR 15

int M;

task main() {
  // configure sensor
  SetSensor(ANGLE, SENSOR_ROTATION);
  ClearSensor(ANGLE);
  SetPower(MOTOR,0);
  OnFwd(MOTOR);
  while(true)
  {
   M = (KU*(ANGLEBAR - abs(ANGLE)))/KL;
   if (M<0) M=0;
   if (M>7) M=7;
   SetPower(MOTOR,M);
  }
}
```