

```
totalhalt : first stop!  
  
one_way :  
turn : look to one side  
servo_turn : wait for the servo to be finished turning  
g 0, 0!  
totalhalt
```

```
the other way:  
turn : look to another side  
servo_turn : wait for the servo to be finished turning  
g 0, 0!  
totalhalt
```

```
so which is the better way?  
a2 : yes!  
b2 : turn  
  
b2 : return
```

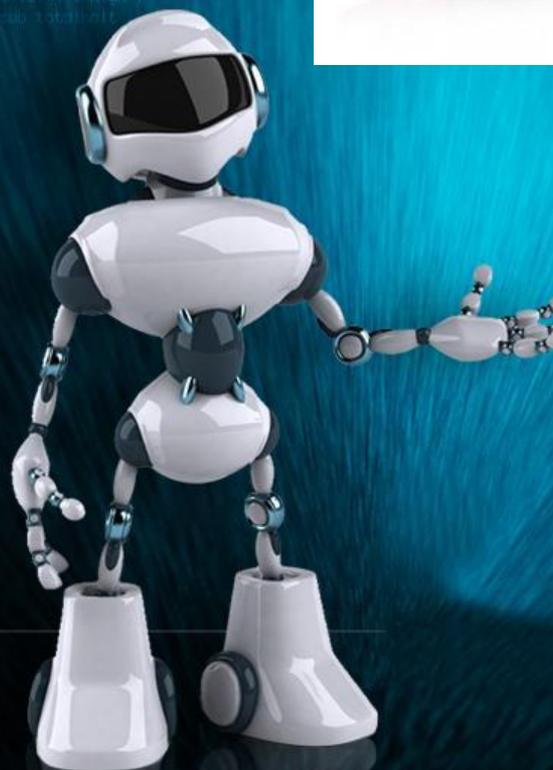
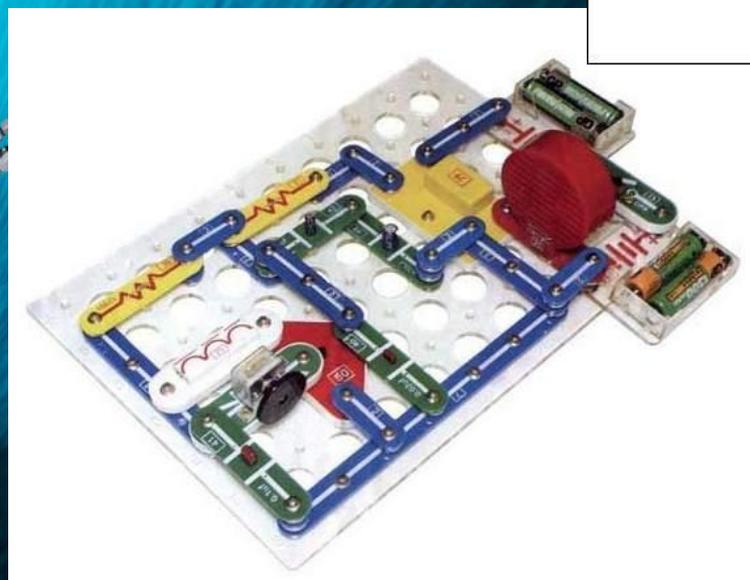
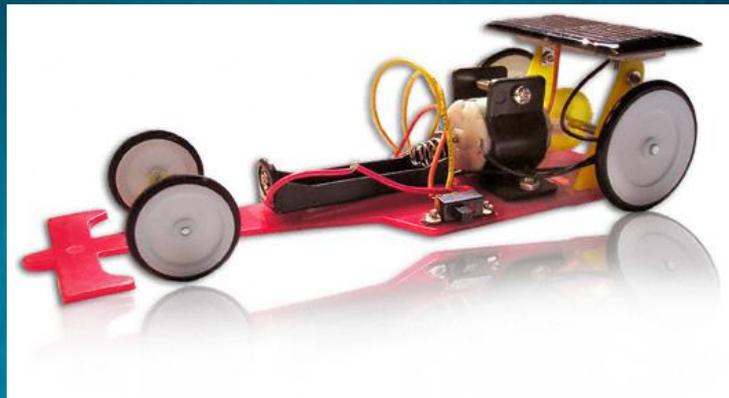
```
turns  
g low 5 : low 7 : high 4  
turn : servo totalhalt
```

```
light  
g low 0 : low 2 : high 7  
turn : servo totalhalt
```



Перворобот

Электронный конструктор



totalhalt : first stop
back :
turn : look to one side
servo : turn & wait for the servo to be finished turning
& go on
totalhalt

the other way :
turn : look to another side
servo : turn & wait for the servo to be finished turning
& go on
totalhalt

on which is the better way :
on : turn
back : turn

back : turn

turn :
- low 5 : low 7 : high 4
turn : servo : totalhalt

high :
- low 6 : low 2 : high 7
turn : servo : totalhalt

```
totalhalt = first_stop()
brn_servo = look_to_one_side()
servo_turn = wait_for_the_servo_to_be_finished_turning()
totalhalt
```

```
the other way:
turn = look_to_another_side()
servo_turn = wait_for_the_servo_to_be_finished_turning()
totalhalt
```

```
on which is the better way:
on_servo
totalhalt
```

```
turns
low 5 : low 7 : high 4
turn = servo_totalhalt
```

```
turns
low 6 : low 7 : high 7
turn = servo_totalhalt
```

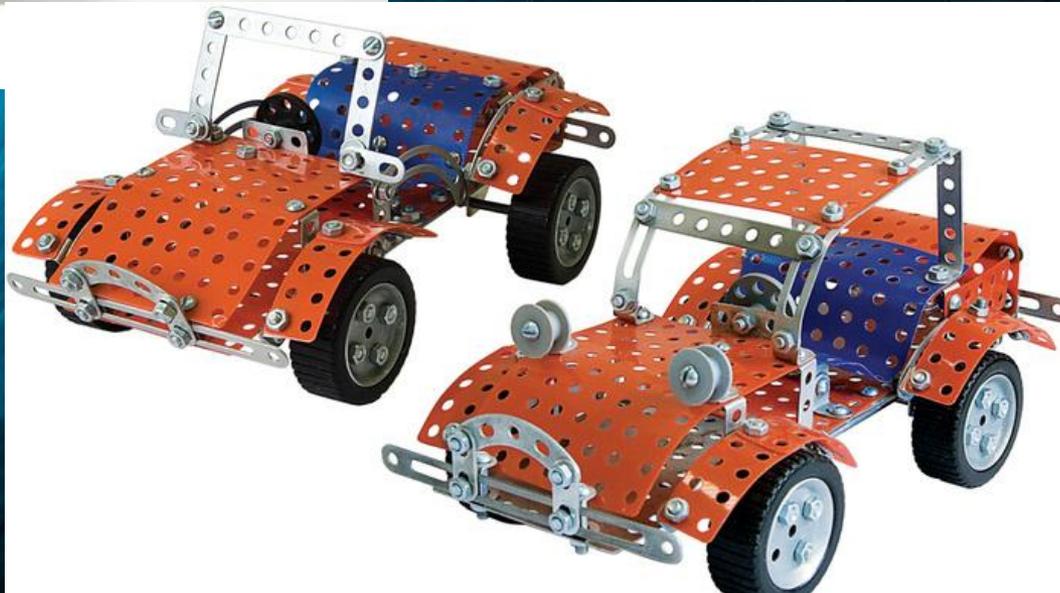
Пластмассовый конструктор



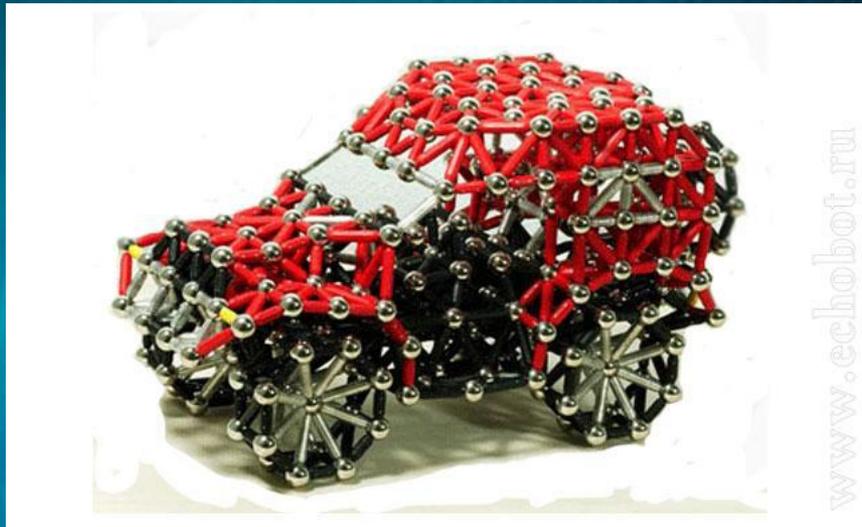
Металлический конструктор



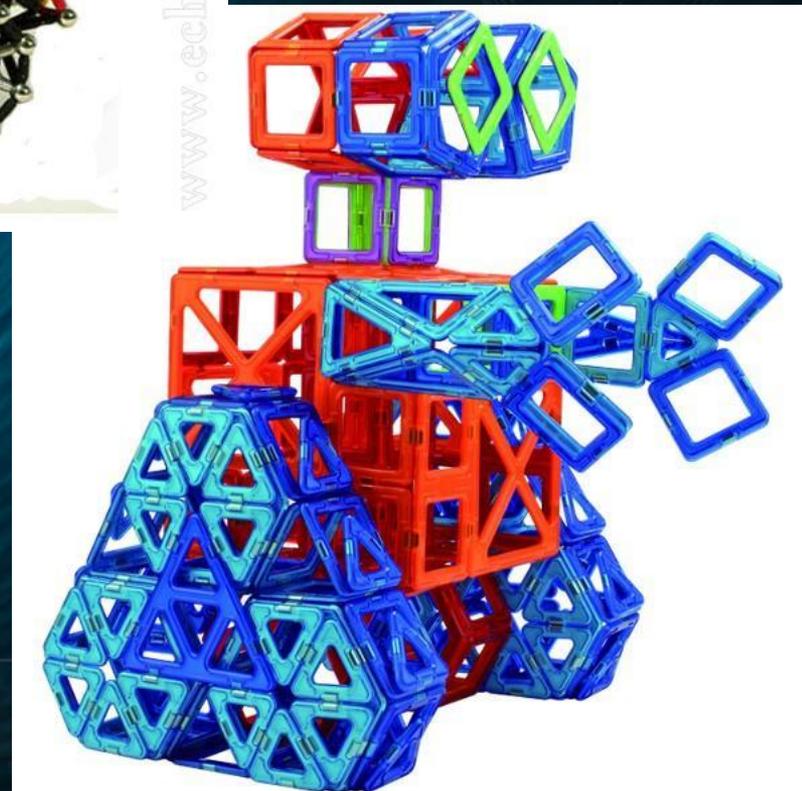
www.karusel-toys.ru



Магнитный конструктор



www.echobot.ru



totalhalt = totalhalt + 1

```
low = 0
high = 1
servo.turn(90)
wait(1000)
totalhalt = totalhalt + 1
```

```
the other way:
return = 180
servo.turn(90)
wait(1000)
totalhalt = totalhalt + 1
```

is which is the better way?
no, then
servo.turn(180)

totalhalt = totalhalt + 1

```
turns = 0
low = 0
high = 1
turn(90)
totalhalt = totalhalt + 1
```

```
high = 180
low = 0
turn(90)
totalhalt = totalhalt + 1
```

Деревянный конструктор

totalHeight = first.steel

base.moveTo

turn() look to one side

servo.turn() wait for the servo to be finished turning

g.moveTo

totalHeight

the other way:

turn() look to another side

servo.turn() wait for the servo to be finished turning

g.moveTo

totalHeight

as which is the better way:

g.moveTo

base.turn

base.moveTo

turns

low 5 : low 7 : high 4

turn() servo totalHeight

turns

low 6 : low 7 : high 7

turn() servo totalHeight



Конструктор из бумаги «Умная бумага»



Конструкторы из Бельгии

Кликс



totalWait: 1000; stop();
oneWay:
turn: look to one side
servoTurn: wait for the
servo to be finished
totalWait;
the other way:
turn: look to another side
servoTurn: wait for the servo to be finished turning
servo; stop();
totalWait;
on which is the better way:
on; turn;
servoTurn;
totalWait;
turns:
turn: low 5; low 7; high 4
turn; servo; totalWait;
light:
turn: low 8; low 9; high 7
turn; servo; totalWait;

Конструктор Lego



totalhalt : first stop!
back : wait!
turn : look to one side
servo.turn : wait for the servo to be finished turning
a : 90
totalhalt!

the other way!
turn : look to another side
servo.turn : wait for the servo to be finished turning
- 90
totalhalt!

as which is the better way?
a2 : turn
back : turn

back : turn

turns
- : low 5 : low 7 : high 4
turn : servo : totalhalt!

turns
- : low 6 : low 4 : high 7
turn : servo : totalhalt!

Спасение самолета

```
totalhalt : first stop!  
  
back : wait  
turn : look to one side  
servo.turn : wait for the servo to be finished turning  
g : 90 deg  
totalhalt
```

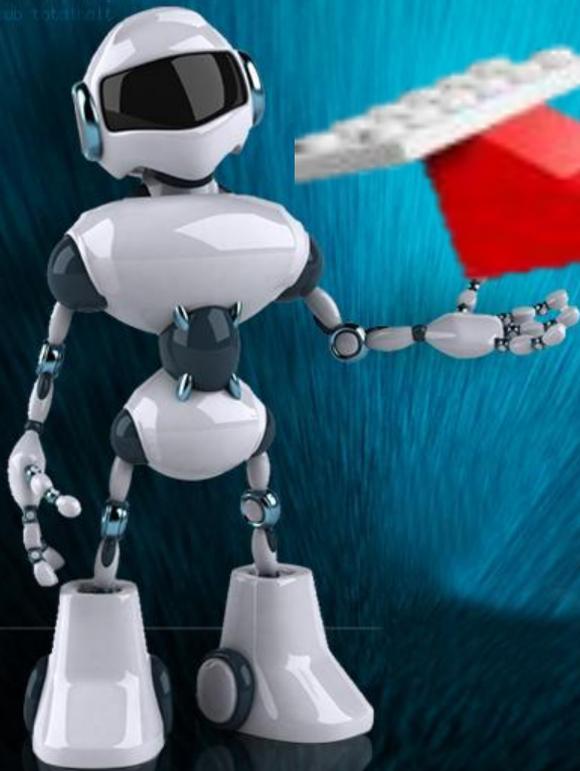
```
the other way:  
turn : look to another side  
servo.turn : wait for the servo to be finished turning  
g : 90 deg  
totalhalt
```

```
so which is the better way?  
a2 : yes!  
back.turn
```

```
back.turn
```

```
turns  
g : low 5 : low 7 : high 4  
turn : servo totalhalt
```

```
light  
g : low 6 : low 4 : high 7  
turn : servo totalhalt
```



Голодный аллигатор



Рычащий лев



Порхающая птица



```
bot.spin(1) # first spin
bot.stop()
bot.turn(90) # look to one side
servo.turn(1) # wait for the servo to be finished turning
bot.spin(1)
bot.stop()
```

```
# the other way:
bot.turn(270) # look to another side
servo.turn(1) # wait for the servo to be finished turning
bot.spin(1)
bot.stop()
```

```
def which_is_the_bot():
    bot.spin(1)
    bot.stop()
```

```
def turn_bot():
    bot.turn(90)
    bot.spin(1)
    bot.stop()
```

```
def turn_bot():
    bot.turn(270)
    bot.spin(1)
    bot.stop()
```

Вратарь



Танцующие ПТИЦЫ

totalHit : 0 ; first : stop ;
bird : left ;
turn : look to one side ;
servo : turn ; wait for the servo to be finished turning ;
g : 0 ;
totalHit ;

the other way :
turn : look to another side ;
servo : turn ; wait for the servo to be finished turning ;
g : 0 ;
totalHit ;

do which is the better way :
do : turn ;
servo : turn ;

end : return ;

turn :
- low 5 ; low 7 ; high 4 ;
turn : servo : totalHit ;

left :
- low 6 ; low 7 ; high 7 ;
turn : servo : totalHit ;



Обезьянка-барабанщица

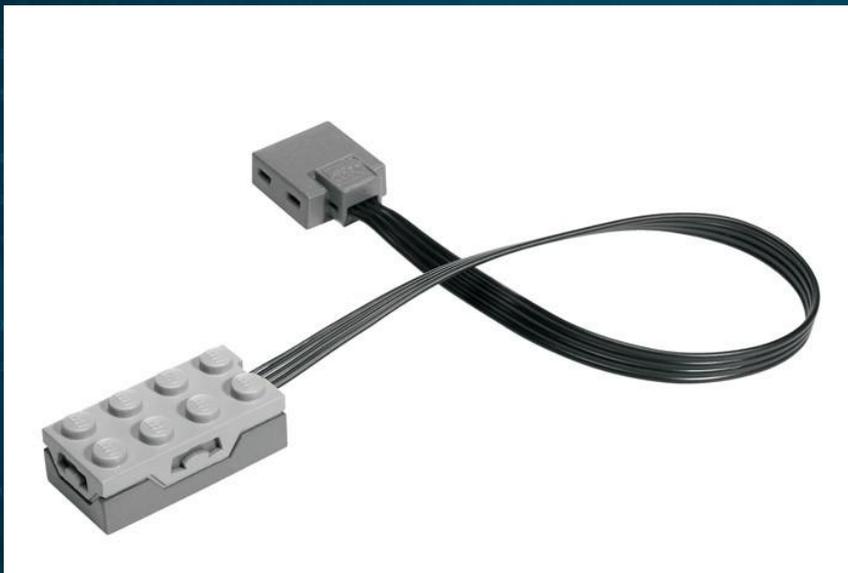


Нападающий



Конструктор Lego Wedo





Датчик наклона



USB коммутатор



Датчик расстояния



Мотор



Конструктор NXT Mindstorm



```
totalhdlt : first stop!
```

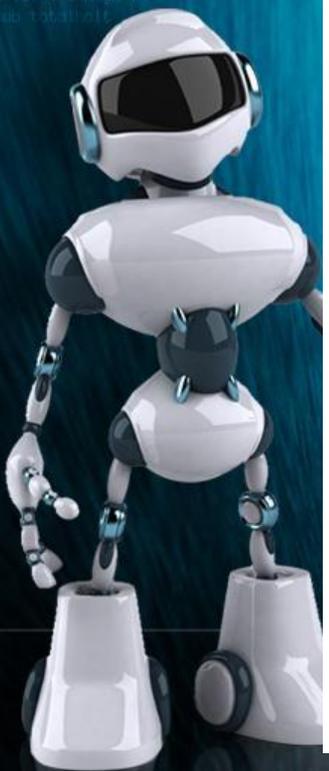
```
brk : stop!  
turn : look to one side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

```
the other way!  
turn : look to another side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

```
ok which is the better way?  
ok then!  
back to turn  
back to turn
```

```
turns  
g : low 5 : low 7 : high 4  
turn : servo totalhdlt
```

```
light  
g : low 6 : low 4 : high 7  
turn : servo totalhdlt
```



```
totalhdlt : first stool
```

```
brk : hvt  
turn : look to one side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

```
the other way:  
turn : look to another side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

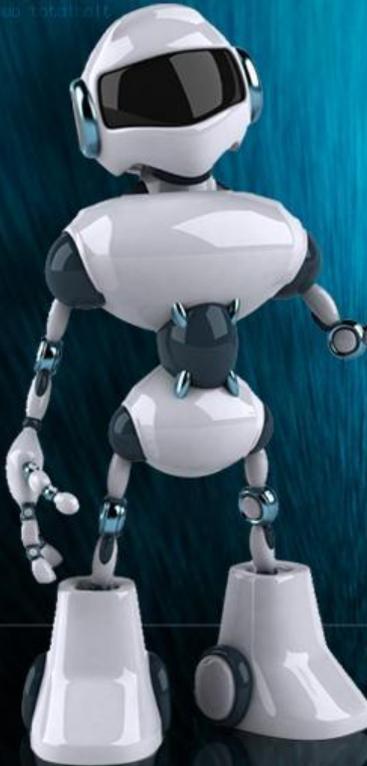
```
de : which is the better way:
```

```
ac : hvt  
brk : turn
```

```
brk : turn
```

```
turn :  
g : low 5 : low 7 : high 4  
turn : servo totalhdlt
```

```
turn :  
g : low 6 : low 7 : high 7  
turn : servo totalhdlt
```



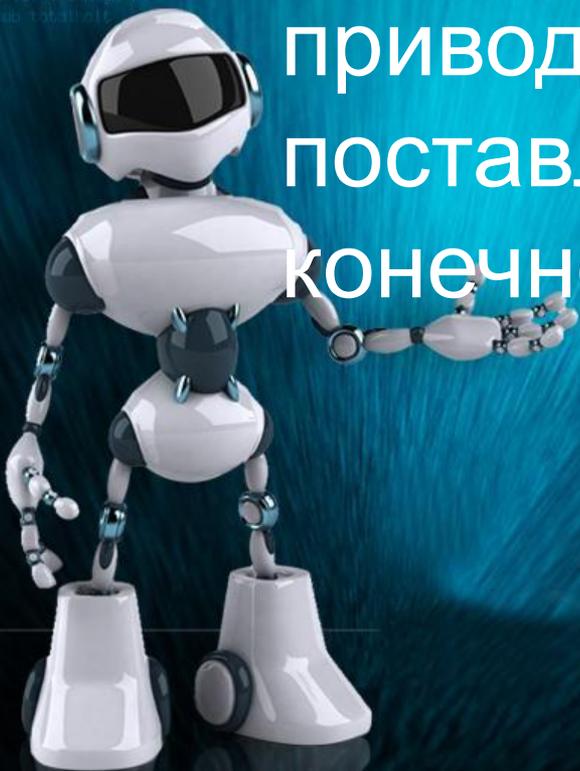
```
totalhalt = first_stop()
loop:
  turn: look to one side
  servo_turn: wait for the servo to be finished turning
  g 0, 0
totalhalt
```

```
the other way:
turn: look to another side
servo_turn: wait for the servo to be finished turning
  g 0, 0
totalhalt
```

```
as which is the better way:
do: turn
servo_turn
end: return
```

```
turns:
  low 5 : low 7 : high 4
turn: servo totalhalt
```

```
turn:
  low 6 : low 4 : high 7
turn: servo totalhalt
```



Алгоритм

- это описание последовательности действий, строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.