

PROJECT: MINDFREAKS

BY-MEGASPARK

TEAM MEMBERS:

- 1.PATEL ATISH KUMAR
2. VICKI ANAND
3. TANUSHREE GUPTA
4. SHREYA BHARTI

TEAM MENTORS:

- *SWAPNIL UPADHYAY
- *VARSHA LALWANI
- *CHIRAG JHA



How do we get the idea??

This project is inspired by one of our phy101 lab experiment i.e. LINEAR AIRTRACK. Doing this experiment a thought came what if we can control the movement of the object on the linear air bed . In addition to this we come to know about **Emotiv EPOC headset** through e-club lecture series, so we combined both .The setup of this project resmbles the air track . We used the concept of linear airtrack and made a game out of it.



Emotiv EPOC HEADSET

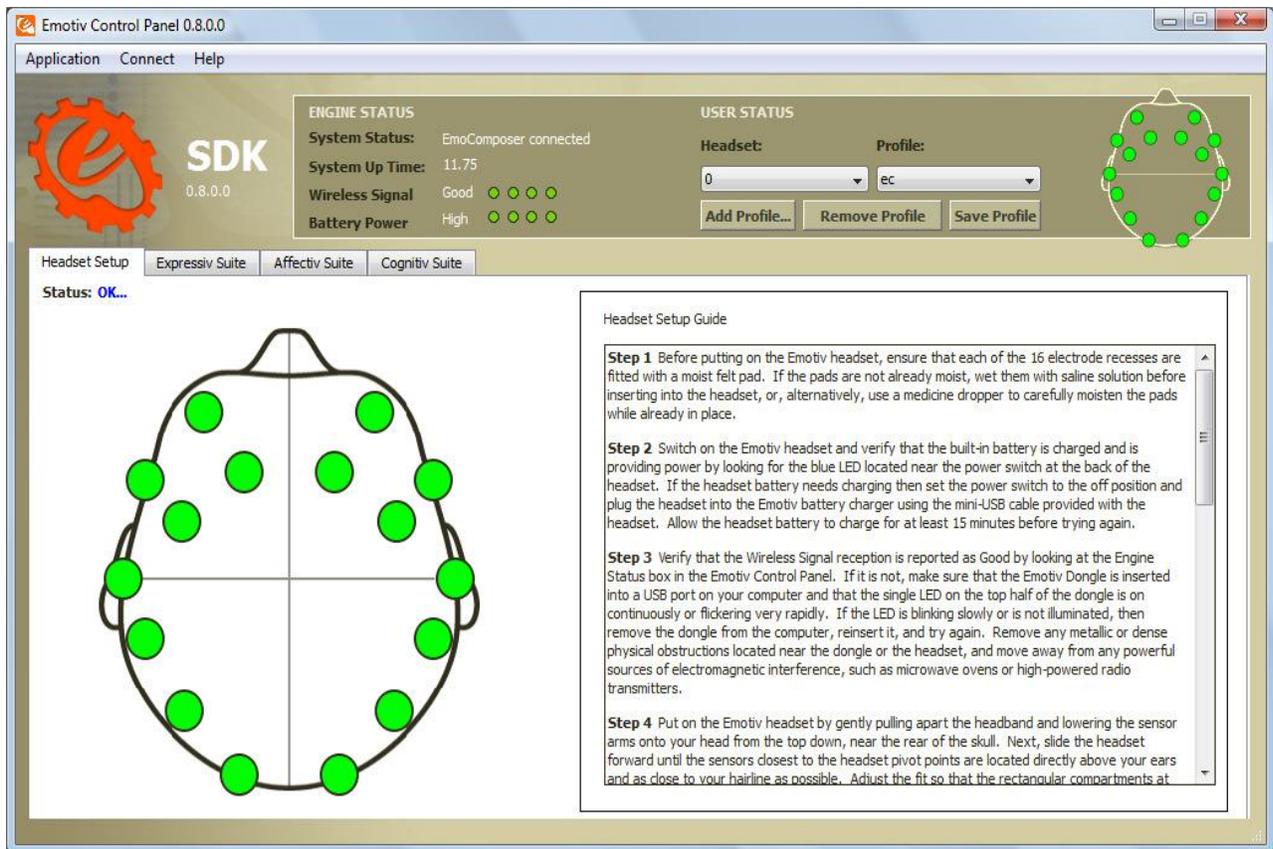
Its a headset which make it possible for games to be controlled and influenced by the player's mind, and facial expressions. The EPOC has 14 electrodes[8][9] (compared to the 19 electrodes of a standard medical EEG, and the 3 of OCZ's NIA features and a multiple of NeuroSky's single electrode). It also has a two-axis gyro for detecting head movements.



in the image above the person is wearing the emotiv EPOC headset. The EPOC brain helmet has eighteen sockets and can hold sixteen nodes or sensor pads. The remaining two sockets usually hold rubber pads, and are known as the secondary reference sensors, which are located immediately below and behind the ears. The primary reference sensors, which generally hold a normal sensor pad, are located immediately above and behind the ears. **The sensors pads detect electrical activity on the surface of the brain.**

EMOTIV CONTROL PANEL

It helps you connect your pc with the headset. Signals from the wireless headset reach your pc via bluetooth and this software shows you signals from your brain at different points.



above is the image of the control panel of emotive headset . Green dots here represent the signal received is good.

For rest of the information about how to use emotiv is in its user manual, which can be downloaded from:

<http://www.google.co.in/url?sa=t&rct=j&q=sd%20user%20manual&source=web&cd=1&cad=ja&ved=0CCKQFjAA&url=http%3A%2F%2Femotiv.com%2Fdeveloper%2FSDK%2FUserManual.pdf&ei=s6H3Ua36K4WzrgeWroDgBA&usg=AFQjCNEJZFyo2kyaDgQ7f4Oz2o8crx8amg&bvm=bv.49967636,d.bmk>

ARDUINO UNO

Arduino is a single-board microcontroller to make using electronics in multidisciplinary projects more accessible. The hardware consists of a simple open source hardware board designed around an 8-bit Atmel AVR microcontroller, or a 32-bit Atmel ARM. The software consists of a standard programming language compiler and a boot loader that executes on the microcontroller.

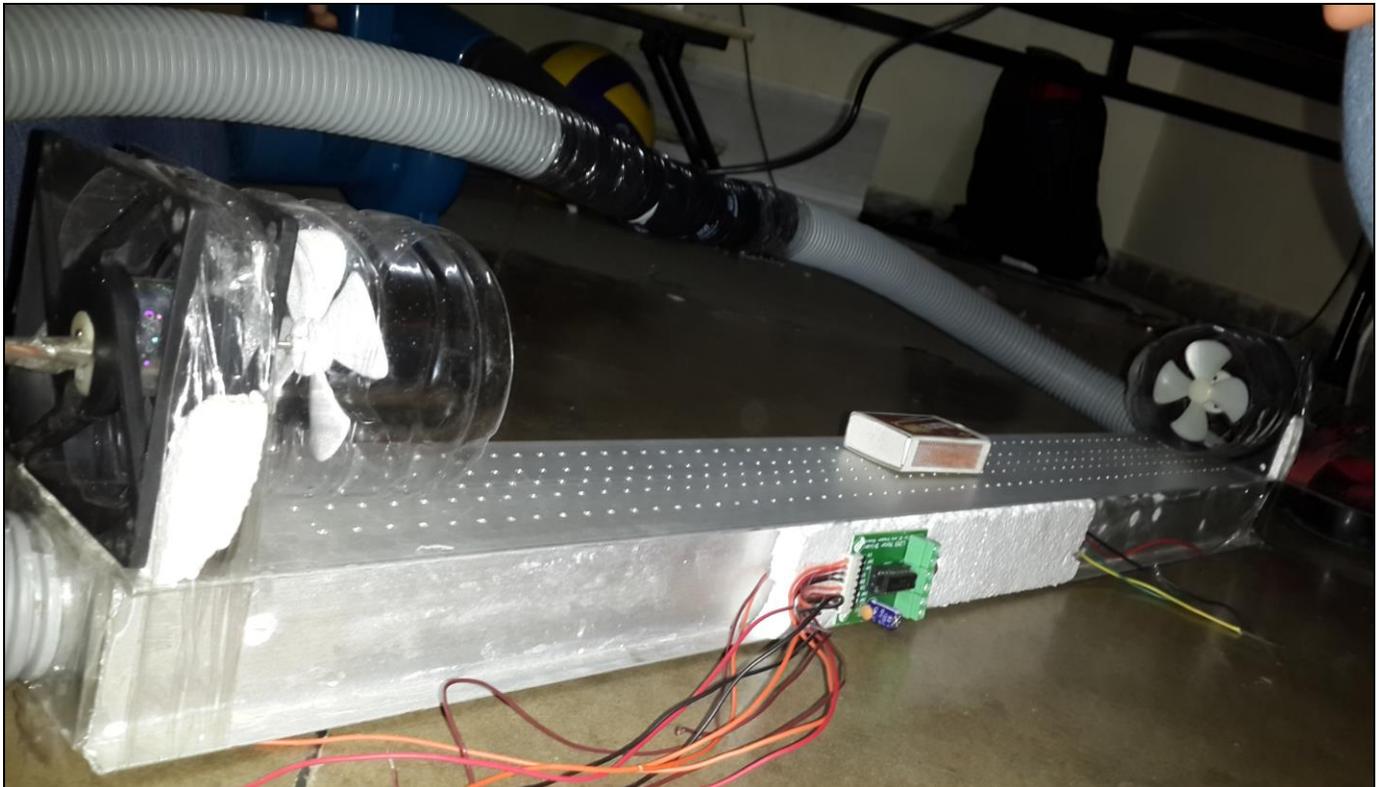
Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the

board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software running on a computer (e.g. Flash, Processing, MaxMSP).



link for arduino uno datasheet: <http://arduino.cc/en/Main/arduinoBoardUno> ,
<http://www.arduino.cc/>

Putting all the components together



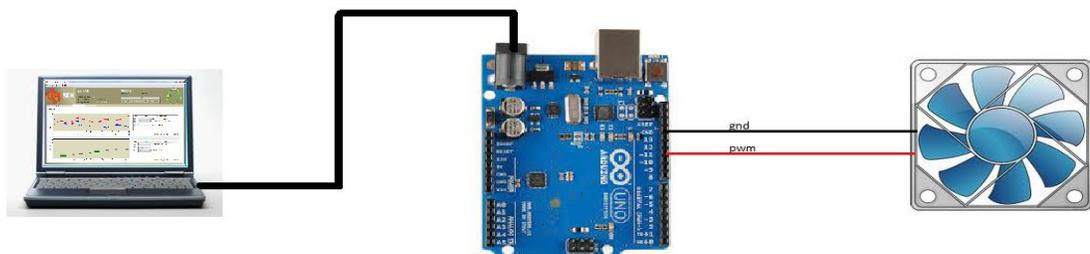
The game constitutes of

- Air bed
- 2 propellers
- Arduino uno
- Pump
- Emotiv headset
- Transistor
- Match box
- 12V Battery

The game starts by placing the box in between the air bed .Now as both the propellers are controlled by emotiv headset ,the game is basically running by players' mind.The propellers speed are controlled by player's mind. The more he concentrate , the more is the speed of propellers. The one who moves the box the other side first wins the game.



How is the propeller controlled by our mind?



here signals from emotiv headset reach your pc ,which are in turn is processed by emo composer and inputs are send to arduino through arduino IDE .Arduino convert the

input into corresponding pwm signals which are send to propeller as output.

The screenshot displays the Eclipse IDE interface. The main editor window shows the source code for `EmoStateLog.java`. The code includes several `System.out.println` statements for logging events like "RightWink", "LookingLeft", and "LookingRight". It also features conditional logic to check for specific states and handle errors, such as "Internal error in Emotiv Engine!".

```
System.out.println("RightWink");
if (EmoState.INSTANCE.ES_ExpressivIsLookingLeft(eState) == 1)
    System.out.println("LookingLeft");
if (EmoState.INSTANCE.ES_ExpressivIsLookingRight(eState) == 1)
    System.out.println("LookingRight");

System.out.print("concentration level: ");
System.out.println(EmoState.INSTANCE.ES_AffectivGetEngagementBor

    }
}
else if (state != EdkErrorCode.EDK_NO_EVENT.ToInt()) {
    System.out.println("Internal error in Emotiv Engine!");
    break;
}
}
```

The Console window at the bottom shows the execution output, including timestamps, user information, and the current state of the application:

```
EmoStateLog [Java Application] /usr/lib/jvm/java-6-openjdk-i386/bin/java (12-Jun-2013 6:13:00 PM)
concentration level: 0.0
0.25021502 : New EmoState from user 0
WirelessSignalStatus: 2
concentration level: 0.4
0.50043005 : New EmoState from user 0
WirelessSignalStatus: 2
concentration level: 0.3
0.7506451 : New EmoState from user 0
WirelessSignalStatus: 2
concentration level: 0.6
```

On the right side of the IDE, there is a control panel with various settings, including a "Send" button, "Auto Repeat" options, and numerical input fields for parameters like "Overall Skill" and "Long Term".

Below is the code for arduino uno:

```
#include "math.h"

char a;

const int analogOutPin_1 = 3; // Analog output pin that the LED is attached to

int b=0;

void setup() {

Serial.begin(9600); // set up Serial library at 9600 bps

pinMode(3,OUTPUT);

pinMode(8,OUTPUT);

pinMode(9,OUTPUT);

//pinMode(9,OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

digitalWrite(8,1);

digitalWrite(9 ,0 );

if(Serial.available(>0)

{ a = Serial.read();

if(a == 'a')

b=13;

if(a == 'b')

b=26;

if(a == 'c')

b=39;

if(a == 'd')

b=52;

if(a == 'e')

b=65;

if(a == 'f')

b=78;

if(a == 'g')
```

```
b=91;

if(a == 'h')

b=104;

if(a == 'i')

b=117;

if(a == 'j')

b=130;

if(a == 'k')

b=143;

if(a == 'l')

b=156;

if(a == 'm')

b=169;

if(a == 'n')

b=182;

if(a == 'o')

b=195;

if(a == 'p')

b=208;

if(a == 'q')

b=221;

if(a == 'r')

b=234;

if(a == 's')

b=247;

if(a == 't')

b=255;

analogWrite(analogOutPin_1, b);

}

}
```

Here every alphabet which is received from emocomposer is assigned a pwm value which is given to the output pin of arduino. This pwm value controls the speed of the propeller.

Useful links:

<http://playground.arduino.cc/Interfacing/Processing>

http://www.youtube.com/watch?v=qUI0qen3_-s

<http://arduinoasics.blogspot.in/2012/07/arduino-basics-simple-arduino-serial.html>

http://yuleblinker.com/silverlight_arduino/images/arduino_silverlight.jpg

<http://docs.oracle.com/javase/tutorial/networking/sockets/definition.html>

<http://docs.oracle.com/javase/tutorial/networking/sockets/readingWriting.html>

<http://docs.oracle.com/javase/tutorial/networking/sockets/clientServer.html>

Problems faced:

When we checked our arduino was giving correct output with DMM and led but when we attached propellers it was not working properly . Finally we get to know that the output voltage was correct but the current supply was not enough to run the propellers, so we used transistor for it.

A WORD OF THANKS

First of all I would like to thanks SHIVENDU BHUSHAN, SWAPNIL UPADHYAY, SONU AGARWAL and mentors of E-club for their support and mentorship. This project helped us experience new things that added to our knowledge .We just wanted to say how much we have enjoyed working on this summer project and how truly grateful we are for the opportunity given to us.....