**Multitouch Paper Recognising Gestures (with vvvv)**

*by Hannah and Joreg*

*PART 1 (12-14:30)* **Multitouch Paper** *or: how to translate world to computer*

- Intro to electricity, resistance - Translating resistance to voltage - Translating analog to digital - Build 3x3 matrix - Internal pull-up resistors - High-impedance INPUTs - Write code to parse the matrix

*PART 2 (15-17:30)* **Recognising Gestures** *or: a taste for visual programming with vvvv*

- Intro to vvvv - Recognising mouse gestures - Connecting matrix sensor - Using blob detection to get touch point - Replace mouse with touch data

*PART 1 (12-14:30)* **translating** world to computer

**sensing**

**ANALOG DIGITAL CONVERTER (ADC)**

**VOLTAGE DIVIDER**

**Voltage (V)** - is electrical pressure or force. Sometimes referred to as potential. Voltage drop is the difference in voltage between the two ends of a conductor through which current is flowing.

**Current (I)** - is the quantity of electronics passing a given point. The unit of current is Ampere. 1 Amp = 6,280,000,000,000,000,000 electronics passing a point in one second.

**Resistance (R)** - conductors are not perfect, they resist the flow of current to some degree. the unit of resistance is the Ohm (Ω).

Forrest M. Mims “Getting Started in Electronics”

*V*=IxR

*resistor variable resistor*

*potential divider*

**Velostat**

**Piezoresistance** *“***Piezo***”, derived from the Greek piezein, which means to squeeze or press*

**Tactile sensing in dexterous robot hands** *https://www.researchgate.net/publication/282557394\_Tactile\_sensing\_in\_dexterous\_ robot\_hands\_-\_Review/figures?lo=1*

*auto-ranging***MULTIMETERS***manual-range*

*light press: 1K Ohm hard press: 200 Ohm*

**13**

**24**

*light press: \_\_\_\_\_\_\_\_ Ohm*

*hard press: \_\_\_\_\_\_\_\_ Ohm*

*hard press: 50 Ohm light press: 300 Ohm*

**A+V**

للمش

**--**

**-**

**-**

**-**

**-**

**--**

**-**

**-**

**-**

**-**

**R1 V1**

**|**

**|**

**1**

**1**

**11**

**---**

**1**

**|**

**|**

**R2 v2.**

**-**

**-**

**-**

**-**

**5V**

**1K Ohm ? V**

**1K Ohm ? V**

**5V**

**1K Ohm 2.5 V**

**1K Ohm 2.5 V**

**5V**

**2K Ohm ? V**

**500 Ohm ? V**

**5V**

**2K Ohm 4 V**

**500 Ohm 1 V**

**-**

**-**

**-**

R2 v2

|

**-**

**--**

**--**

**-**

**-**

**-**

**-**

**||**

|

조 R1 v1

**5V**

*4.9* **1000 Ohm**

*4.8 V - 300 50 Ohm - 0.2 0.1 - V*

**5V**

**200 Ohm** *50 - 300 Ohm 1 -3 V4 - 2 V*

*light press: \_\_\_\_\_\_\_\_ V*

*hard press: \_\_\_\_\_\_\_\_ V*

**VOLTAGE DIVIDER PULL-UP RESISTOR**

**VARIBLE RESISTOR**

**QUESTIONS?**

**VOLTAGE DIVIDER**

**ANALOG TO DIGITAL CONVERSION**

**Arduino**

NEGA 328P-PUN

ATMEGA328

0 0907

**ADC**

HN

O O O NO

NW

AREF

GND

TEXT

BYL

DIGITAL - PWM

UNO

NO

RXISTE

ARDUINO

ICSP

OO

RESET EN

103

**=**

**=**

**=**

**=**

**=**

**=**

1820DST ATMEGA328P U AMEL 35473D

CS

25V

POWER

ANALOG IN

IOREF

RESET

17:N 3.3V

5V

GND

Vin

sensor

ха

pressure

RESET EN

OWER

AREF GND

AMEL 354730 ATMEGA328P U 1820DST

NALOG

***A*RDUTSA** OXUNO

DIGITAL - PWM

on

ARDUINO.CC

FON

RX 0

*open Arduino IDE...*

*open: File >> Examples >> Communication >> Graph*

*select Board*

*select Port*

*upload*

*open Serial Monitor*

**ANALOG DIGITAL CONVERTER (ADC)**

**ANALOG DIGITAL CONVERTER (ADC)**

**we are ALMOST ready to build the MATRIX!**

*EXTERNAL PULLUP INTERNAL PULLUP (20K)*

*20K Ohm*

*R*

pinMode*(A*O, INPUT\_PULLUP);

void setup() {

pinMode*(A*O, INPUT\_PULLUP); Serial.begin(9600);

*v*oid loop() {

Serial.println(analogRead*(*AO)); delay(2);

14TX

SA

AREF

GND

13

12

~11

~10

TAL (PWM

PWM

AREF

OND

DIGT

**111**

SOV

ARDUITNO

23R 54730

T16.000

NIN A4

A3

A5

352

35

i vin

GND

5V

GND

IOREF

RESET

3.3V

ANALOG IN

P*OWER*

pressure

sensor

*open Serial Monitor*

**5V 5V**

*EXTERNAL PULLUP INTERNAL PULLUP (20K)*

*4* **200 Ohm** *2 V -* **20K Ohm**

*4.9 5 - V*

*300 50 Ohm - 3 1 - V 300 50 Ohm - 0.1 0 - V*

**QUESTIONS?**

**NOW we are ready to build the MATRIX!**

**147**

**2 3**

**58**

**69**

*A*

*1 1 2 2 3 3*

**columns**

**s wor***B C*

*B C*

*A*

pinMode*(*AO, INPUT\_PULLUP);

**Γ**

**+ HWWW**

Serial.print(analogRead(AO));

**Τ**

**Τ**

**Γ**

pinMode(*A*O, INPUT\_PULLUP); pinMod*e(*A1, INPUT\_PULLUP);

Serial.print(analogRead(AO)); Serial.print("\t"); Serial.print(analogRead(A1));

pinMod*e(*AO, INPUT\_PULLUP); pinMode*(A*1, INPUT\_PULLUP); pinMode*(*A2, INPUT\_PULLUP);

Serial.print(analogRead*(A*O)); Serial.print("\t"); Serial.print(analogRead(A1)); Serial.print("\t"); Serial.println(analogRead*(*A3));

**w*w***

TIT

***A*.**

| pinMode(PIN#, OUTPUT); – digitalWrite(PIN#, LOW);

**|**

**A.**

I pinMode(PIN*#*, OUTPUT); = digitalWrite(PIN#, LOW);

Ž*A*.

**11**

*I*

*II* -ROW1- *IIIIIIII* pinMode(5, OUTPUT); I digitalWrite(5, LOW); + pinMode(6, INPUT);X pinMode(7, INPUT); X

*////// -A- ////////* pinMode*(*A*O*, INPUT\_PULLUP); pinMode*(*A1, INPUT); pinMode(A2, INPUT); Serial.print(analogRead*(*AO)); Serial.print("\t"); delay(10);

TALÀ

*////////* -B- *////////* pinMode*(*AO, INPUT); pinMode*(*A1, INPUT\_PULLUP); pinMode*(*A2, INPUT); Serial.print(analogRead*(A*1)); Serial.print("\t"); delay(10);

*/////// -C- ////////* pinMode*(A*O, INPUT); pinMode*(*A1, INPUT); pinMode*(*A2, INPUT\_PULLUP); Serial.print(analogRead(A2)); Serial.print("\t"); delay(10);

*III/II/I* -ROW1- *III ///* pinMode(5, OUTPUT); digitalWrite(5, LOW); pinMode(6, INPUT); pinMode(7, INPUT);

*/////// -*A*- ////////* pinMode(AO, INPUT\_PULLUP); + pinMode*(A*1, INPUT); X pinMode*(*A2, INPUT); X Serial.print(analogRead(AO)); Serial.print("\t"); delay(10);

**Ala**

ALÀ

*////////* -B- *////////* pinMode*(*AO, INPUT); pinMode*(A*1, INPUT\_PULLUP); pinMode*(*A2, INPUT); Serial.print(analogRead*(A*1)); Serial.print("\t"); delay(10);

*/////// -C- ////////* pinMode*(A*O, INPUT); pinMode*(*A1, INPUT); pinMode*(*A2, INPUT\_PULLUP); Serial.print(analogRead(A2)); Serial.print("\t"); delay(10);

*III/II/I* -ROW1- *III ///* pinMode(5, OUTPUT); digitalWrite(5, LOW); pinMode(6, INPUT); pinMode(7, INPUT);

*////// -A- ////////* pinMode*(*A*O*, INPUT\_PULLUP); pinMode(A1, INPUT); pinMode(A2, INPUT); Serial.print(analogRead(AO)); Serial.print("\t"); delay(10);

**Ala**

| Á

*///////*/ -B- *IIIIIII/* pinMode*(*AO, INPUT);X pinMode*(*A1, INPUT\_PULLUP); pinMode*(*A2, INPUT);X Serial.print(analogRead*(*A1)); Serial.print("\t"); delay(10);

*//////// -C- ////////* pinMode(*A*O, INPUT); pinMode*(*A1, INPUT); pinMode*(*A2, INPUT\_PULLUP); Serial.print(analogRead(A2)); Serial.print("\t"); delay(10);

*III/II/I* -ROW1- *III ///* pinMode(5, OUTPUT); digitalWrite(5, LOW); pinMode(6, INPUT); pinMode(7, INPUT);

*////// -A- ////////* pinMode*(*A*O*, INPUT\_PULLUP); pinMode(A1, INPUT); pinMode(A2, INPUT); Serial.print(analogRead(AO)); Serial.print("\t"); delay(10);

A

*////////* -B- *////////* pinMode*(*AO, INPUT); pinMode*(*A1, INPUT\_PULLUP); pinMode*(*A2, INPUT); Serial.print(analogRead*(A*1)); Serial.print("\t"); delay(10);

*////////* -(- *////////* pinMode(AO, INPUT); X pinMode*(*A1, INPUT); X pinMode(A2, INPUT\_PULLUP); + Serial.print(analogRead(A2)); Serial.print("\t"); delay(10);

*MUTITII* -ROW2- *////////* pinMode(5, INPUT); X pinMode(6, OUTPUT); I digitalWrite(6, LOW); + IpinMode*(*7, INPUT); X

*//////// -*A*- ///////* pinMo*de(*A*O*, INPUT\_PULLUP); pinMode*(*A1, INPUT); pinMode(A2, INPUT); Serial.print(analogRead*(A*O)); Serial.print("\t"); delay(10);

*I/III///* -B- *////////* pinMode*(*AO, INPUT); pinMode*(*A1, INPUT\_PULLUP); pinMode*(*A2, INPUT); Serial.print(analogRead*(A*1)); Serial.print("\t"); delay(10);

*/////// -C- ////////* pinMode*(A*O, INPUT); pinMode*(*A1, INPUT); pinMode*(*A2, INPUT\_PULLUP); Serial.print(analogRead(A2)); Serial.print("\t"); delay(10);

UD

AREF GND

DIGITAL - PWM

®

ARDUINO

Rx

CS

UNO..ON

POWER

ANAU

pressure matrix

A6

P

*A2*

**cheat-sheet**

**write the code to parse through the matrix........**

*Serial Monitor*

pressure matrix

**-**

**-**

**-**

**-**

**-**

**-**

**-**

**QUESTIONS?**

**- - - BREAK - - -**

*PART 2 (15-17:30)* **a taste for** visualprogramming with**vvvv**

**extra slides**

**cheat-sheet**