

# Soirée Pratique

# Build your own robot

# Motor control

**Part 1:**  
**Introduction**  
**and assembling the standard platform**

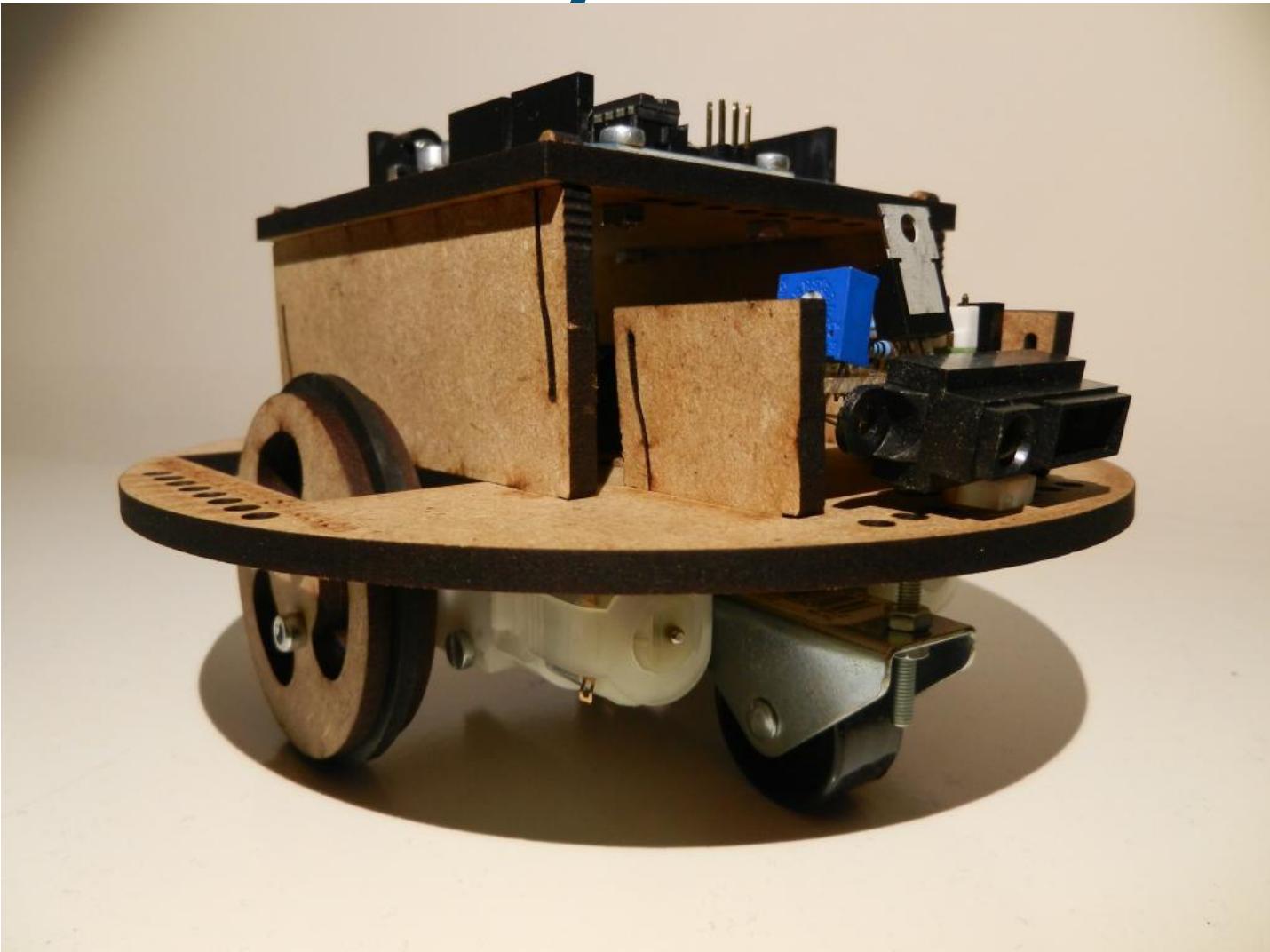
# Roadmap SP 2012-2013 (sem 1)

1. The brains: Arduino
2. The muscles: motor and power (today)
3. The eyes: sensors (**5/11**)
4. More brains: programming  
(19/11, **20h**)
5. Sumo Competition (26/11)

# All info online!

- <http://www.ieee-sb-leuven.be/node/108>
- Motor spec:  
<http://www.pololu.com/catalog/product/1120>
- Controller spec:  
<http://www.pololu.com/catalog/product/2135>

# Frame assembly



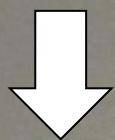




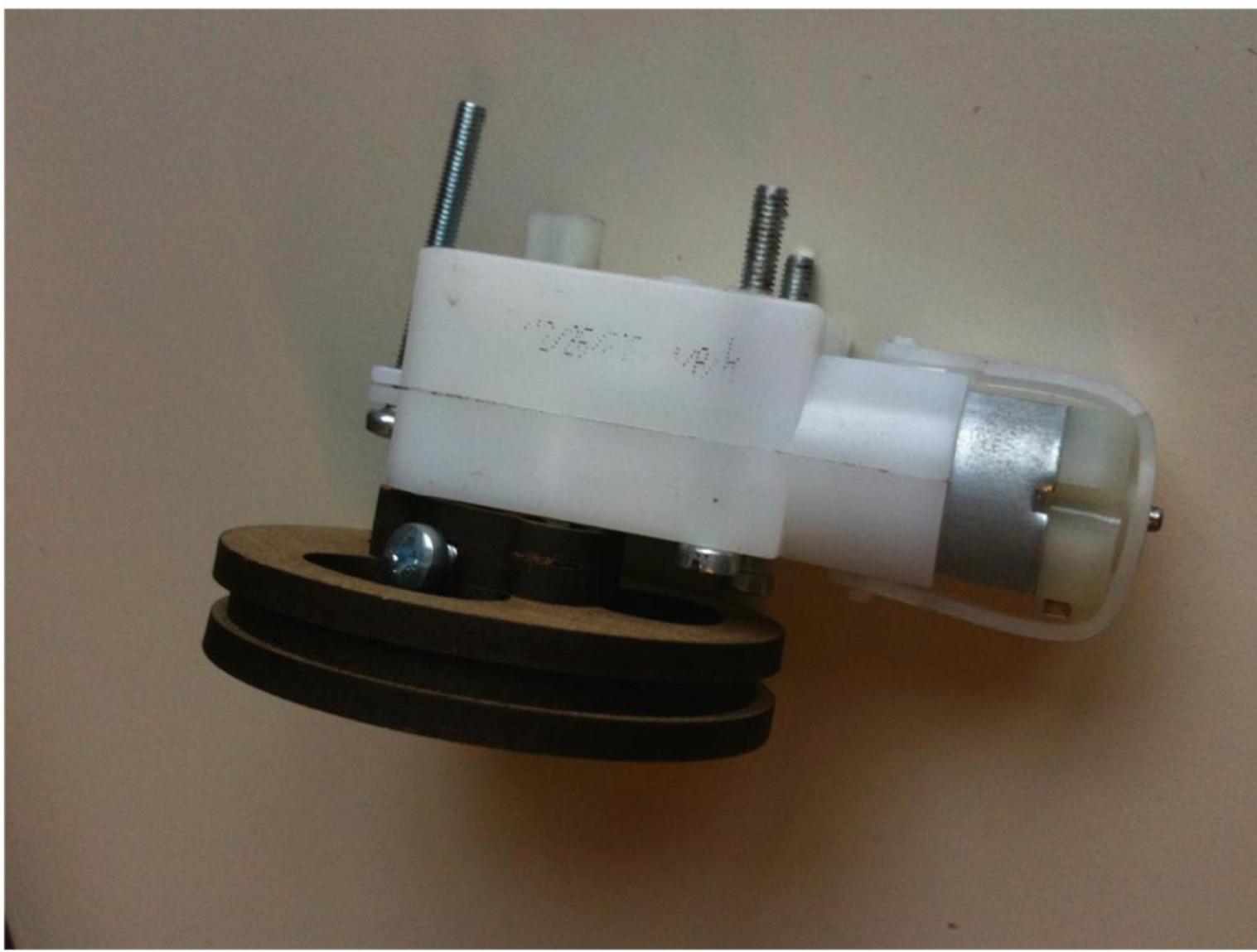


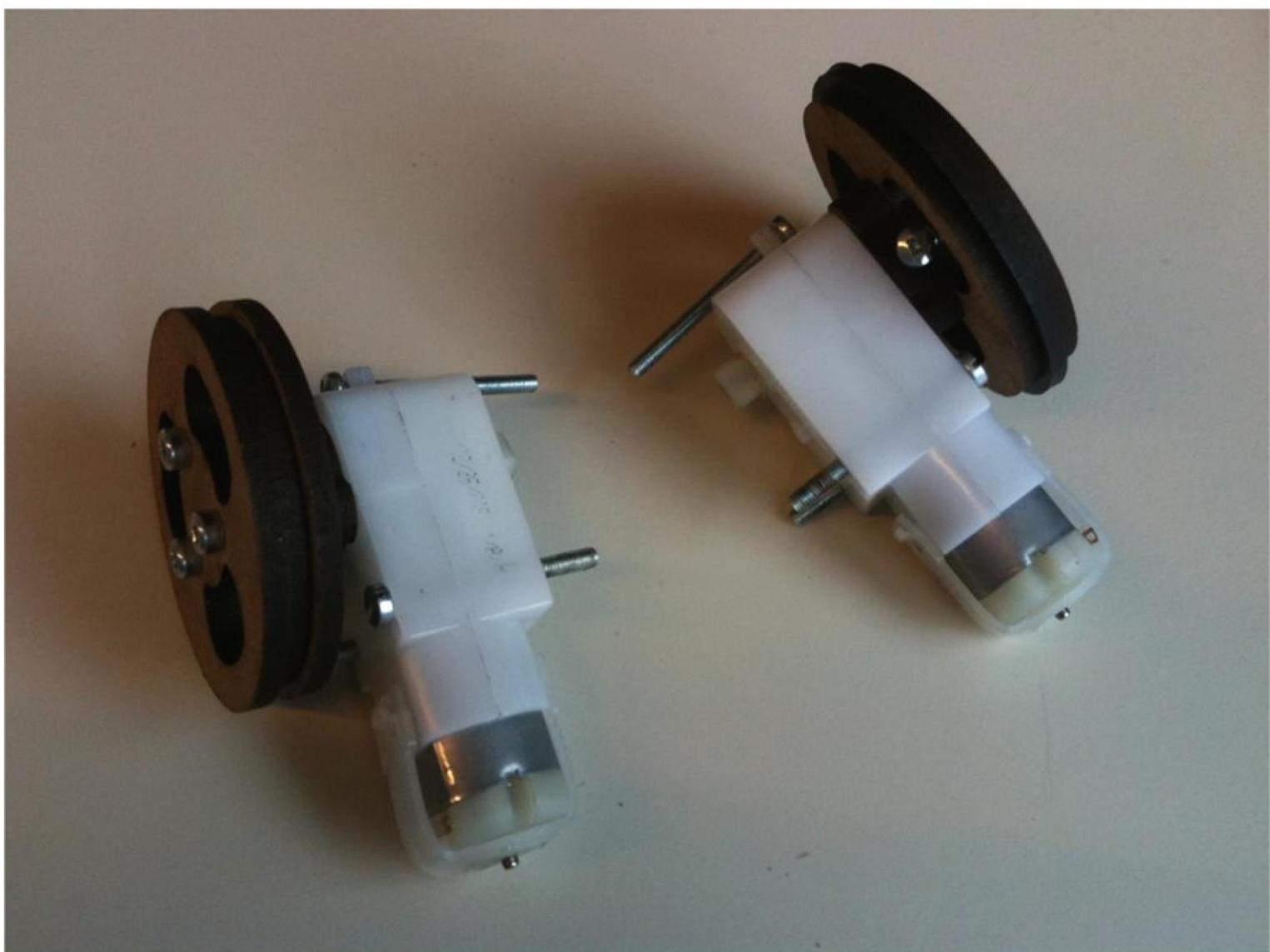


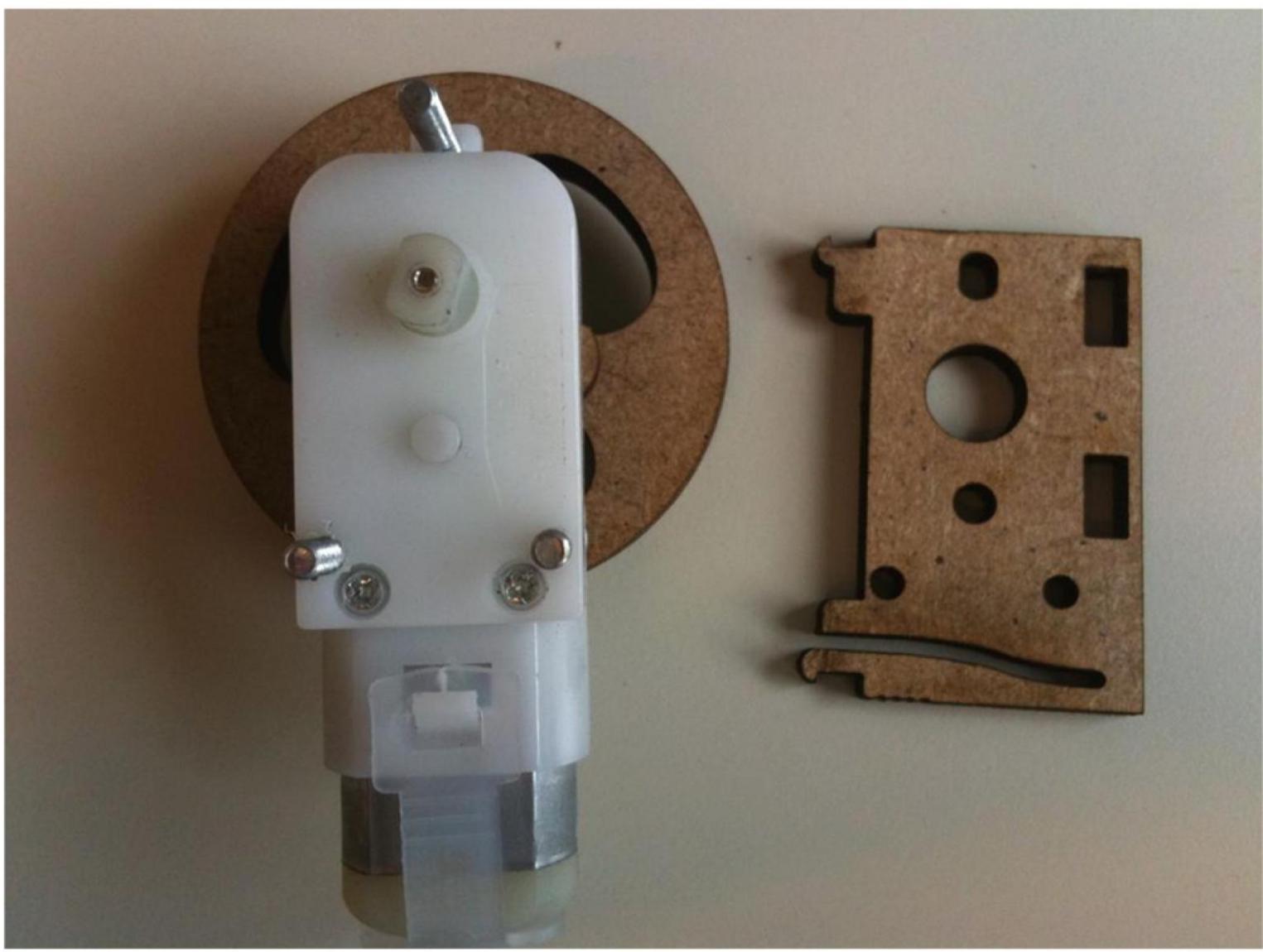
**white side**



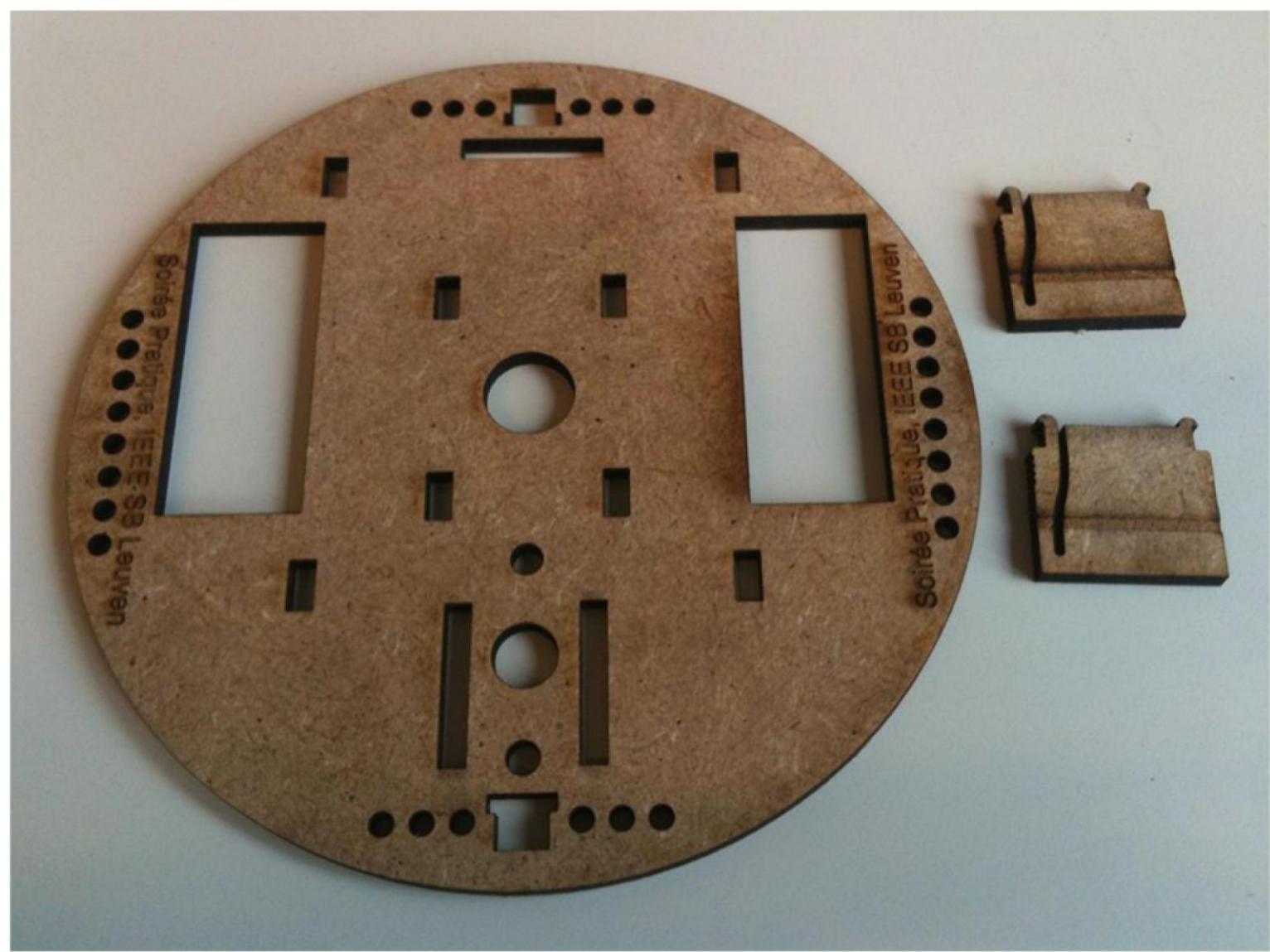
**green side**

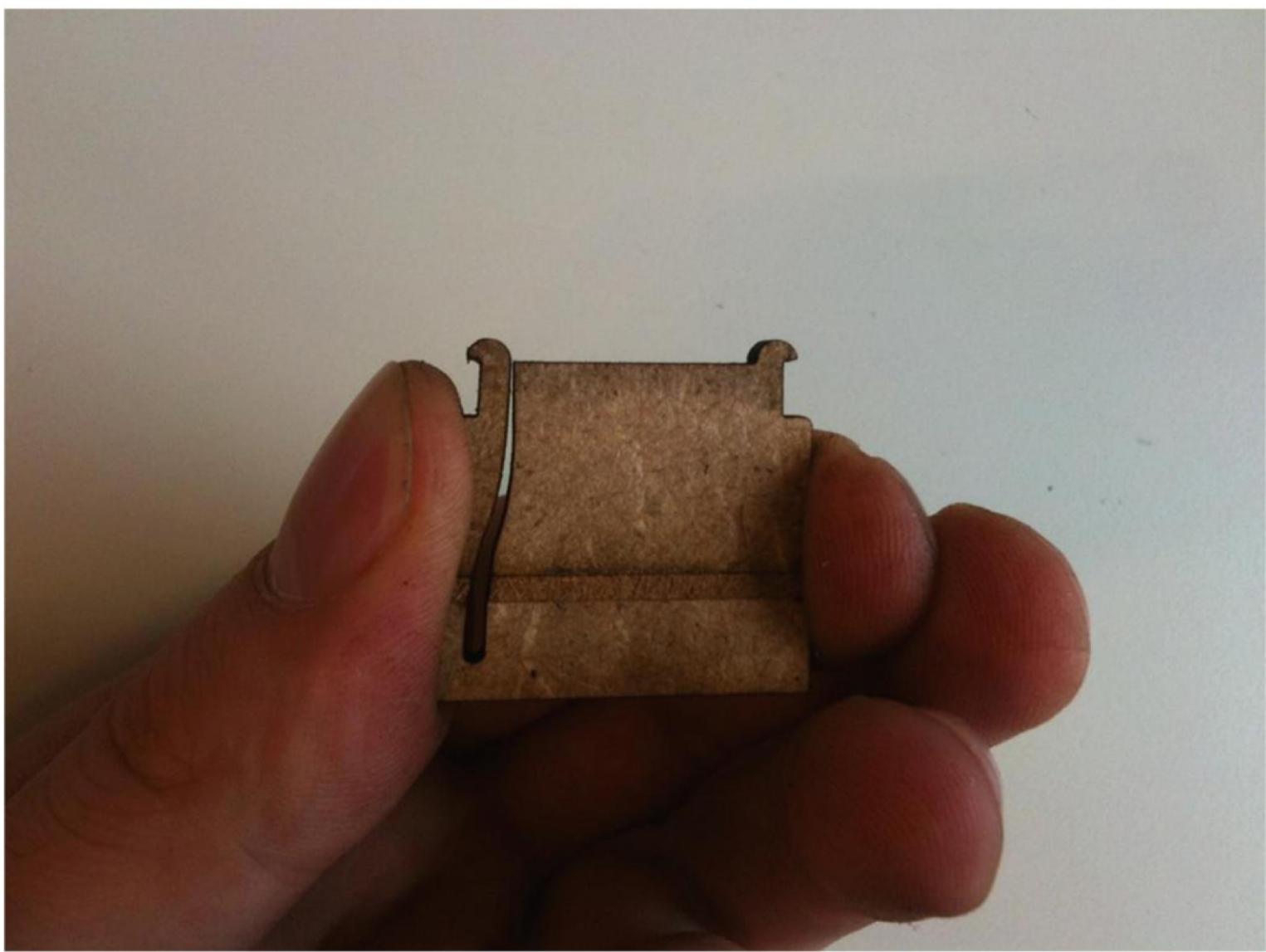


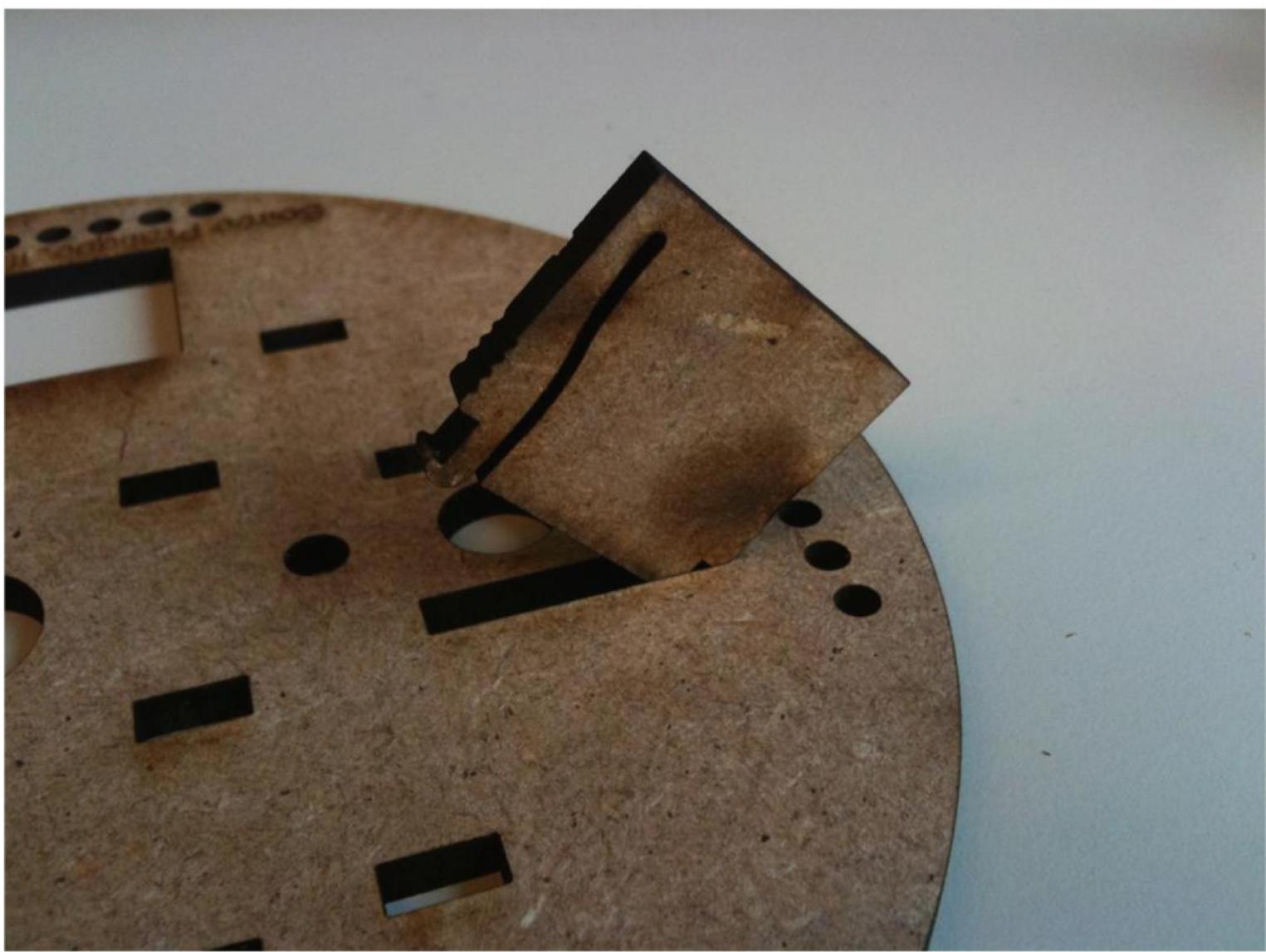


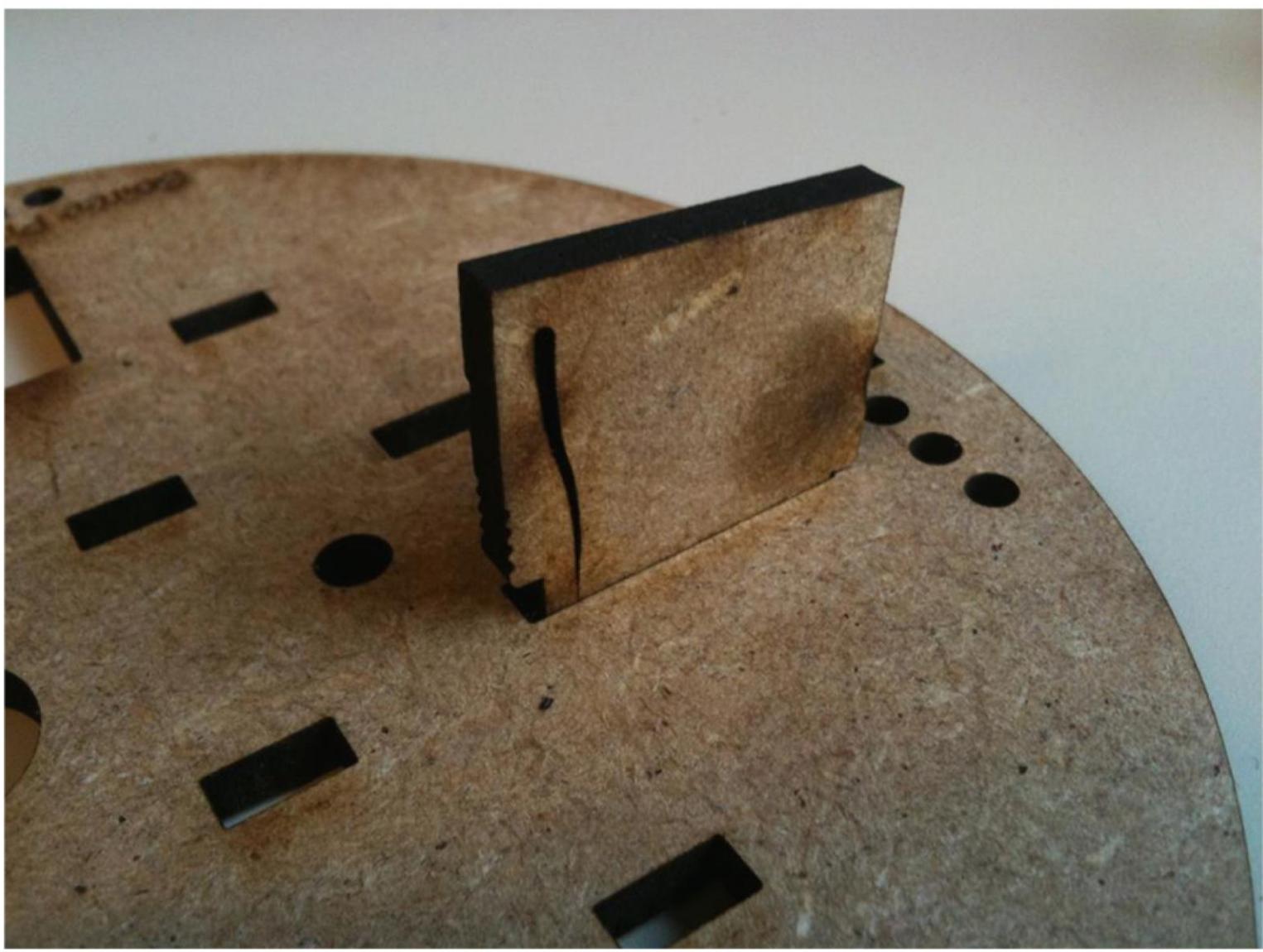


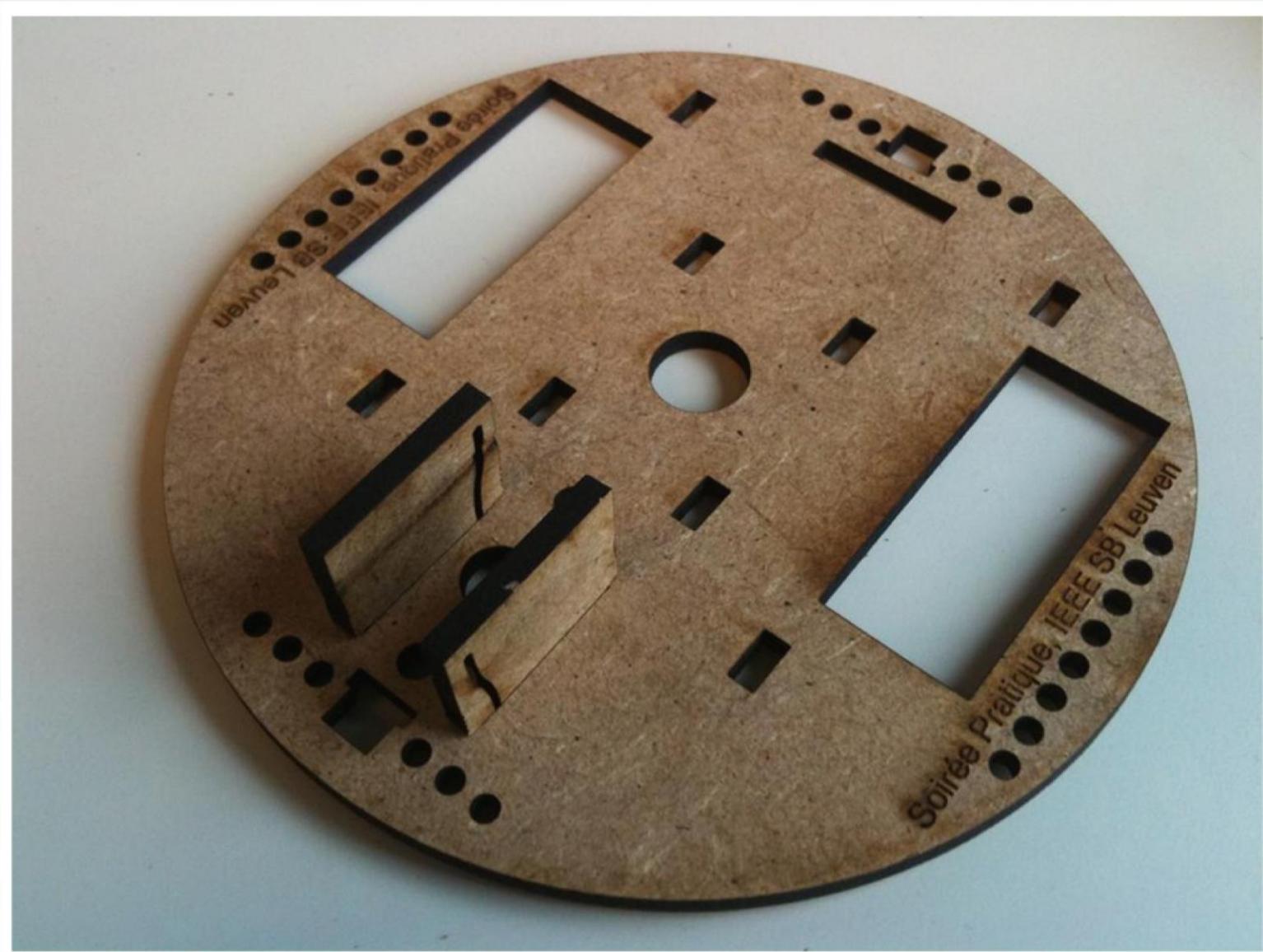


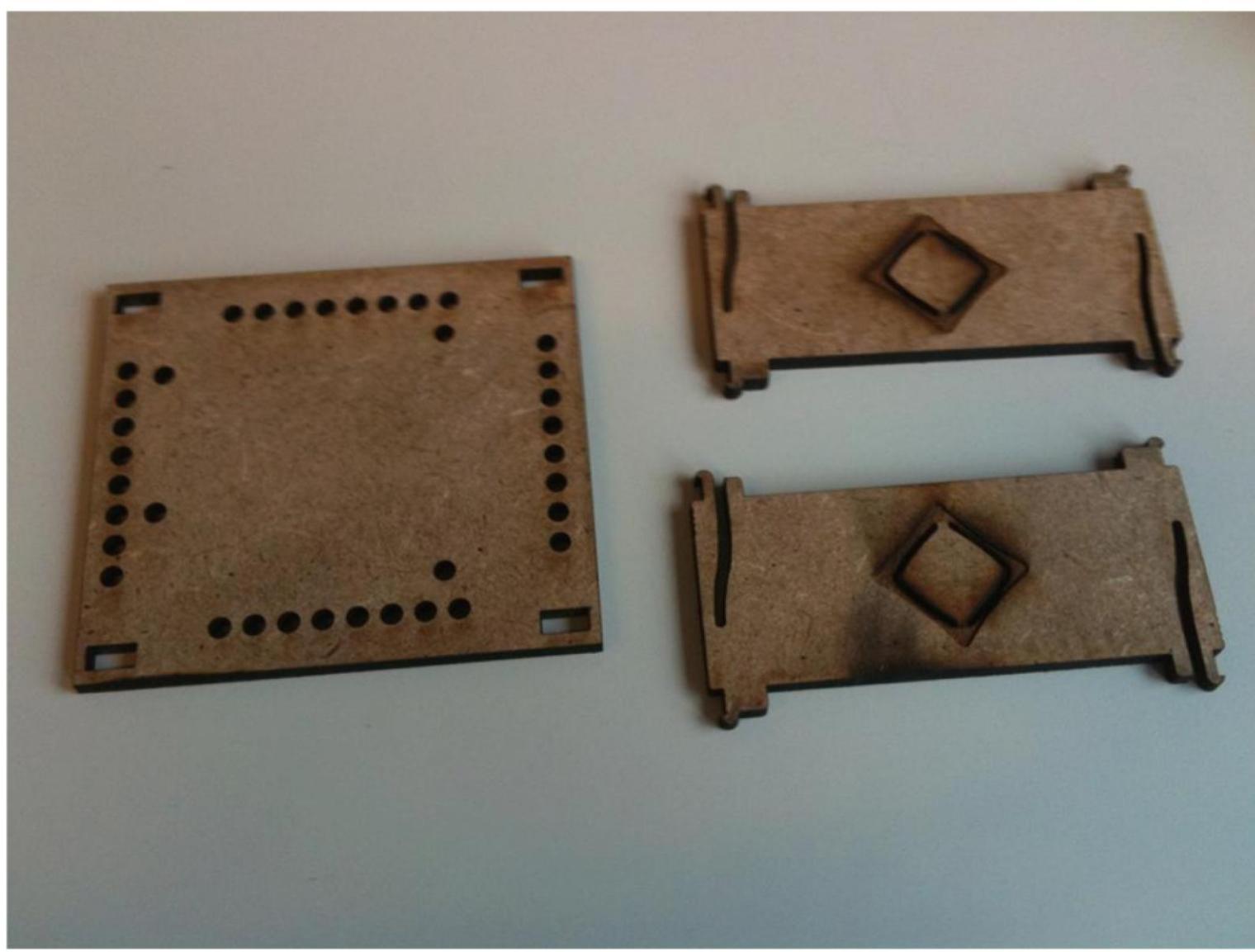


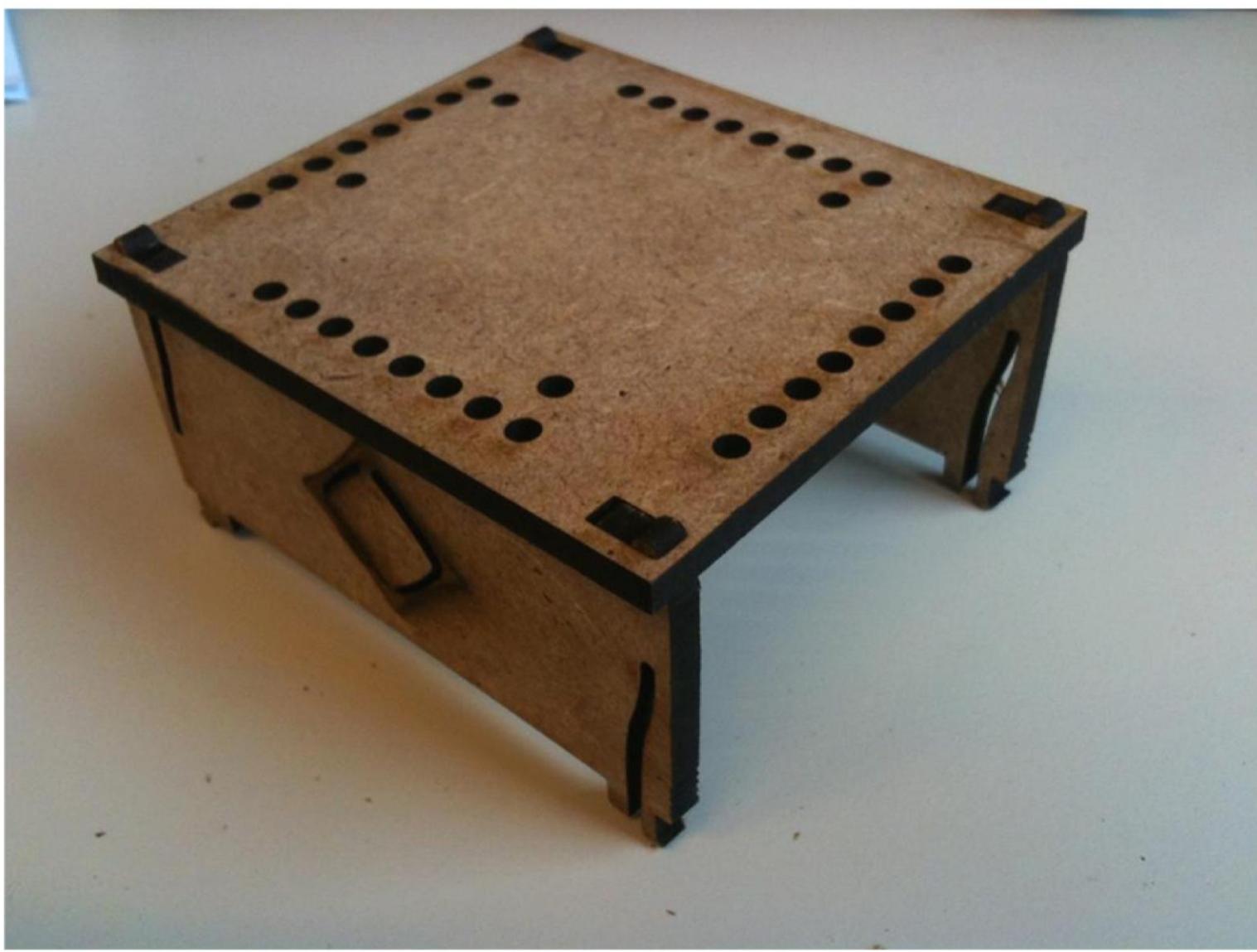




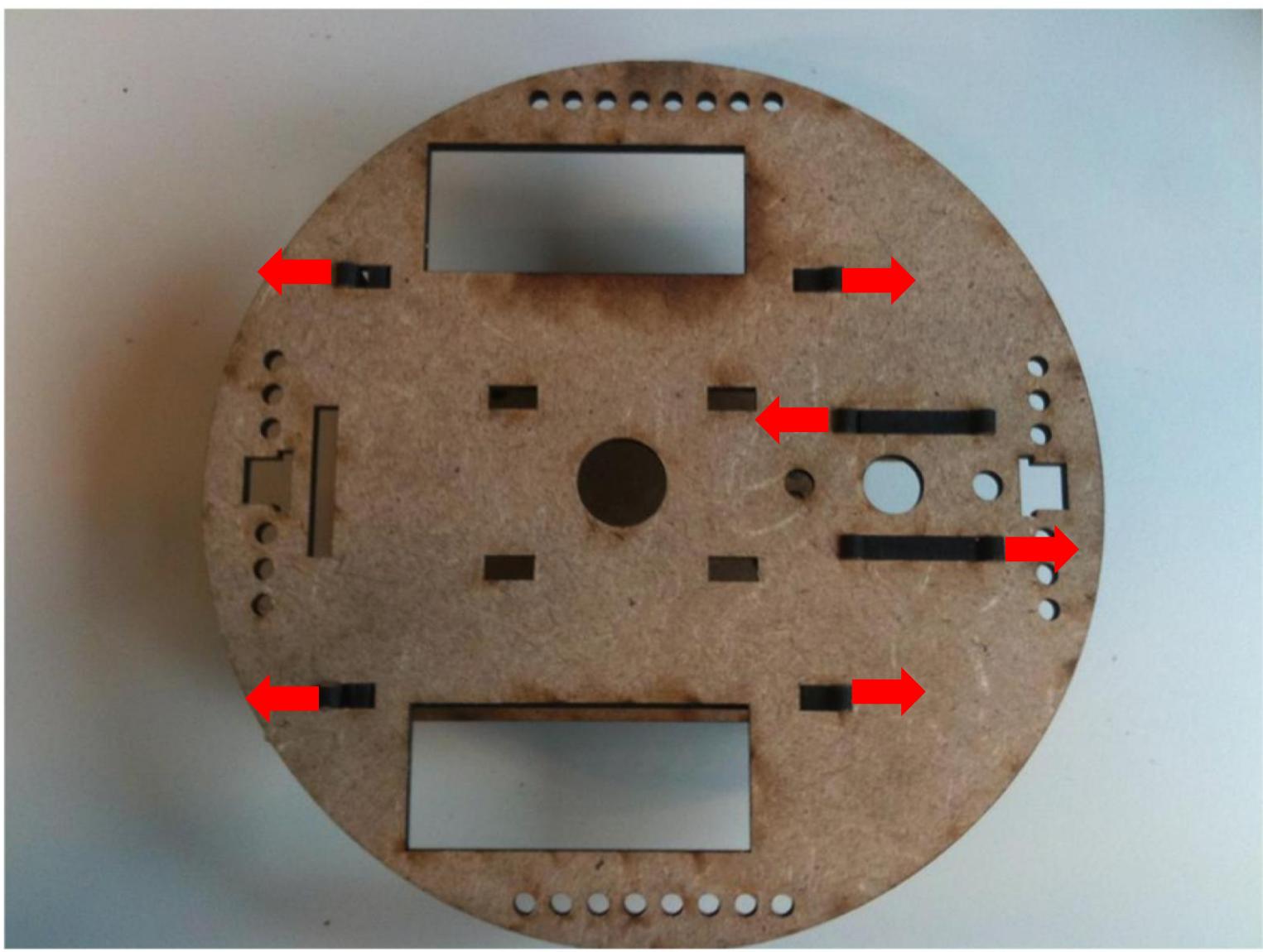


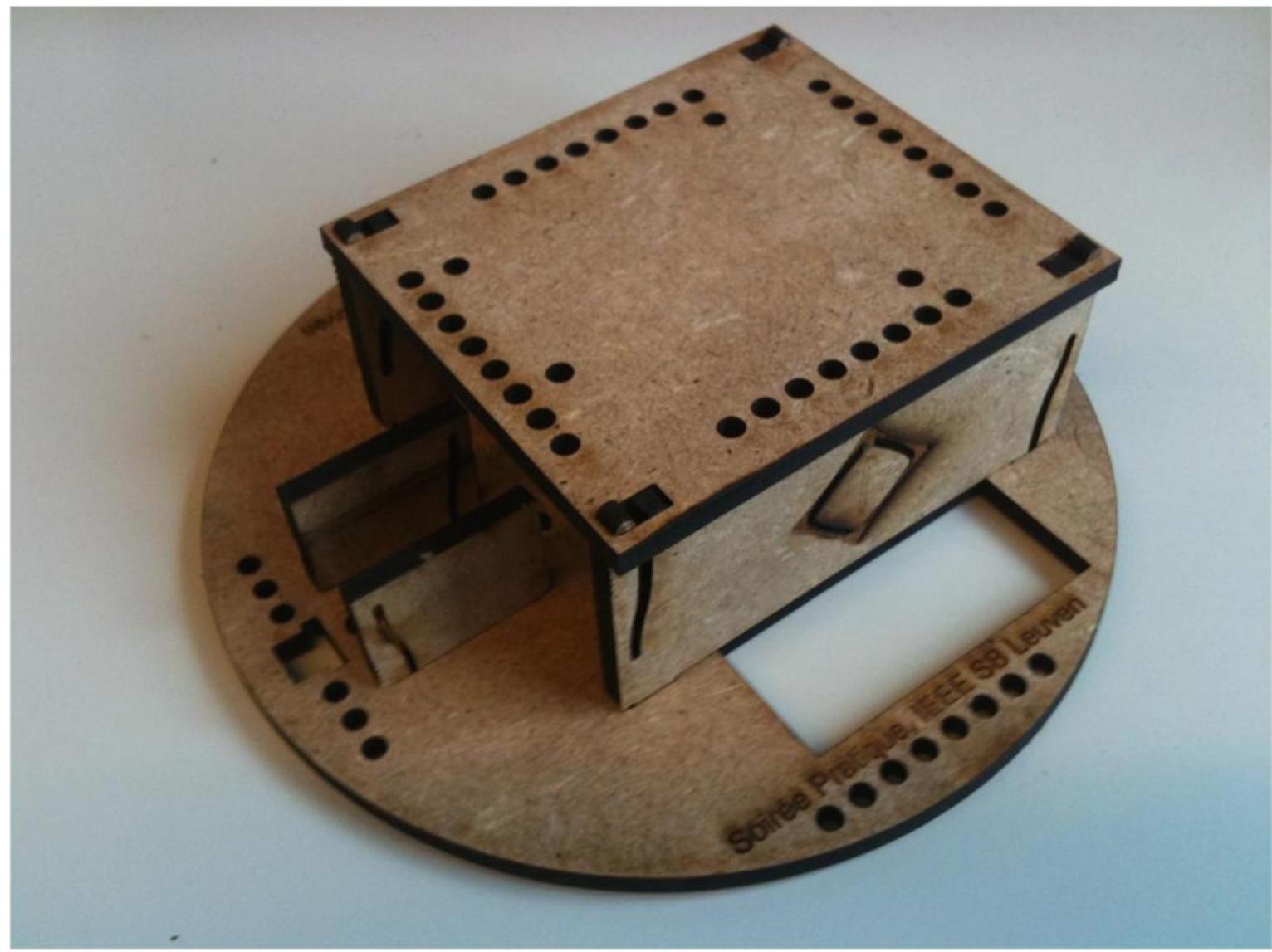


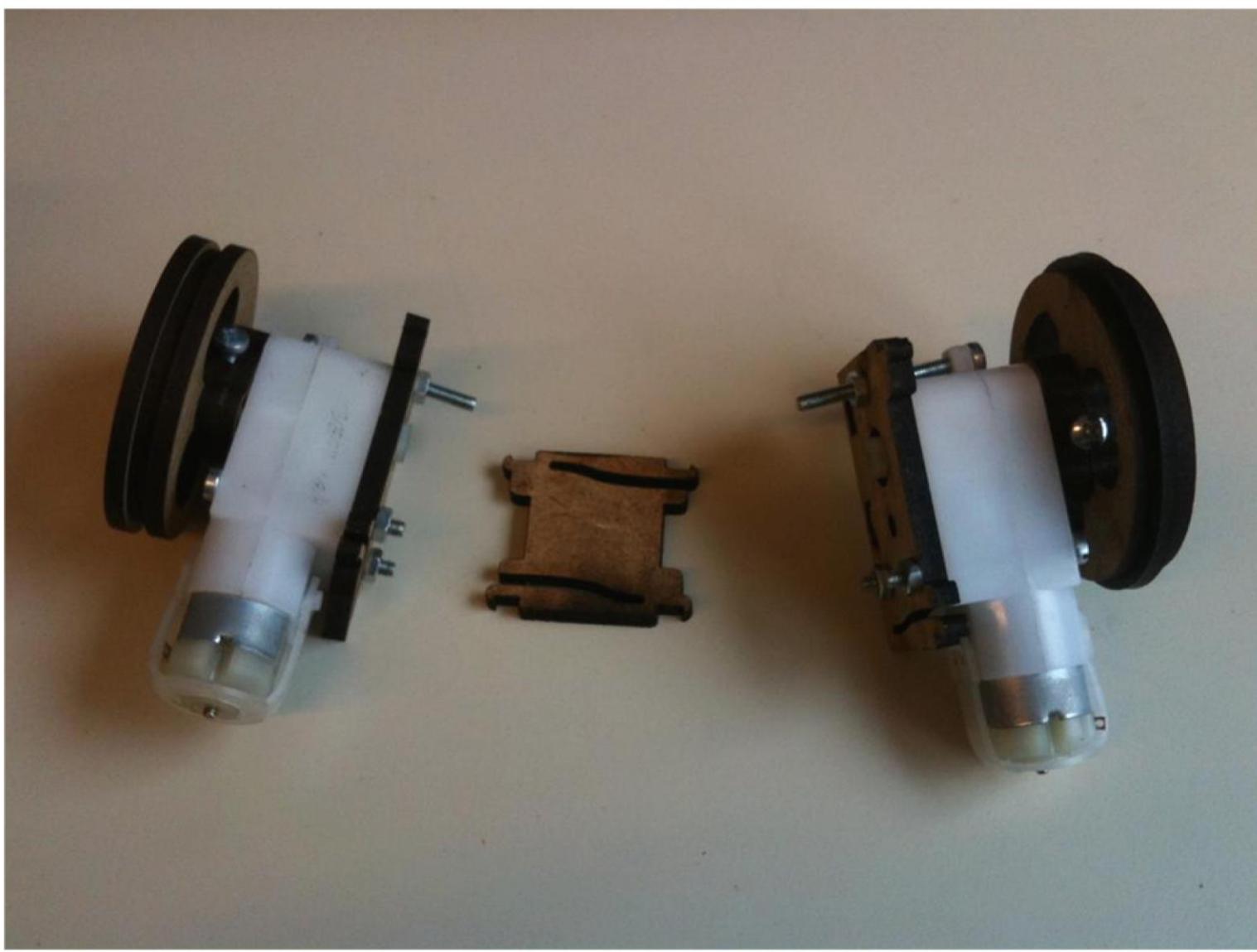


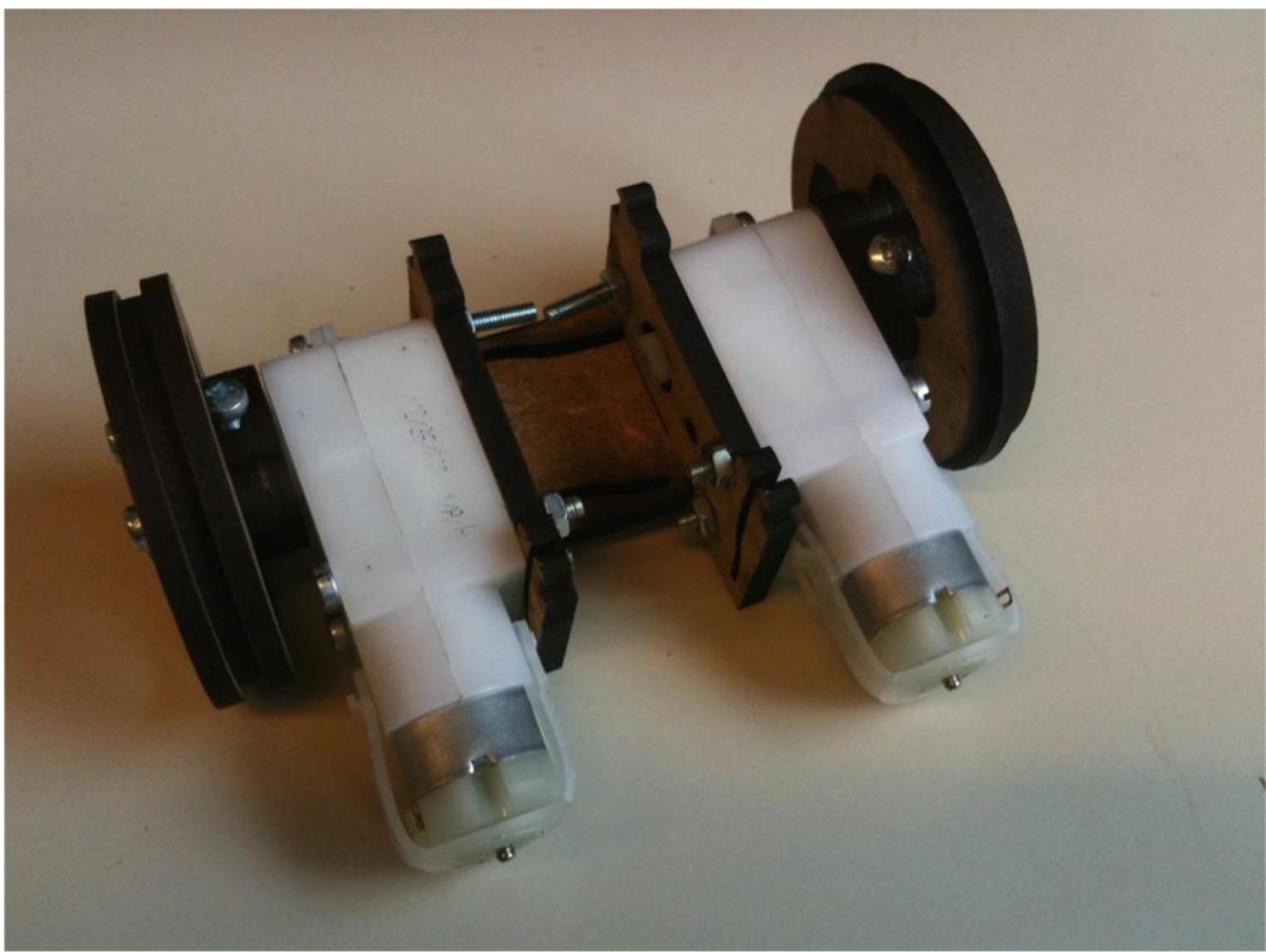


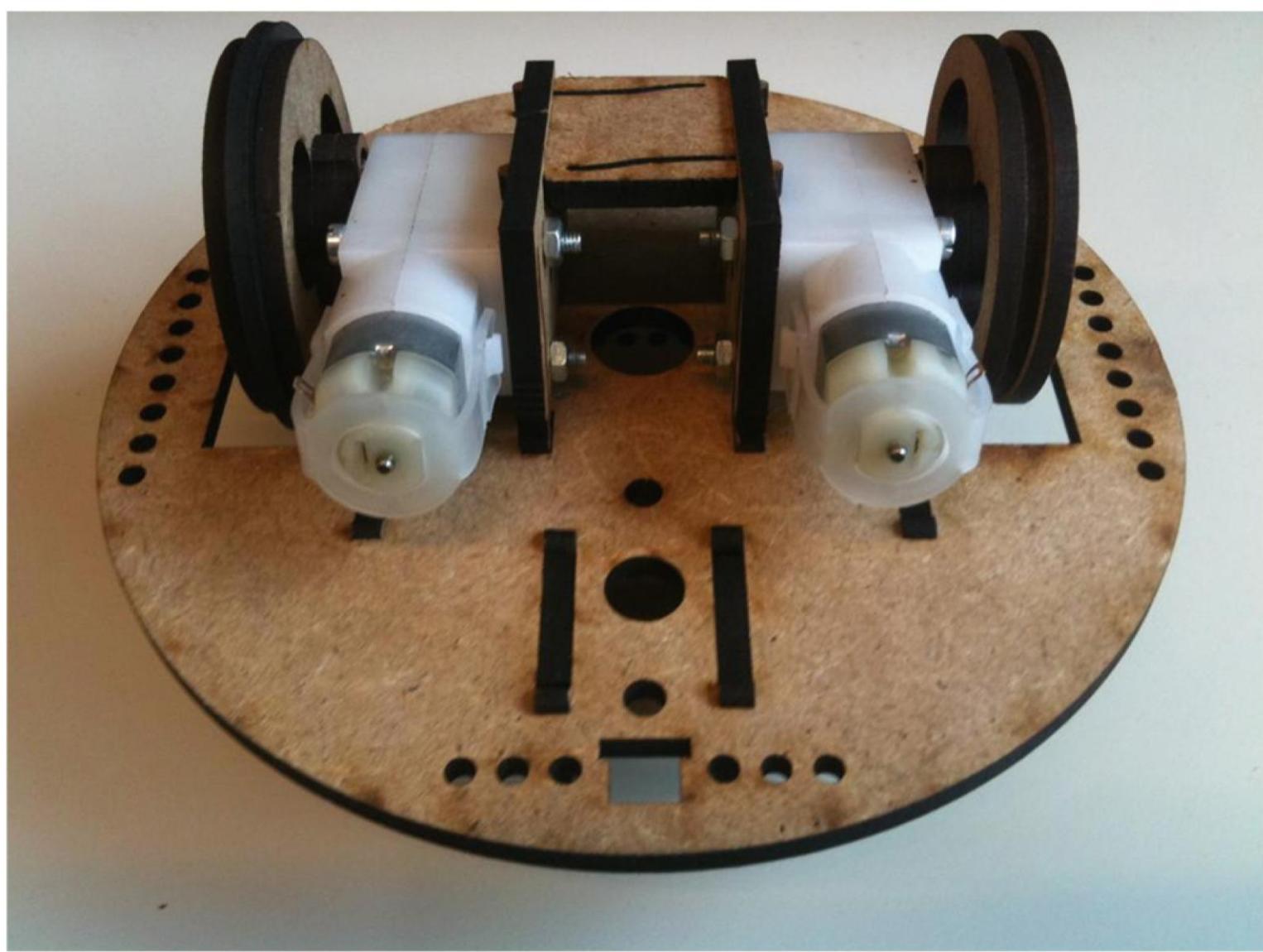


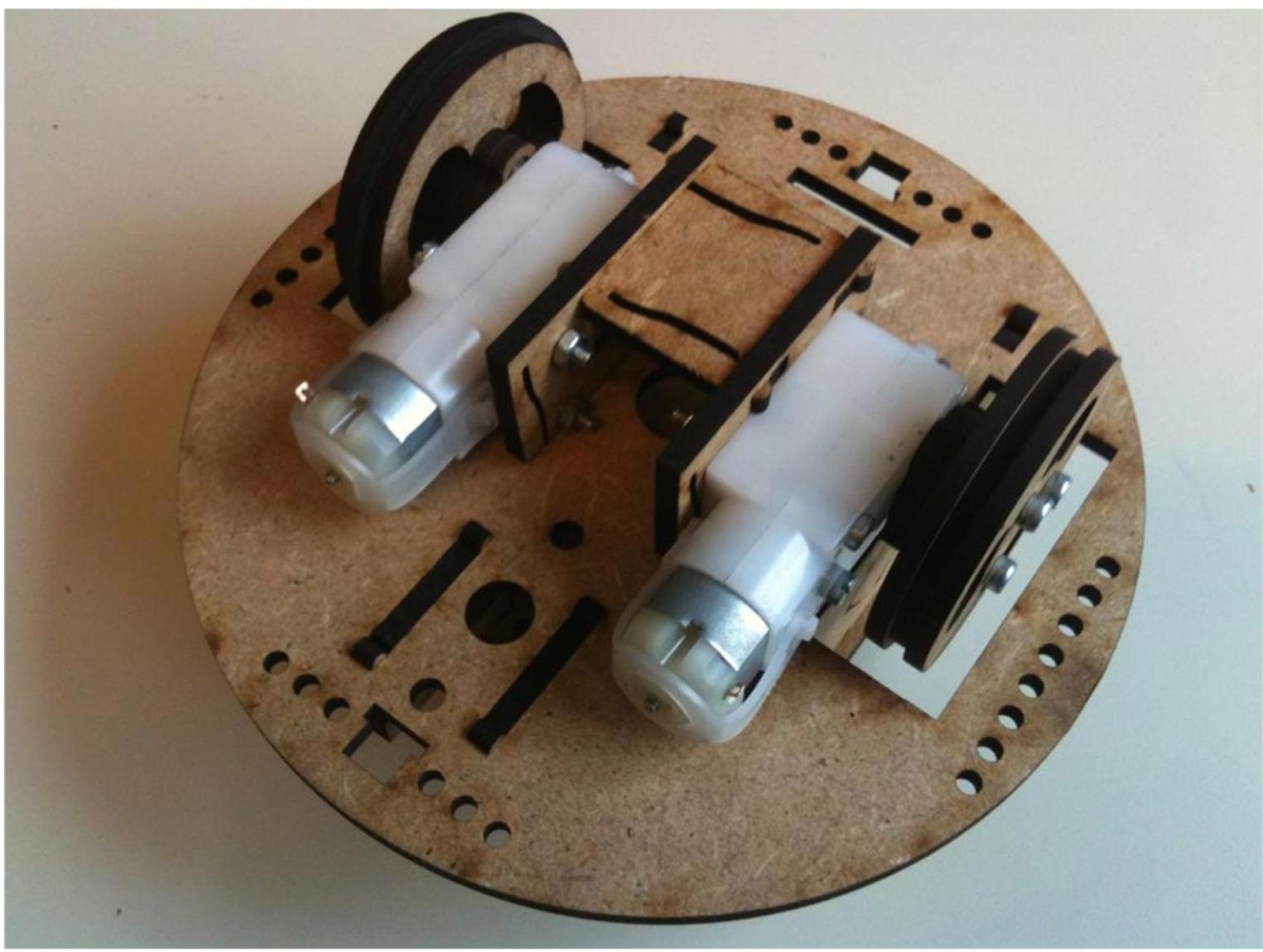


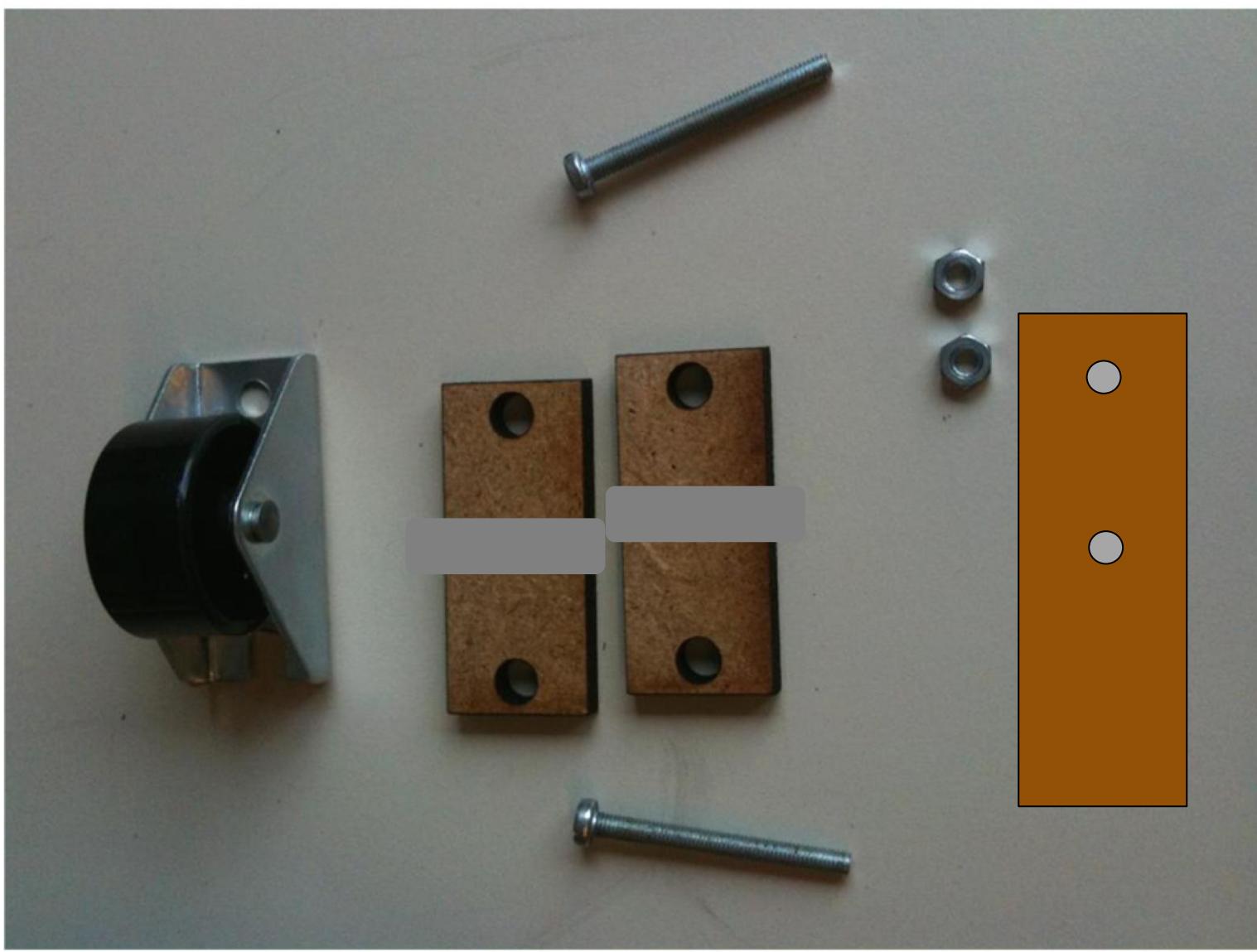


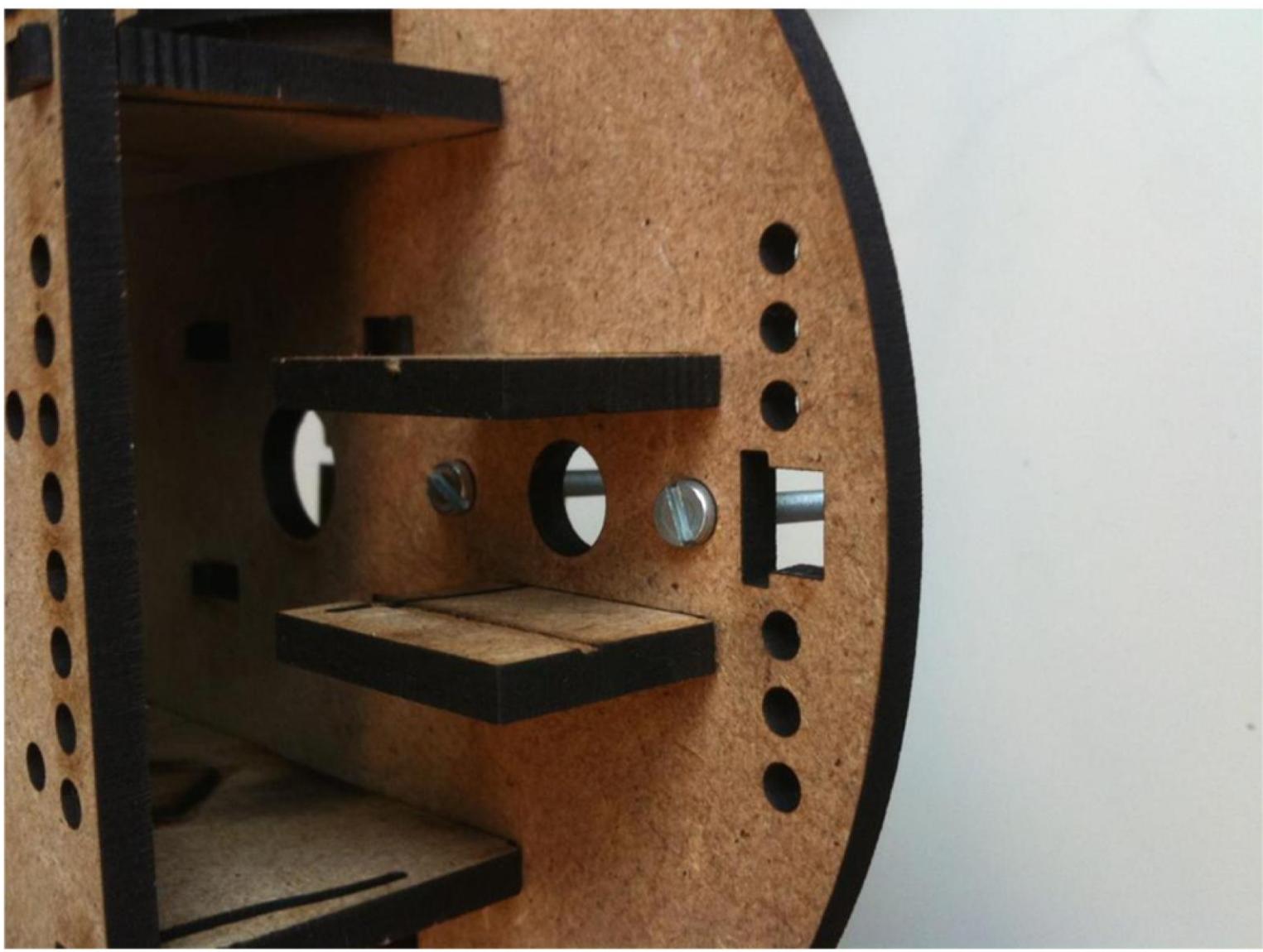


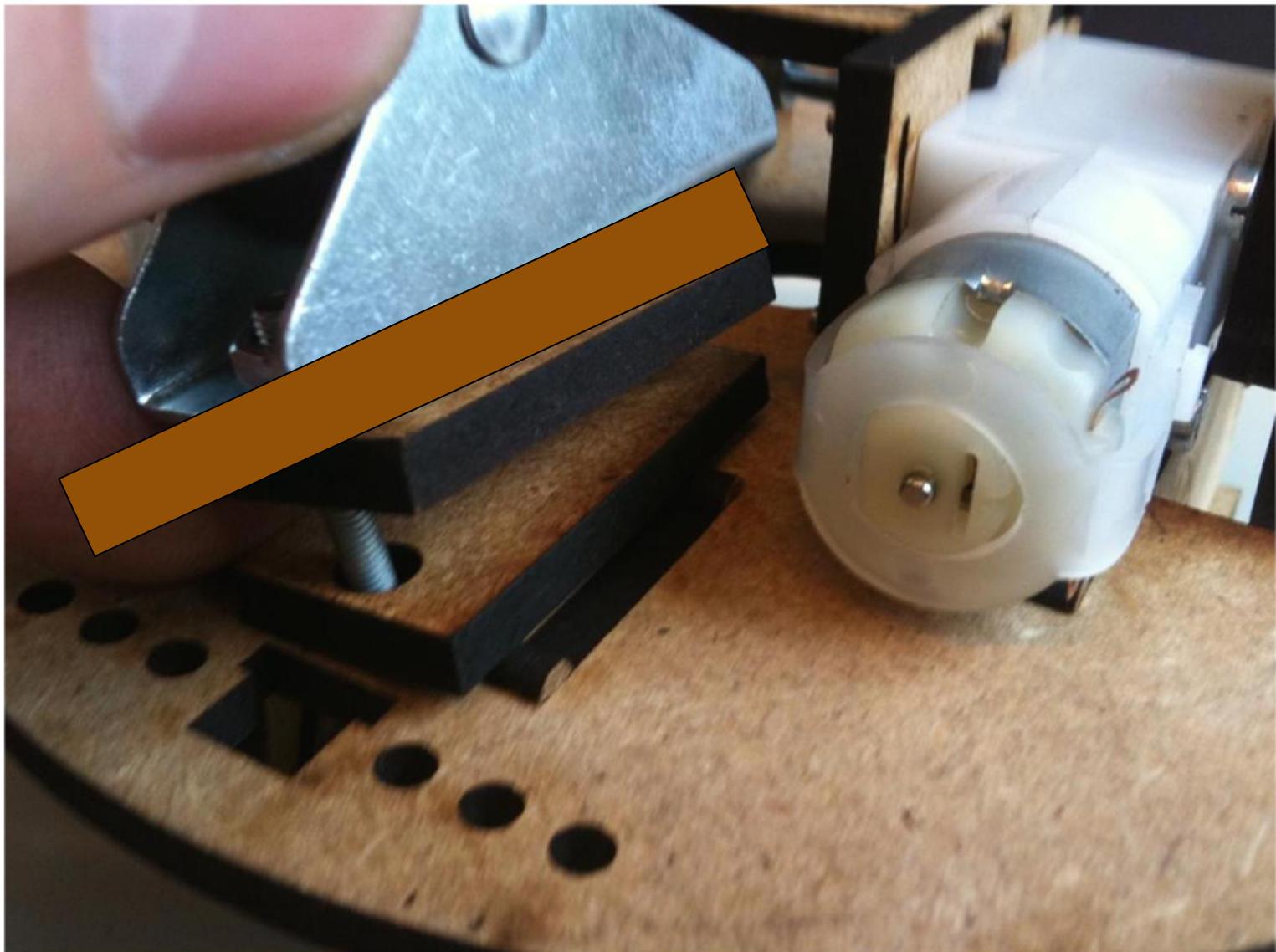


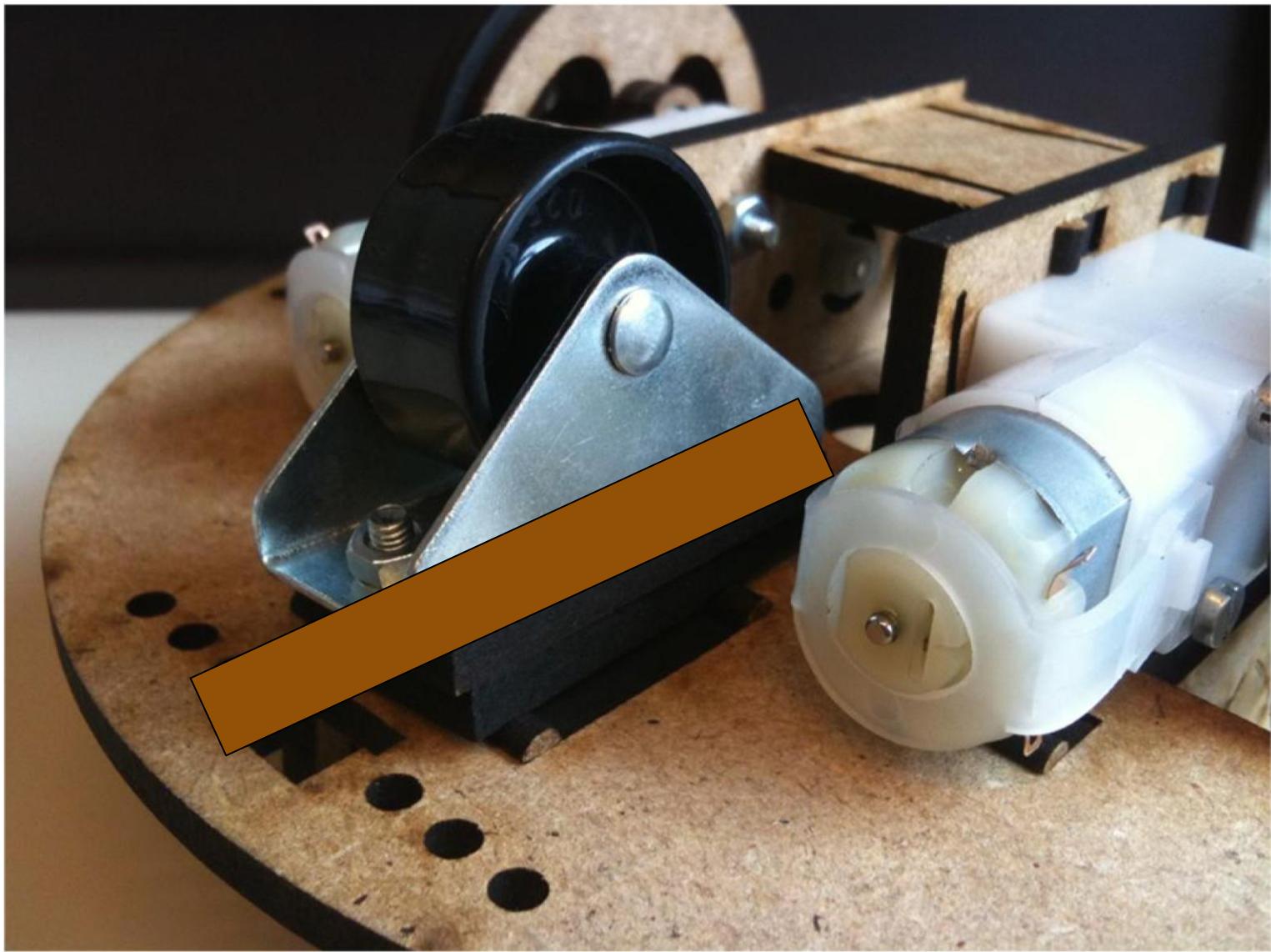




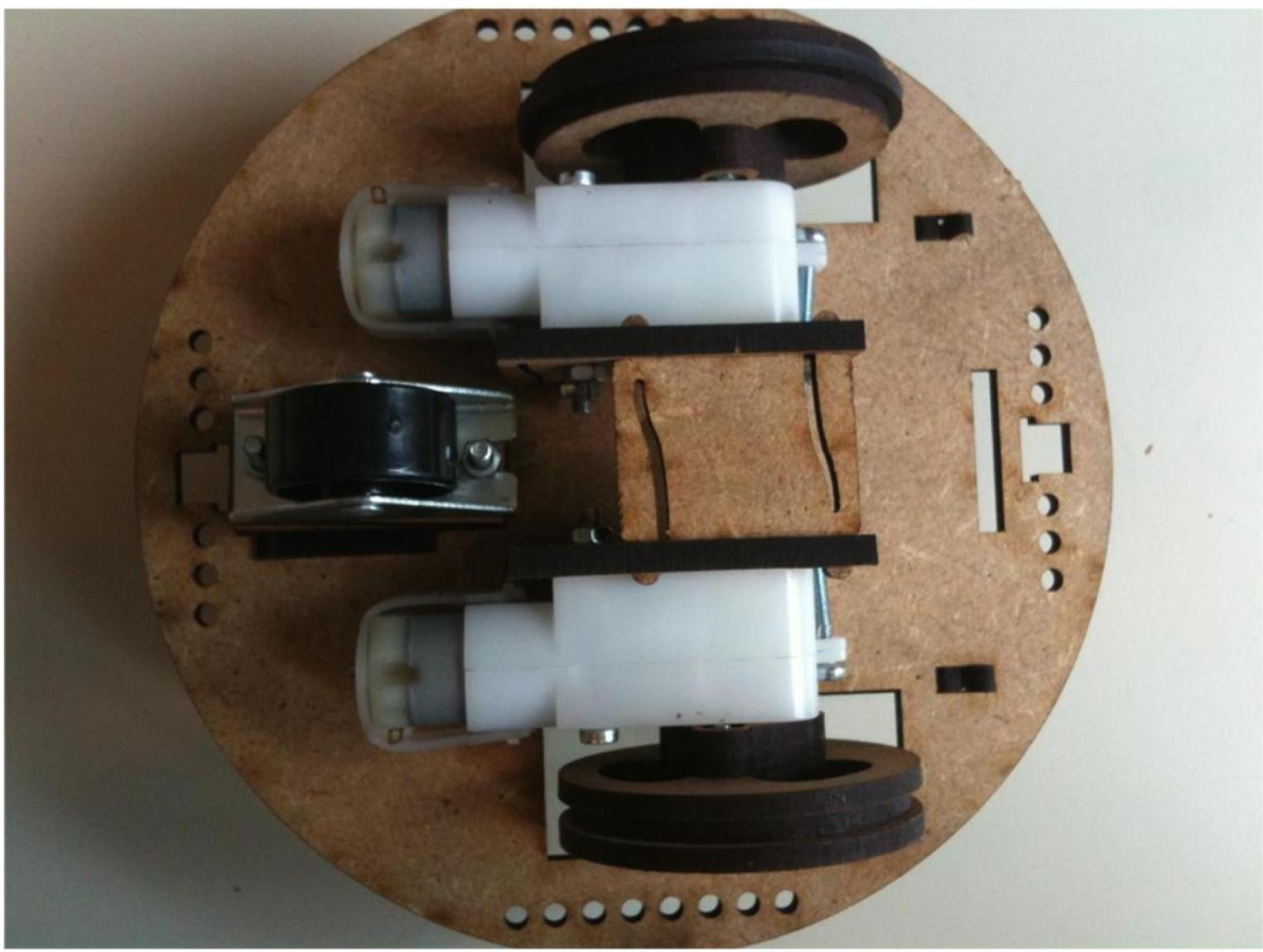


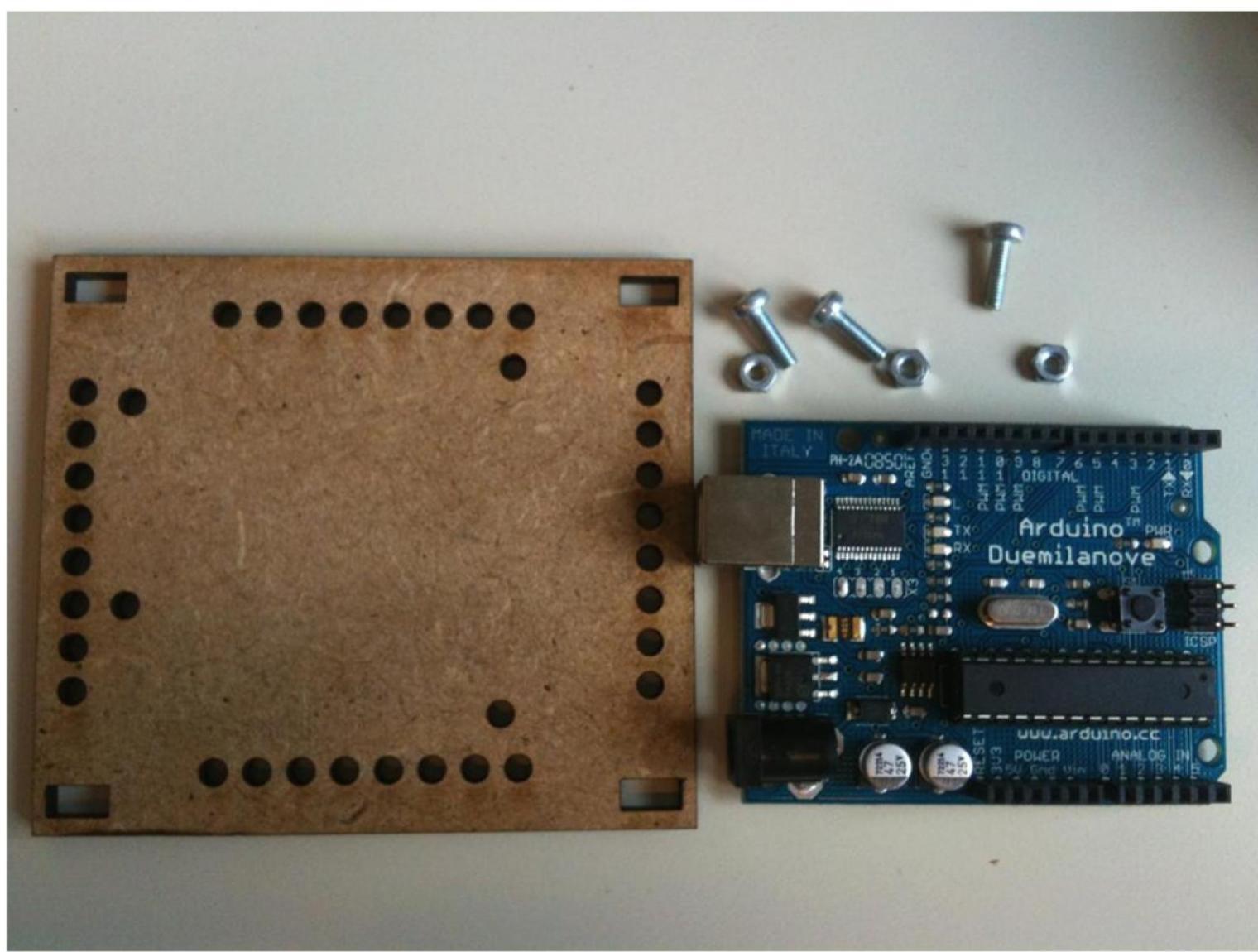


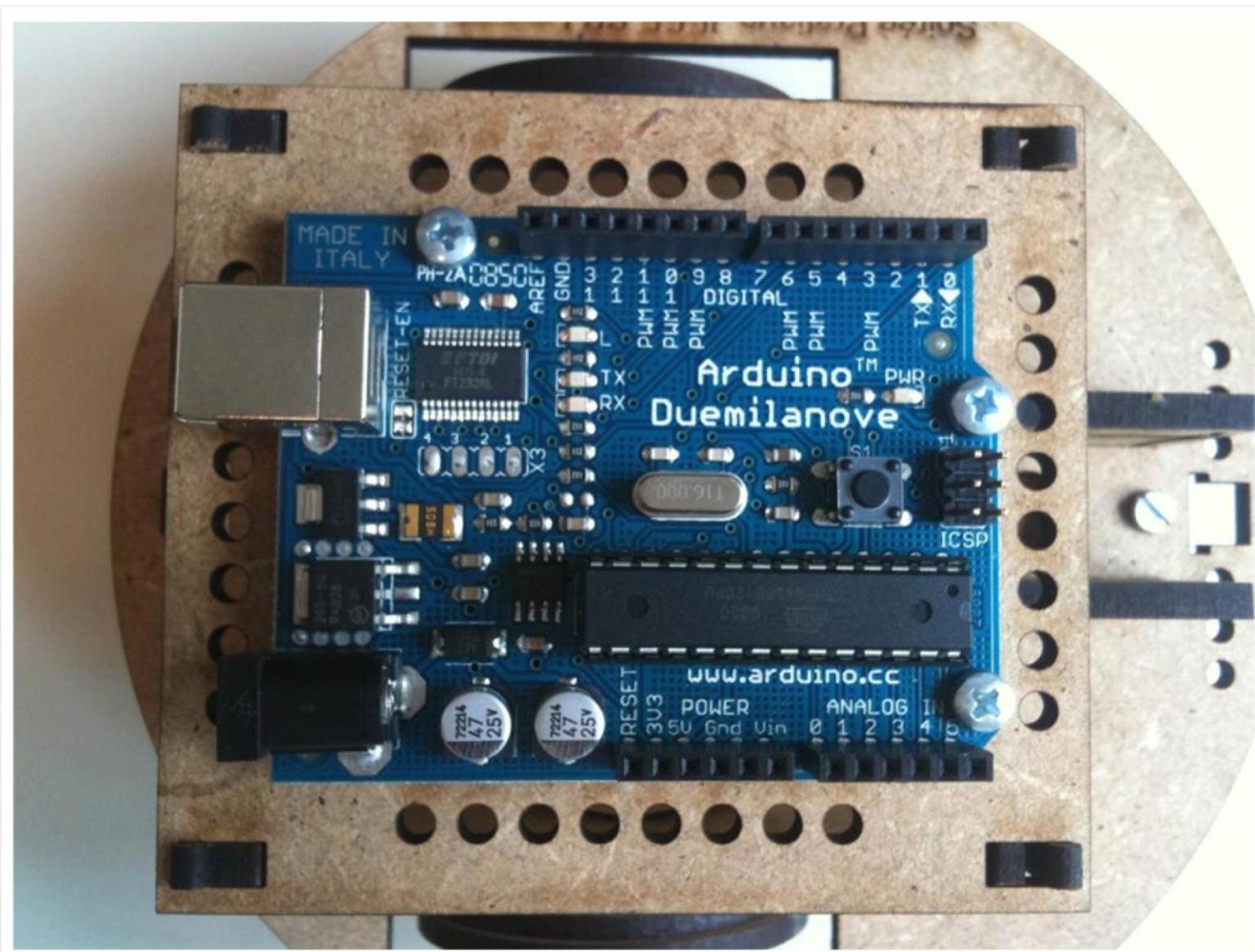


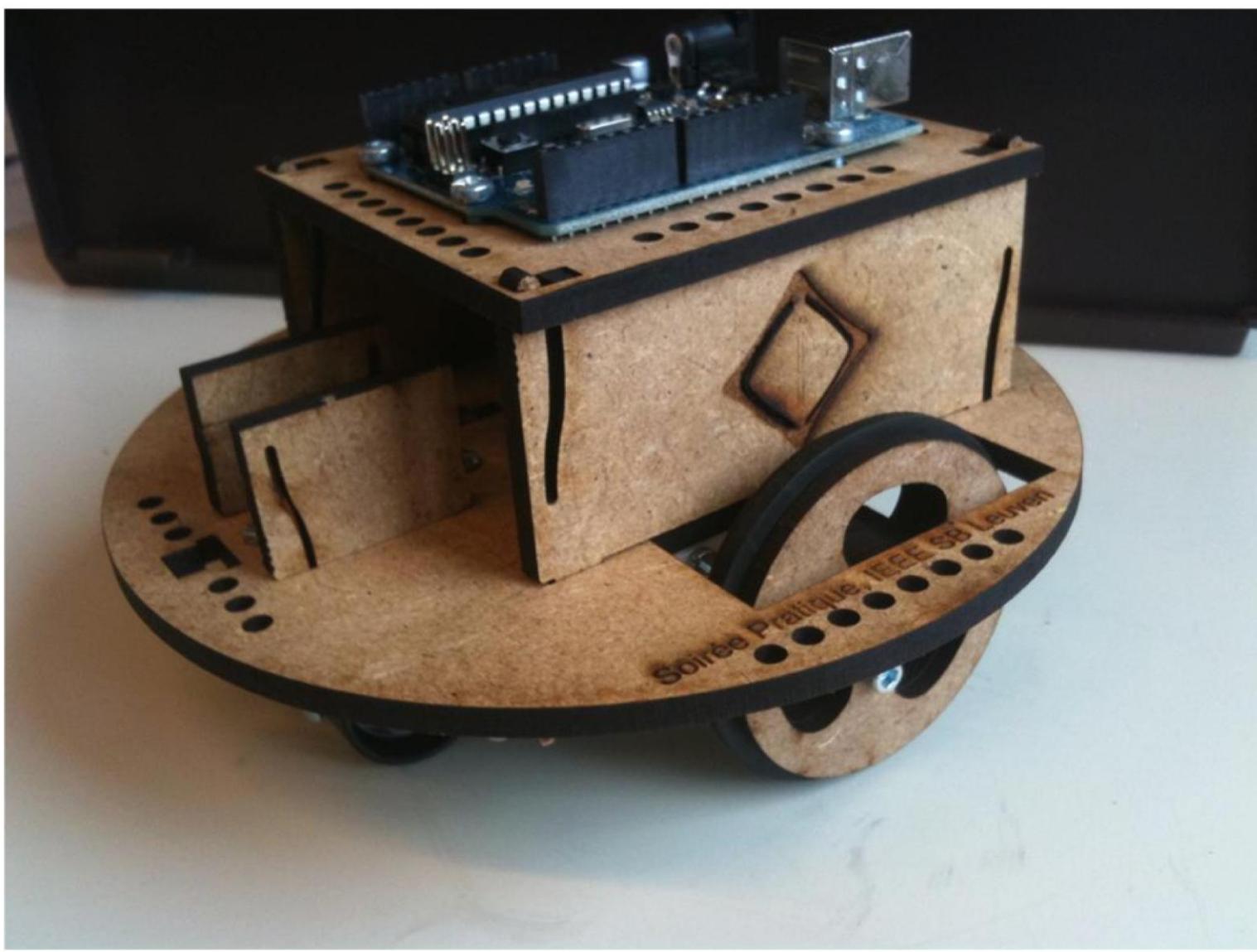


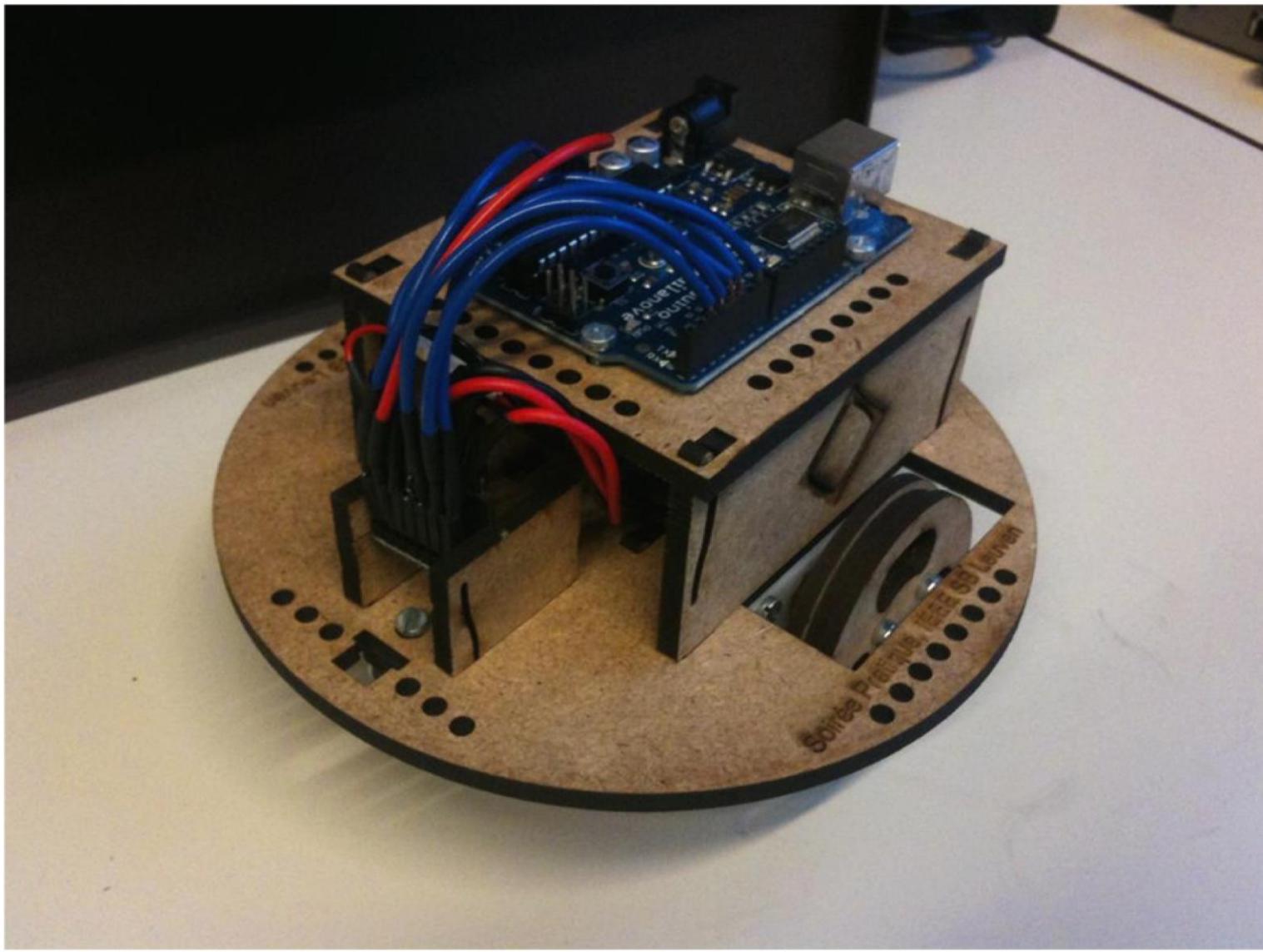










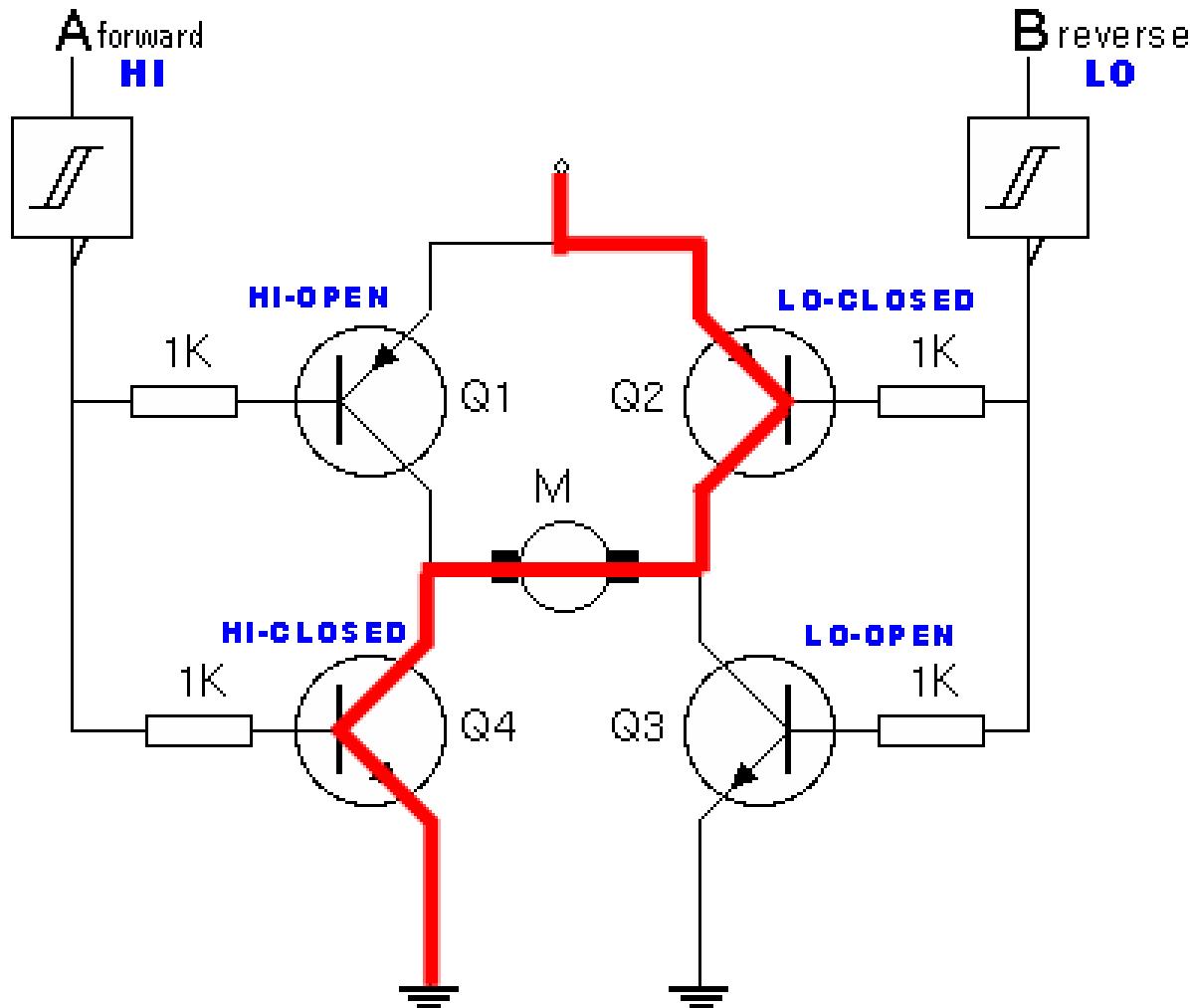




# Soirée Pratique Build your own robot Motor control

**Part 2:  
Using the standard motor controller**

# H-bridge

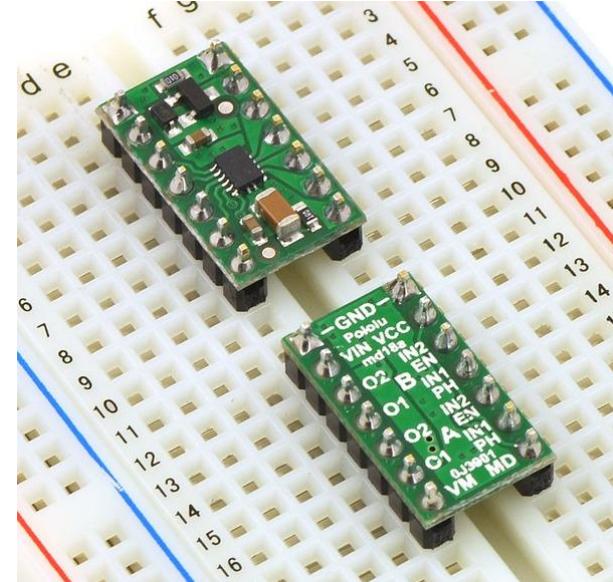


# The motor controller

- Dual-H-bridge motor driver
- Motor supply voltage: 2–11 V
- Logic supply voltage: 2–7 V
- Output current: 1.2 A continuous (1.5 A peak) per motor
- Two possible interface modes: IN/IN (outputs mostly mirror inputs) or PHASE/ENABLE (one pin for direction and another for speed)

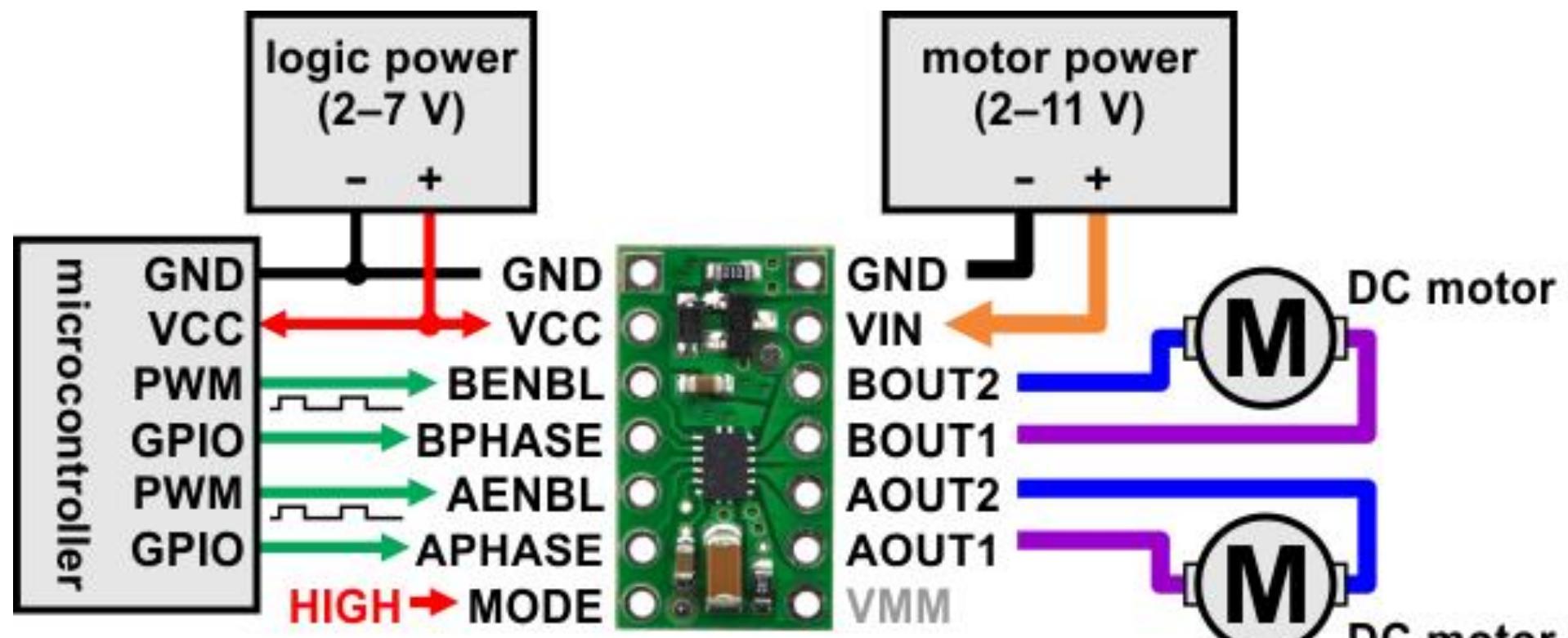
# The motor controller

- Undervoltage, overcurrent, and thermal shutdown
- Reverse-voltage protection circuit
- Compact size with the form factor of a 14-pin DIP package => it can fit on your breadboard
- You'll have to solder the leads yourself (ask help i.s.o. burning your controller!)



# The motor controller

## Arduino Batteries



## Arduino

# Turn 1 wheel back and forward setup

```
int enablePin = 5; // enable of motor controller connected to digital pin 5  
int phasePin = 4; // phase of motor controller connected to digital pin 4  
int modePin = 7; // mode of motor controller connected to digital pin 7  
int turn_direction = LOW;  
  
void setup() {  
    pinMode(modePin, OUTPUT);  
    pinMode(phasePin, OUTPUT);  
}  
}
```

# Turn 1 wheel back and forward loop

```
void loop() {  
    digitalWrite(modePin, HIGH); // put motor controller in phase/enable mode  
    digitalWrite(phasePin, turn_direction); // switch turn direction  
    // turn harder from min to max in increments of 5 points:  
    for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5) {  
        analogWrite(enablePin, fadeValue); // sets the value (range from 0 to 255)  
        delay(30); // wait for 30 milliseconds to see the effect  
    }  
    // turn slower from max to min in increments of 5 points:  
    for(int fadeValue = 255 ; fadeValue >= 0; fadeValue -=5) {  
        analogWrite(enablePin, fadeValue); // sets the value (range from 0 to 255)  
        delay(30); // wait for 30 milliseconds to see the effect  
    }  
    if(turn_direction == LOW)  
        turn_direction = HIGH;  
    else  
        turn_direction = LOW;  
}
```

# **Next session (in 2 weeks)**

## **Sensors**

- Bring sensors or order them with us
- Order as soon as possible! (This week)
  
- See you all next session!