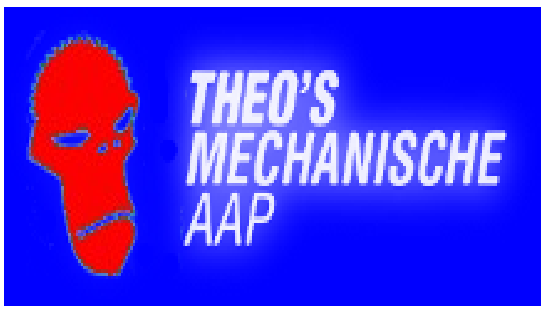
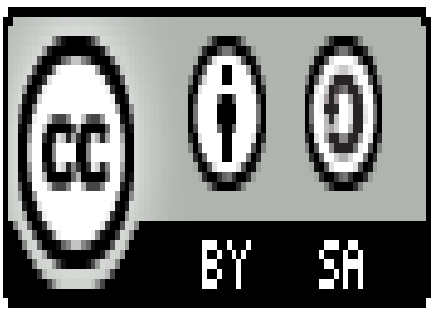
TMA Robot mobility module

Author: Rein Velt

Version: 6

Datum: 14. Dec. 2009





TMA robot mobility module

by Rein Velt

Published by Theo's Mechanic Ape http://mechanicape.com

Version history

Date	Version	Author	Description
20091214	1	Rein Velt	Initial version

For more information about this project contact theo@mechanicape.com or visit our website at http://mechanicape.com..

This document is dedicated to …..

Table of Contents

Introduction

This document describes the design of the TMA Robot mobility module. This module makes it possible to move from point a to point b while carrying a payload of 5 – 20 kilograms.

Functional demands

We want a moving robot. It should be able to move, avoid collisions and it should have a lot of cool and hi tech features to impress people.

What features are required for our module?

1. Move (forward, backward)
2. Turn (left, right)
3. Carry a payload weight varying from 5 – 20 kilograms.
4. Avoid collisions
5. Receive commands from a host
6. Send process- and sensordata to a host

What features are nice to have?

- On-board camera
- Object recognition
- Face tracking
- Speech Recognition
- Voice synthesizer

We decided to split the project in two parts, a mobility module and an additional module with all the cool (processor intensive) features. This makes it possible to re-use both modules separately for other robots with different purposes. So we focus at the movement basics for now and continue to the list of features.

Feature list

1. Move
    1. The device must be able to move forward and backward.
    2. The speed must be controllable
2. Turn
    1. The device must be uble to turn left and right
    2. The rotation should be a absolute compass direction
3. Carry
    1. The device must be able to carry a payload from maximum 20 kilograms
    2. The device must be able to supply the payload module with a powerline (12V) and a serial connection (rs232)
4. Communication
    1. Receive commands from a host (via rs232)
    2. Send process- and sensordata to a host (via rs232)
5. Security
    1. Watchdog for failsafe operation

Problem analysis

Move

The requirements say:

- 1.1 - The device must be able to move forward and backward.
- 1.2 - The speed must be controllable

We can translate this intro:

- **A.- The device needs one or more motors**
- **B - The device needs a power source**
- **C - The device needs a PWM motor controller (speed + direction)**

Turn

The requirements say:

- 2.1 - The device must be uble to turn left and right
- 2.2 - The rotation should be a absolute compass direction

We can translate this into:

- **D. - The left- and right wheels should be driven independent (like a tank, because tanks look cool)**
- **E. - The device should have an internal compass**

Carry

The requirements say:

- 3.1 - The device must be able to carry a payload from maximum 20 kilograms
- 3.2 - The device must be able to supply the payload module with a powerline (12V) and a serial connection (rs232)

We can translate this into:

- **F. - We need hi-torque motors and a decent transmission to carry the load**
- **G. - We cannot drive the wheels directly. We need separate axles for the wheels**
- **H. - We don't want to overload the motors, so we need a belt-drive**
- **I. - We need to provide some nice and decent connectors for the 12Volt and rs232 lines.**

Communication

The requirements say:

- 4.1 - Receive commands from a host (via rs232)
- 4.2 - Send process- and sensordata to a host (via rs232)

All demand have been covered by **point I**

Security

The requirements say:

- 5.1 Watchdog for failsafe operation

We can translate this into:

- **J. We need an internal watchdog timer**
- **K. The device should be disabled when the watchdog gives a timeout**

Additional

We need a microcontroller to bind all the pieces together. The microntroller must be able to do the following tasks:

- Send signal to the PWN motor driver
- Receive data from the internal compass
- Implement a basic collision detection system
- Receive serial data from host
- Send serial data to host
- Implement a Watchdog timer failsafe feature

We have already experience with the Arduino, a low-cost open source Atmle AVR based board. The device costs 15 euros and is perfectly fit for our application. This leads to the final demand:

- **L. - We need an Arduino microcontroller board to bind all electronic pieces together.**
- **M. - We need a Chassis to bind all hardware pieces together**

Global technical design

We can group our technical demand based on functionality. The result is below.

Chassis

1. We need a Chassis to bind all hardware together
2. We need to provide some nice and decent connectors for the 12Volt and rs232 lines.

Transmission

3. The device needs one or more motors
4. The device needs a power source
5. The device needs a PWM motor controller (speed + direction)
6. The left- and right wheels should be driven independent (like a tank, because tanks look cool)
7. We need hi-torque motors and a decent transmission to carry the load
8. We cannot drive the wheels directly. We need separate axles for the wheels
9. We don't want to overload the motors, so we need a belt-drive

Controller

10. The device should have an internal compass
11. We need an Arduino microcontroller board to bind all electronic pieces together.
12. We need an internal watchdog timer
13. The device should be disabled when the watchdog gives a timeout

Diagram 1 - Top view

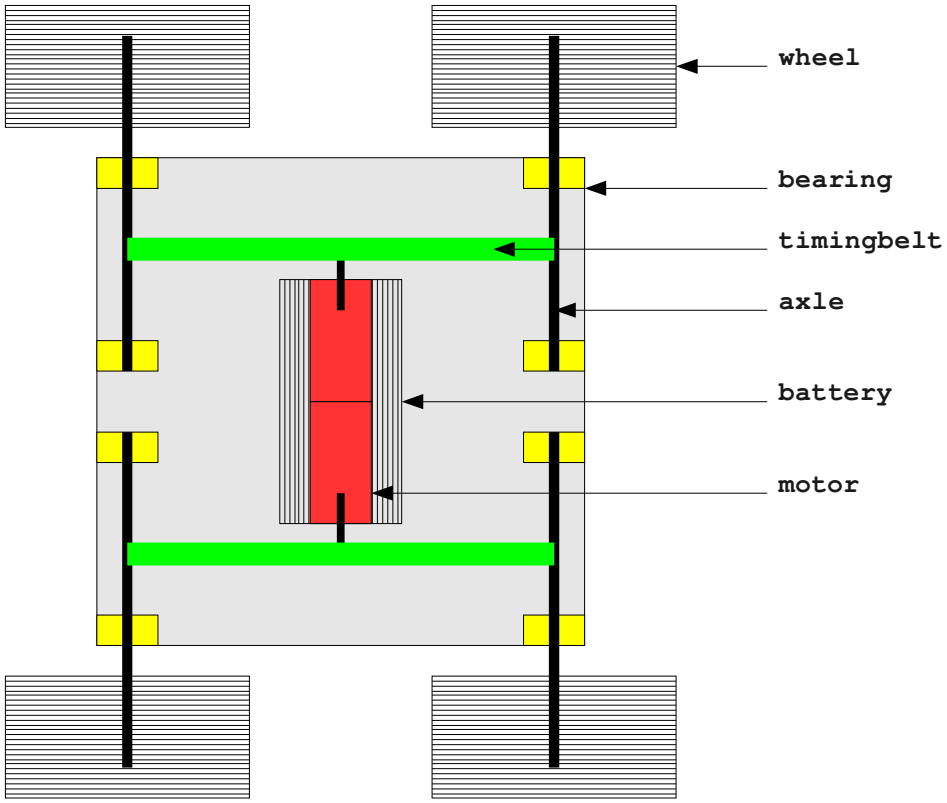


Diagram 2 - Side view

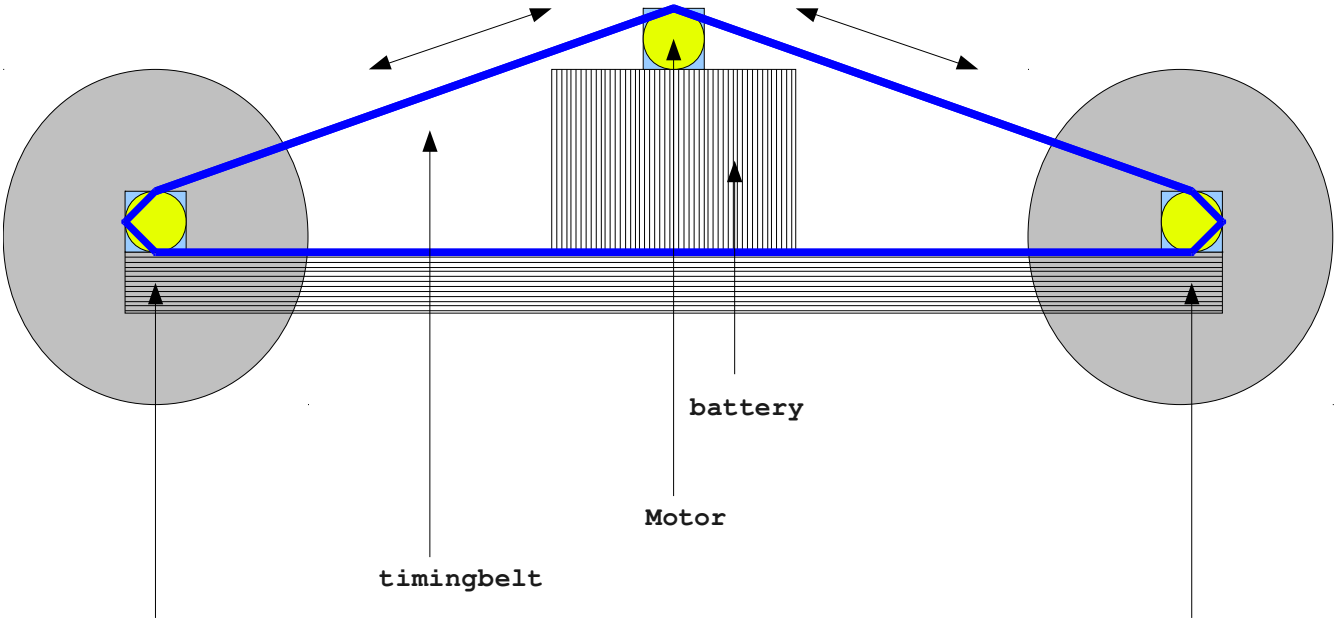


Diagram 3 – 3D view (artist impression)

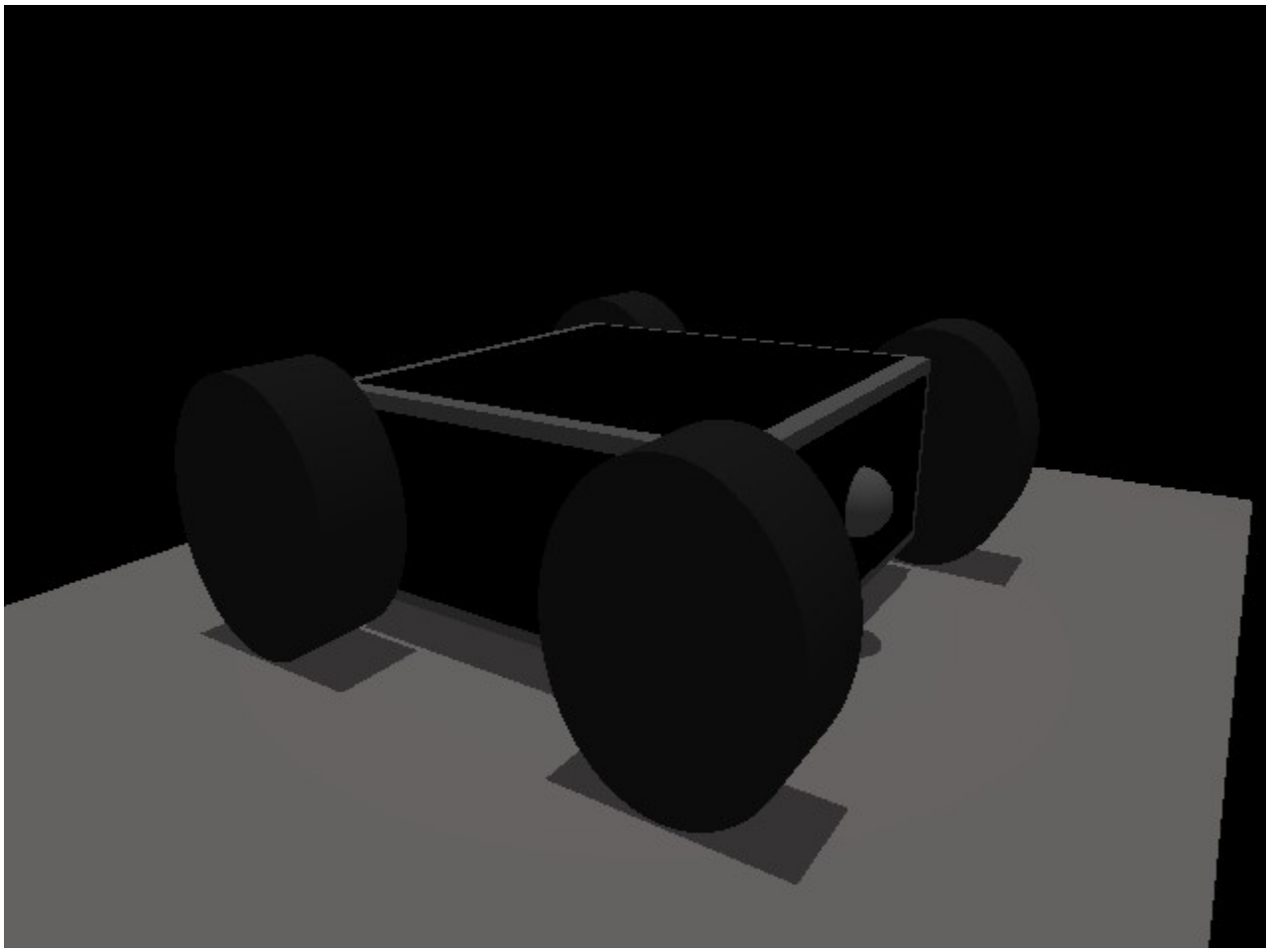


Diagram 4 – 3D side view (artist impression)

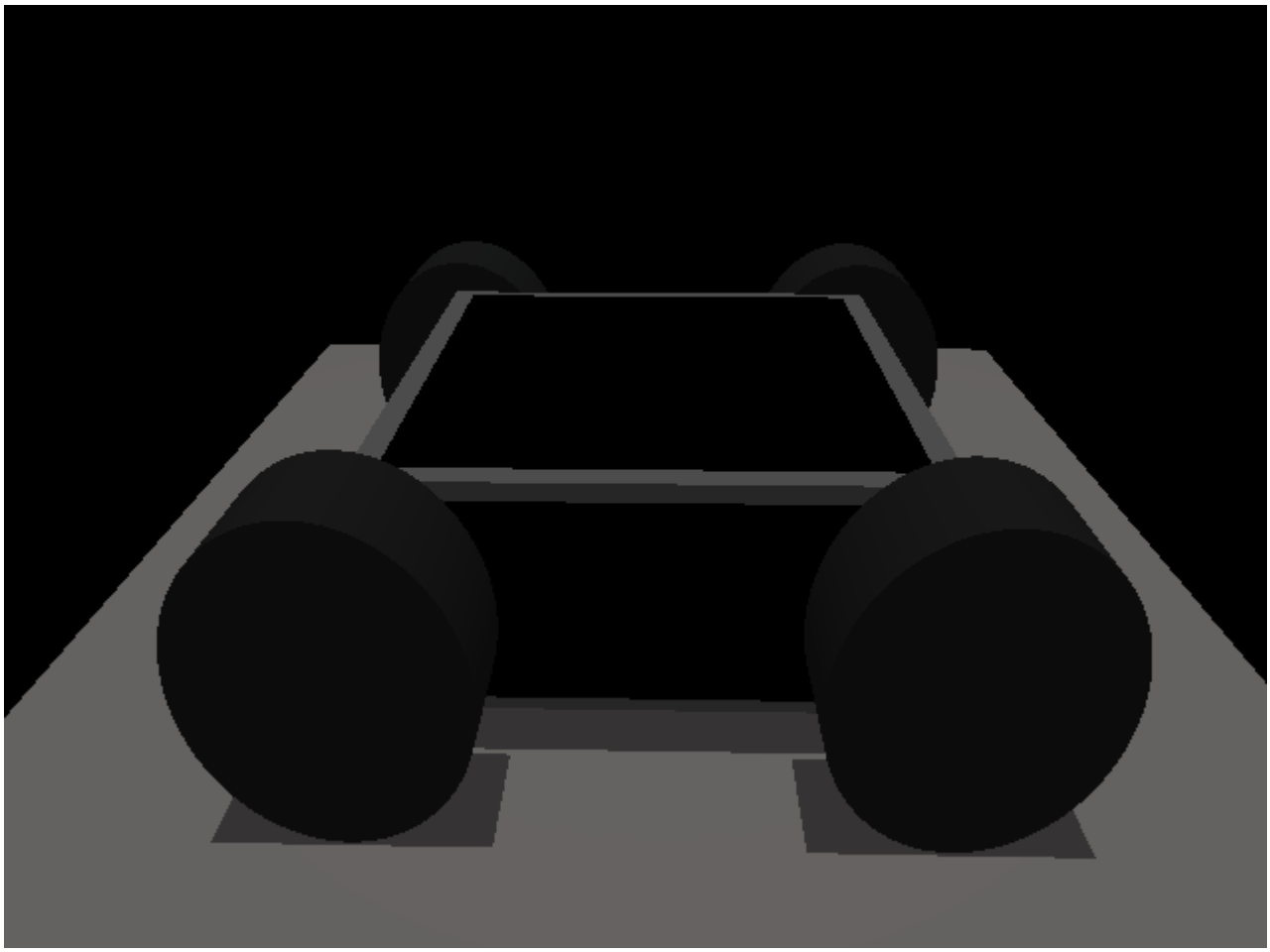
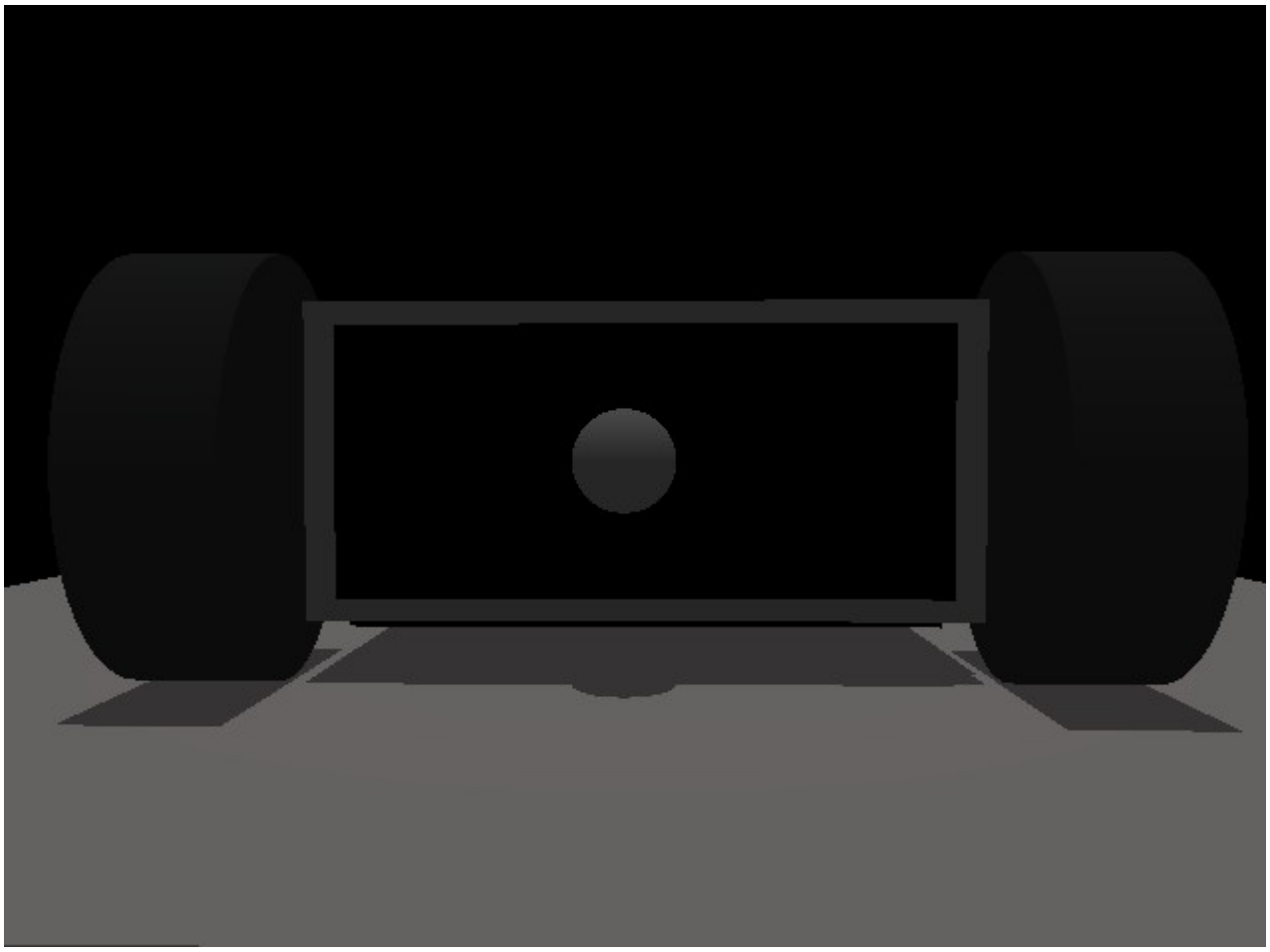


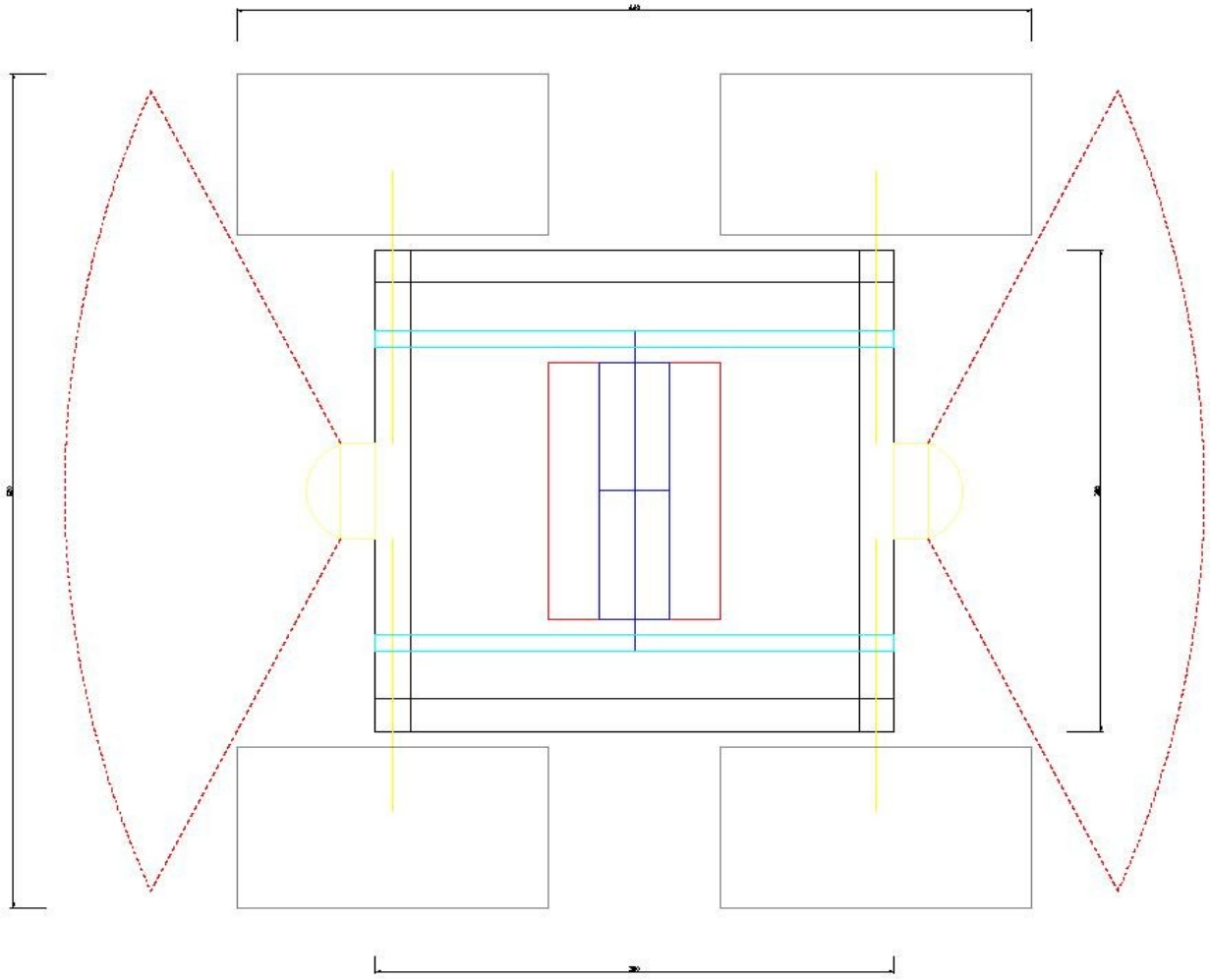
Diagram 5 – 3D front view (artist impression)



Detailed technical description

The most mechanical parts used in this design can be order at your local DIY-shop (Gamma, Hubo, ...). The motors, timingbelt, wheels, batteries and sensors can be obtained at the Conrad webshop at http://www.conrad.nl. The Arduino and the Ardumoto shield can be obtained from Sparkfun at http://www.sparkfun.com.

Chassis



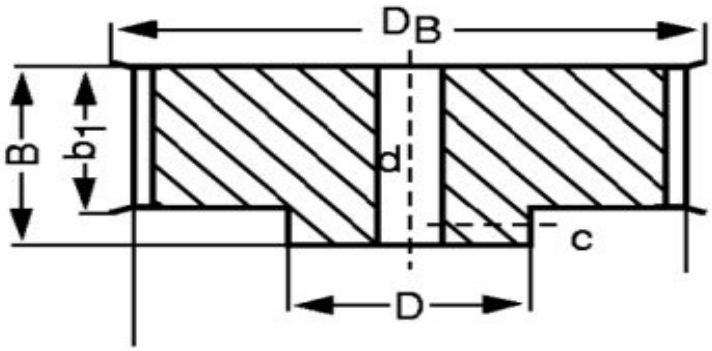
Transmission

The transmission must be able to carry a weight of maximum 20 kilograms plus the weight of the vehicle itself. All mechanical components should be strong enough to carry this weight.

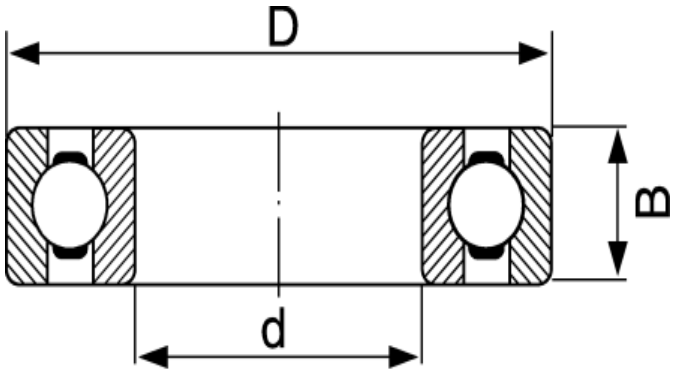
Motors

Timingbelt

22 61 06

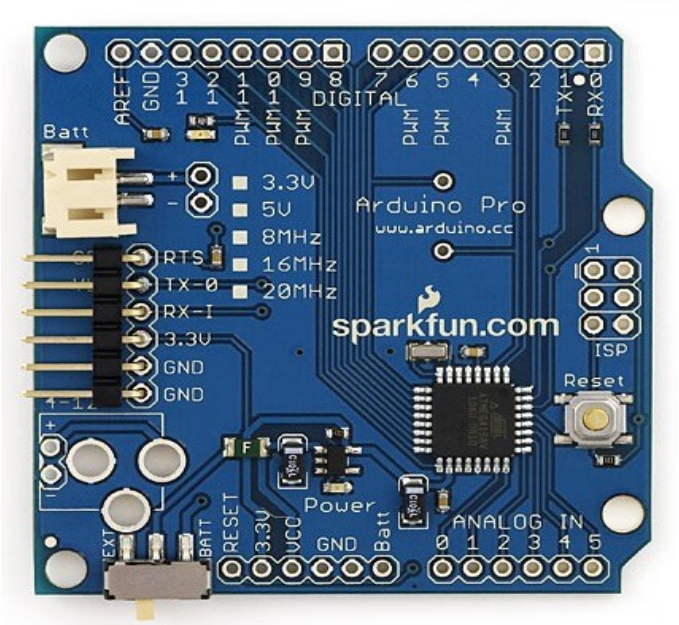


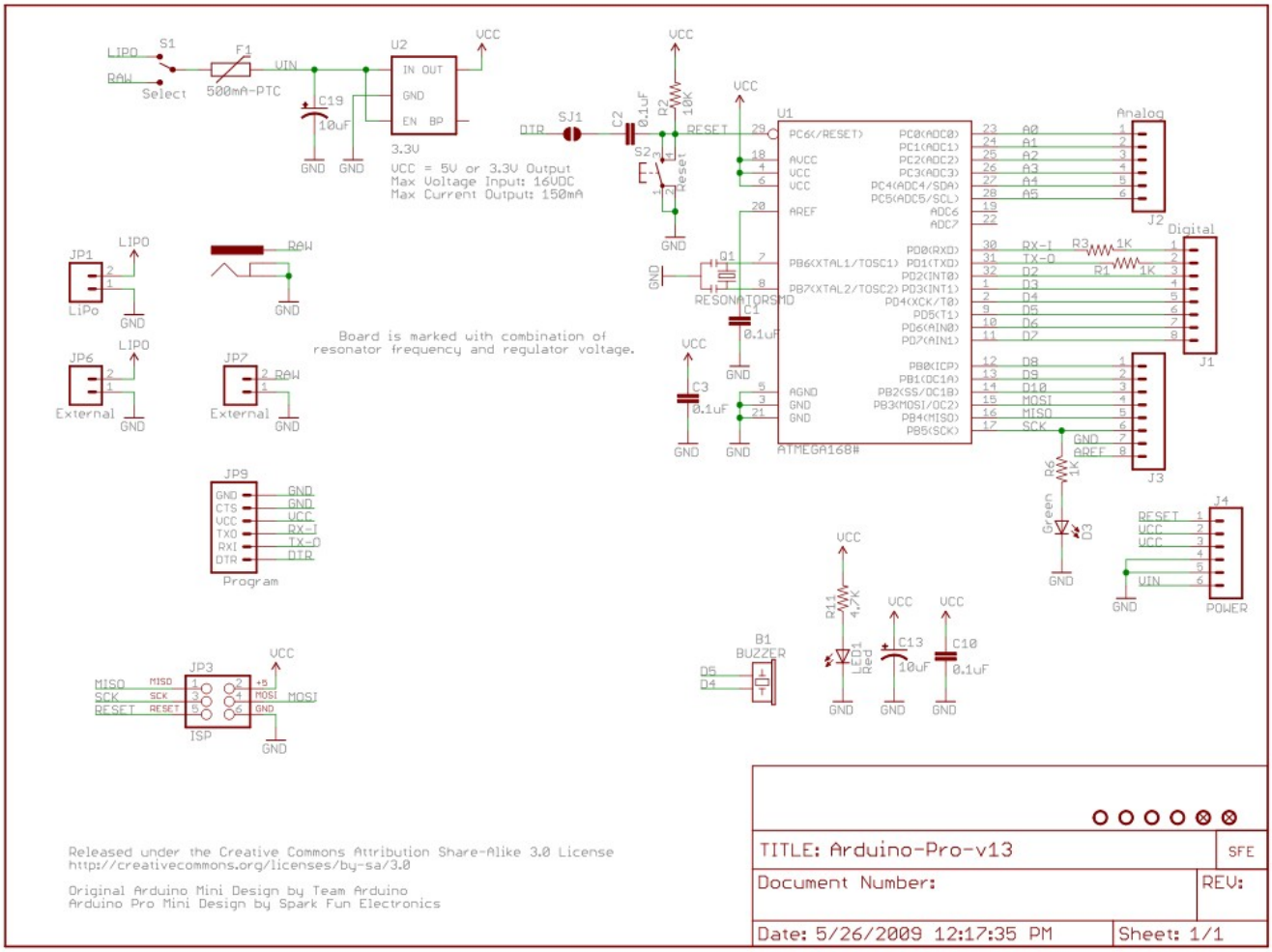
Axle & bearings



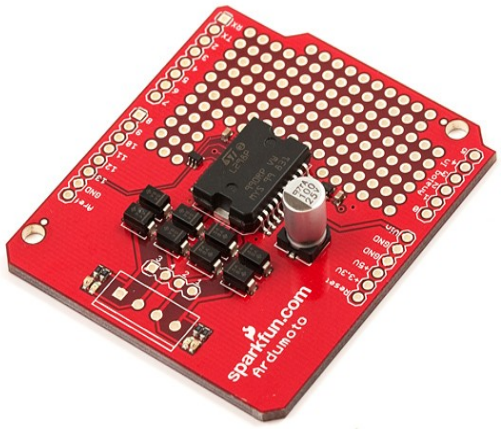
Wheels

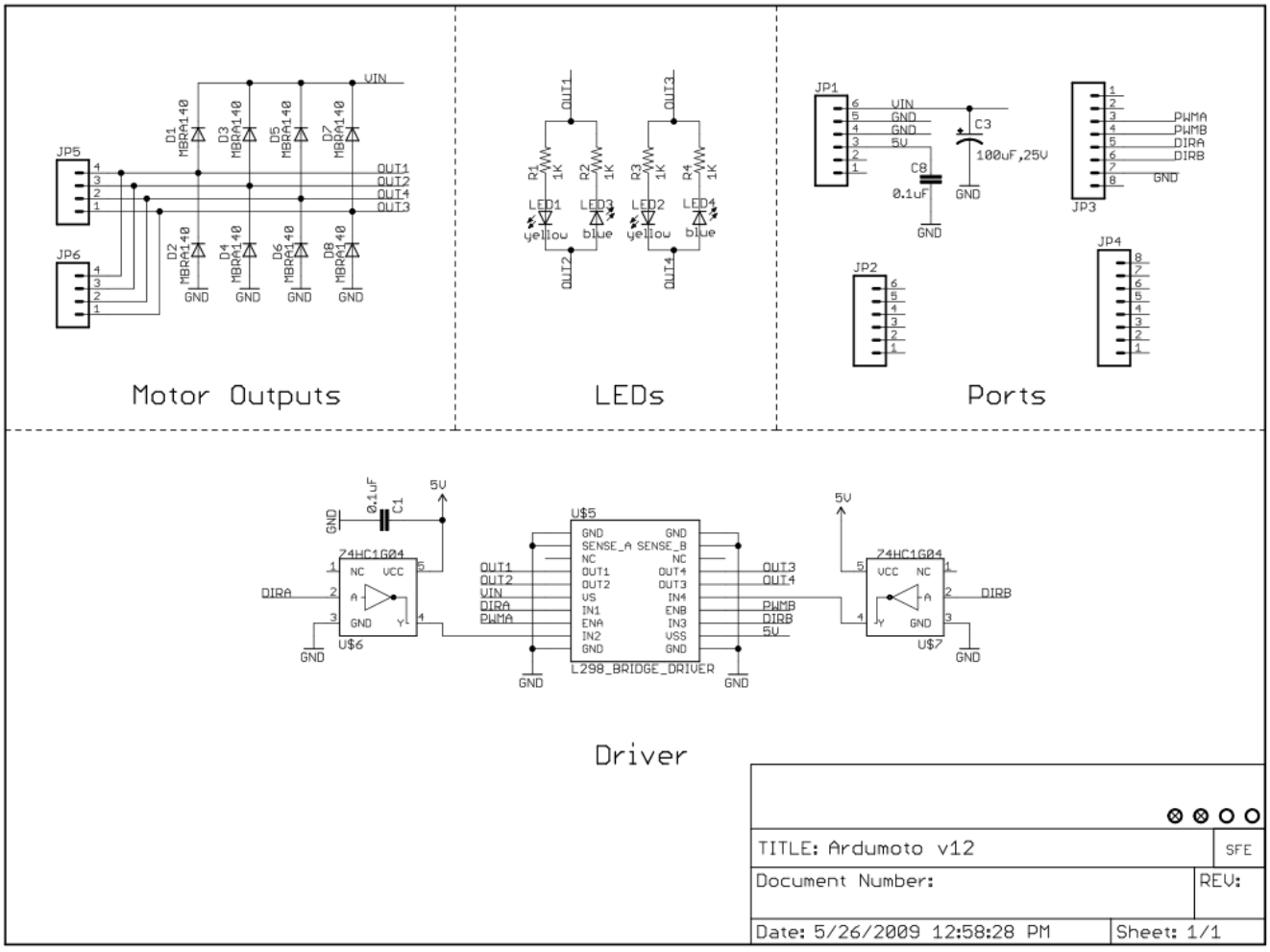
Controller


Batt
AREF
GND
DIGITAL
PWM
TX-1
RX-0
3.3V
5V
8MHz
16MHz
20MHz
RTS
TX-0
RX-I
3.3V
GND
GND
Arduino Pro
www.arduino.cc
sparkfun.com
ISP
Reset
Power
ANALOG IN
RESET
3.3V
VCC
GND
Batt
EXT
BATT


LIPO
RAW
Select
S1
F1
500mA-PTC
VIN
U2
IN OUT
GND
EN BP
3.3V
C19
10uF
VCC = 5V or 3.3V Output
Max Voltage Input: 16VDC
Max Current Output: 150mA
SJ1
C2
0.1uF
R2
10K
RESET
S2
Reset
U1
ATMEGA168#
RESONATORSMD
Q1
C1
0.1uF
C3
0.1uF
JP1
LiPo
JP6
External
JP7
External
Board is marked with combination of
resonator frequency and regulator voltage.
JP9
Program
JP3
ISP
Analog
J2
Digital
J1
J3
J4
POWER
R3
1K
R1
1K
R6
1K
Green
D3
R11
4.7K
Red
C13
10uF
C10
0.1uF
B1
BUZZER
Released under the Creative Commons Attribution Share-Alike 3.0 License
http://creativecommons.org/licenses/by-sa/3.0
Original Arduino Mini Design by Team Arduino
Arduino Pro Mini Design by Spark Fun Electronics
TITLE: Arduino-Pro-v13
SFE
Document Number:
REV:
Date: 5/26/2009 12:17:35 PM
Sheet: 1/1

PWM controller





Sensor firmware

Communication firmware

Communication protocol specification

Software

Firmware

Host software

Alphabetical Index