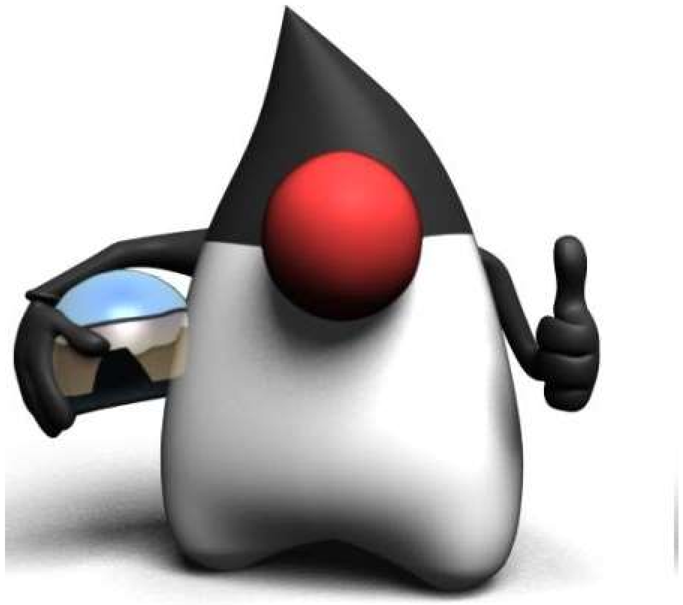
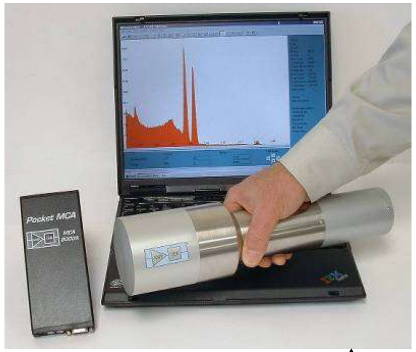
Android
-
Senzory



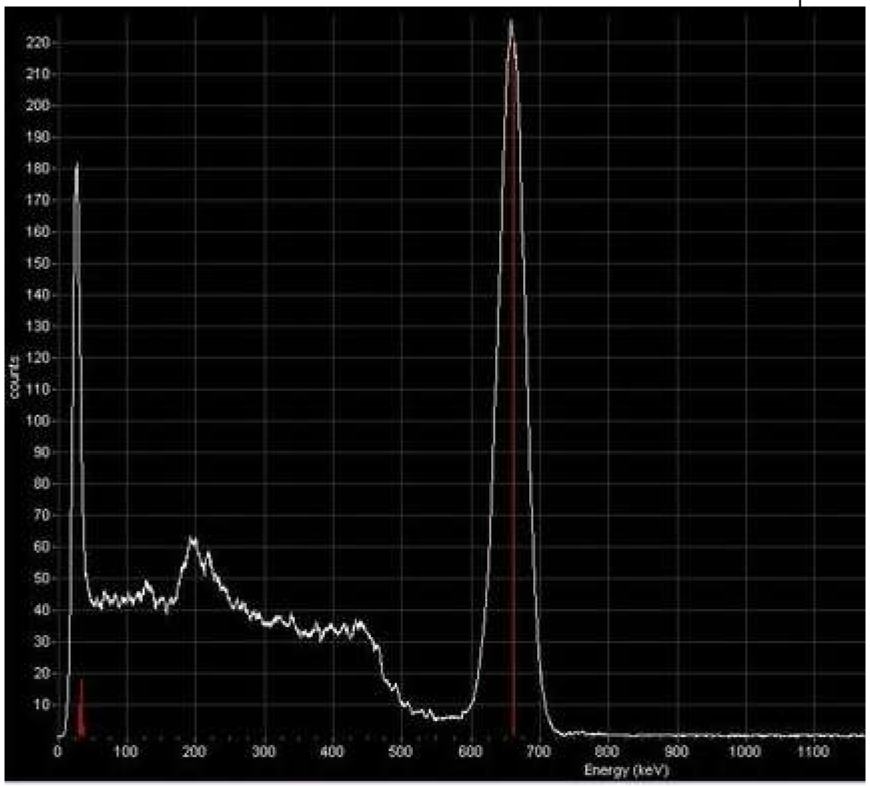
Peter Borovanský
KAI, I-18
MS-Teams: 2sf3ph4, List, github
borovan 'at' ii.fmph.uniba.sk

Senzor a čo s ním



- dnes to bude viac o fyzike, spracovávaní signálov, ako o programovaní
- pomerne jednoduché je mať hŕbu dát zo senzora, ale
  - čo znamenajú
  - aká je ich presnosť, odchylka, fyzikálna jednotka, ...
  - čo s nimi môžeme robiť
  - akú hodnotu má potom výsledok

Cézium 137



(niektorí) máme pomerne slabý zmysel pre interpretáciu/verifikáciu dát, meraní, či výsledkov výpočtov

(niektorí) informatici (často) sa uspokoja s akýmkoľvek výsledkom, ktorý typovo súhlasí, nepýtajú sa, či môže súvisieť s realitou

https://www.youtube.com/watch?v=T-bX0OjaxQM

Úvodný (fyzikálny) kvíz

Na úvod sa zamyslime, ako pristáva lietadlo ?
Na určenie výšky nad touch-down zónou používa:

- laserový diaľkomer,
- getAltitude z GPS senzora,
- výškomer,
- odhadom voľným okom ? resp. info veže, podľa výšky okolitých stromov,
- ultrasonický senzor vzdialenosti,
- tlakomer,
- barometer,
- altimeter ?
- pomocou lievika ?

Senzory

- v mobile/hodinkách – ľahko použiteľný
  - zaujímavá cena/výkon
  - limitovaná presnosť
- mimo mobilu – aká je komunikácia s ním

  viac o týždeň na prednáške
  - BT, Sériová linka (RS 232)
    - **HRM** (Heart rate Monitor)
    - **Glukomer** (diabetes)
    - **externé GPS**
    - Speedict – eBike (Volt/Amper)
    - **OBDII** - ELM 327 – diagnostika motora
    - Sensordrome
    - domáca meteo stanica - Netatmo
    - Gamma sonda
    - Blue Santa Claus – prekvapenie o týždeň
  - BT 4.0 (Low Energy)
  - ANT + ☹ 
  - nie BT, napr. cyklokomp (Wahoo), tu neurobíte nič ☹ ☹ ☹
  - uzavreté systémy, cadence senzor, Garmin, tu tiež neurobíte nič ☹☹☹☹

Tu roky vyhráva iPhone

HRM

GPS

E-bike

OBDII

Netatmo

Senzory

Skutočné (HW):

- LIGHT
- PROXIMITY
- PRESSURE
- AMBIENT_TEMPERATURE
- ACCELEROMETER
- GYROSCOPE
- MAGNETIC_FIELD
- RELATIVE_HUMIDITY (???)

Odvodené, syntetické (SW):

Hodnoty sa vypočítavajú z hodnôt iných:

- ROTATION_VECTOR
- LINEAR_ACCELERATION
- GRAVITY
- ORIENTATION

Výhoda odvodených SW-senzorov je, že

- integrujú
- korigujú

hodnoty viacerých HW senzorov

? S tlakomerom a teplomerom chceme urobiť z mobilnú meteo-stanicu ?

Aktuálne počasie (aj predpoveď) ľahšie dostanete z netu, ako by ste ho uhádli z dvoch hodnôt tlaku a teploty (aj to vo vašom vačku :-)

Ide o pomoc pre GPS, keďže GPS-nadm.výška má väčšiu chybu, ako tlakomer.

Lenže GPS potrebuje viditeľné satelity, tak barometer/altimeter treba kalibrovať

Základné senzory

- TYPE_LIGHT [lux]
- TYPE_PROXIMITY [cm]
- TYPE_RELATIVE_HUMIDITY [%]
- TYPE_AMBIENT_TEMPERATURE [ºC]
- TYPE_ACCELEROMETER [3axis, m/s^2]
- TYPE_GYROSCOPE [3axis, radian/s]
- TYPE_MAGNETIC_FIELD [μT]

LIGHT:

- `SensorManager.LIGHT_NO_MOON`: 0.001
- `SensorManager.LIGHT_FULLMOON`: 0.25
- `SensorManager.LIGHT_CLOUDY`: 100
- `SensorManager.LIGHT_SUNRISE`: 400
- `SensorManager.LIGHT_OVERCAST`: 10000
- `SensorManager.LIGHT_SHADE`: 20000
- `SensorManager.LIGHT_SUNLIGHT`: 110000
- `SensorManager.LIGHT_SUNLIGHT_MAX`: 120000

- Maximum range
- Minimum delay
- Name
- Power
- Resolution
- Type
- Vendor
- Version

Stav:

- `SensorManager.SENSOR_STATUS_ACCURACY_HIGH`
- `SensorManager.SENSOR_STATUS_ACCURACY_MEDIUM`
- `SensorManager.SENSOR_STATUS_ACCURACY_LOW`
- `SensorManager.SENSOR_STATUS_UNRELIABLE`

Frekvencia:

- `SENSOR_DELAY_FASTEST`
- `SENSOR_DELAY_GAME`
- `SENSOR_DELAY_UI`
- `SENSOR_DELAY_NORMAL`

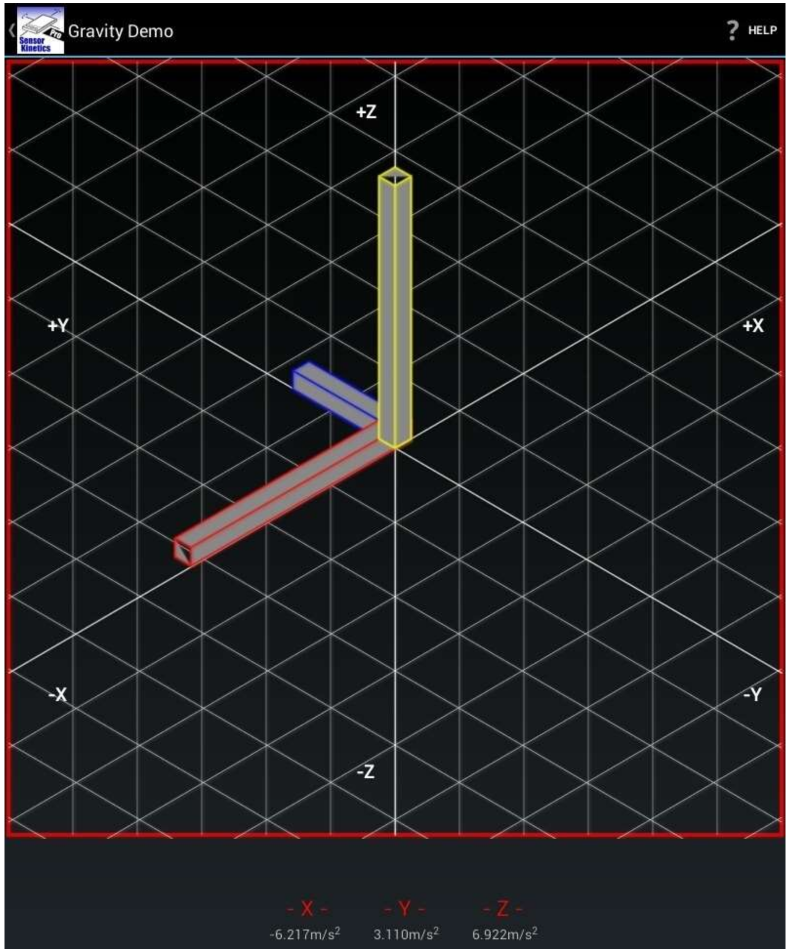
Physics Toolbox Accel/Gyro/Light/Magnet

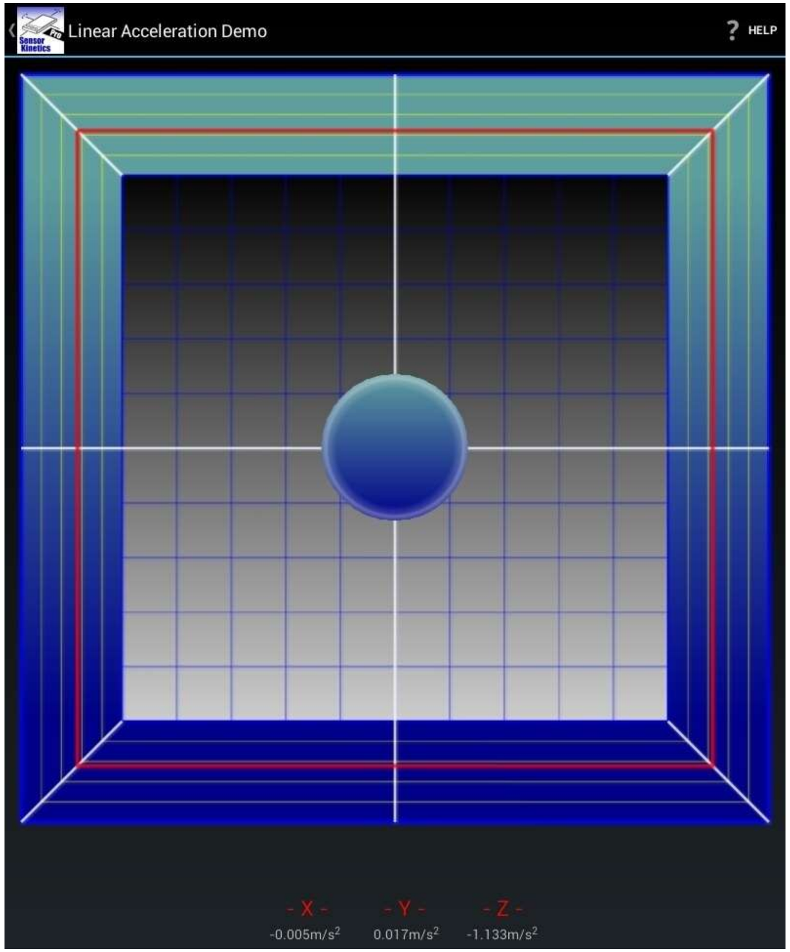
https://play.google.com/store/apps/details?id=com.chrystianvieyra.android.physicstoolboxaccelerometer

time,x,y,z ,
0.050, -0.00, 0.01, 1.04 ,
0.121, -0.00, 0.01, 1.04 ,
0.194, -0.00, 0.00, 1.04 ,
0.261, -0.00, 0.01, 1.04 ,
0.331, -0.01, 0.01, 1.04 ,
0.401, -0.01, 0.00, 1.04 ,
0.471, -0.00, 0.00, 1.04 ,
0.541, -0.00, 0.01, 1.04

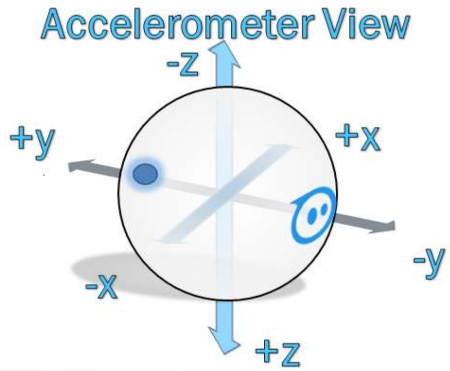
Sensor Kinetics

https://play.google.com/store/apps/details?id=com.innoventions.sensorkinetics

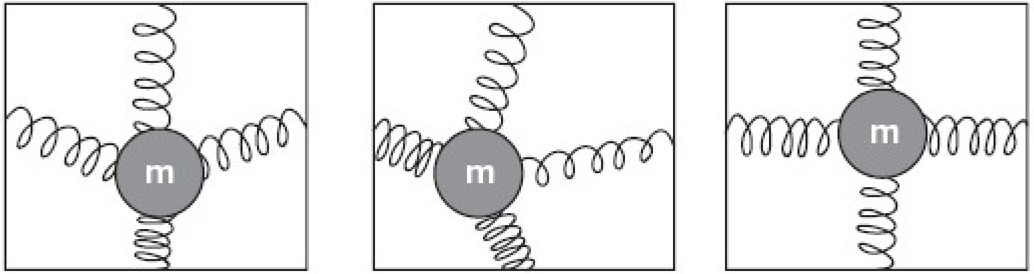




Akcelerometer [m/s^2]



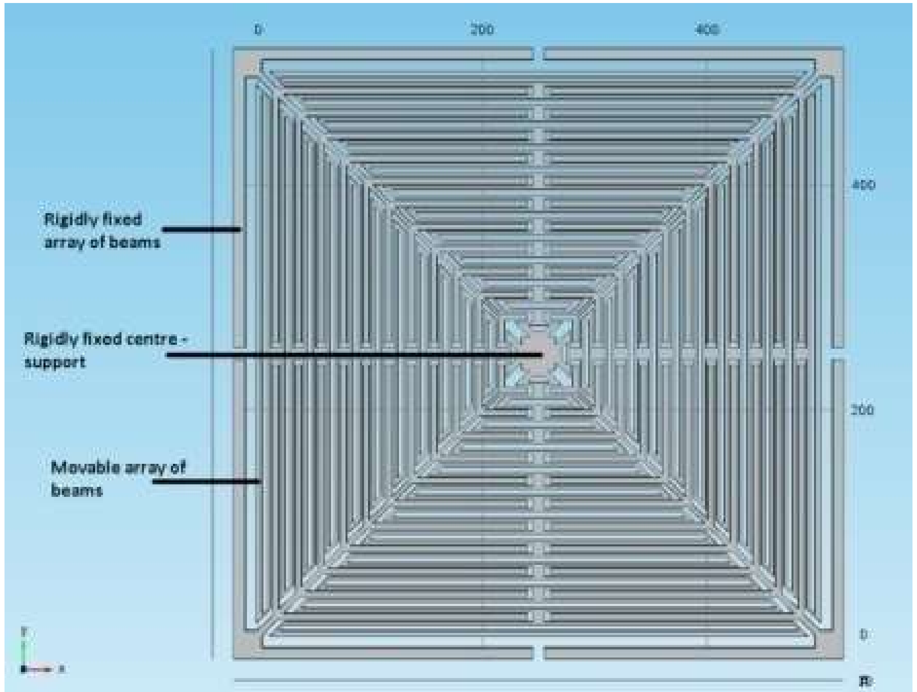
- meria zrýchlenie v smere osí (v kľude ukazuje x,y,z=0,0,1g=9.81)
- idea



- Realita

žiadne pružinky

ale silikón



Ako získať nejaké dáta

```java
public class SensorActivity extends Activity
    implements SensorEventListener {
    . . .
    SensorManager sm =
         (SensorManager)getSystemService(SENSOR_SERVICE);
    Sensor sAcc =
         sm.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
    if (sAcc != null) {
         tvInfo.setText( sAcc.getName()+
         ", maxRange:" + sAcc.getMaximumRange()+
         ", resol.:" + sAcc.getResolution()+
         ", minDelay:" +sAcc.getMinDelay());
    }    // zoznam všetkých accelerometrov
    List<Sensor> lstsen =
         sm.getSensorList(Sensor.TYPE_ACCELEROMETER);
    if (lstsen.size() > 0) ...
```

Project:SensorAccel.zip

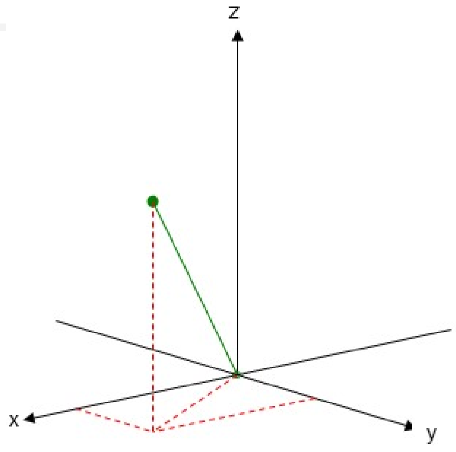
SensorListener

```
 public void onAccuracyChanged(Sensor sensor, int accuracy) {
     Log.d("SEN","accuracy:"+accuracy);
}
public void onSensorChanged(SensorEvent event) {
   if (startTime == 0) startTime = event.timestamp;
   tvX.setText(""+event.values[0]);      // 3axis sensor
   tvY.setText(""+(event.values[1])); // hodnoty values[0..2]
   tvZ.setText(""+(event.values[2]-SensorManager.GRAVITY_EARTH));
   tvTime.setText(""+(event.timestamp - startTime));
}
protected void onResume() {                   // registrujeme senzor
   sm.registerListener(this, sAcc, SensorManager.SENSOR_DELAY_UI);
}                                     // SensorManager.SENSOR_DELAY_GAME
protected void onPause() {
   sm.unregisterListener(this);   // odregistrujeme senzor
}
```

Project:SensorAccel.zip

Meranie zrýchlenia


```
public void onSensorChanged(SensorEvent event) {
currAcc = Math.round(
                Math.abs(
                    Math.sqrt(
                        Math.pow(event.values[0], 2)+
                        Math.pow(event.values[1], 2) +
                        Math.pow(event.values[2], 2))
                  - SensorManager.STANDARD_GRAVITY));

new Timer().scheduleAtFixedRate(new TimerTask() {
  public void run() {
    runOnUiThread(new Runnable() {
      public void run() {
        tvCurrAcc.setText(currAcc /9.81+ "g"); .invalidate();
        tvMaxAcc.setText(maxAcc /9.81+ "g"); .invalidate();}});}},
    0, 100);              // delay, period [ms]
```

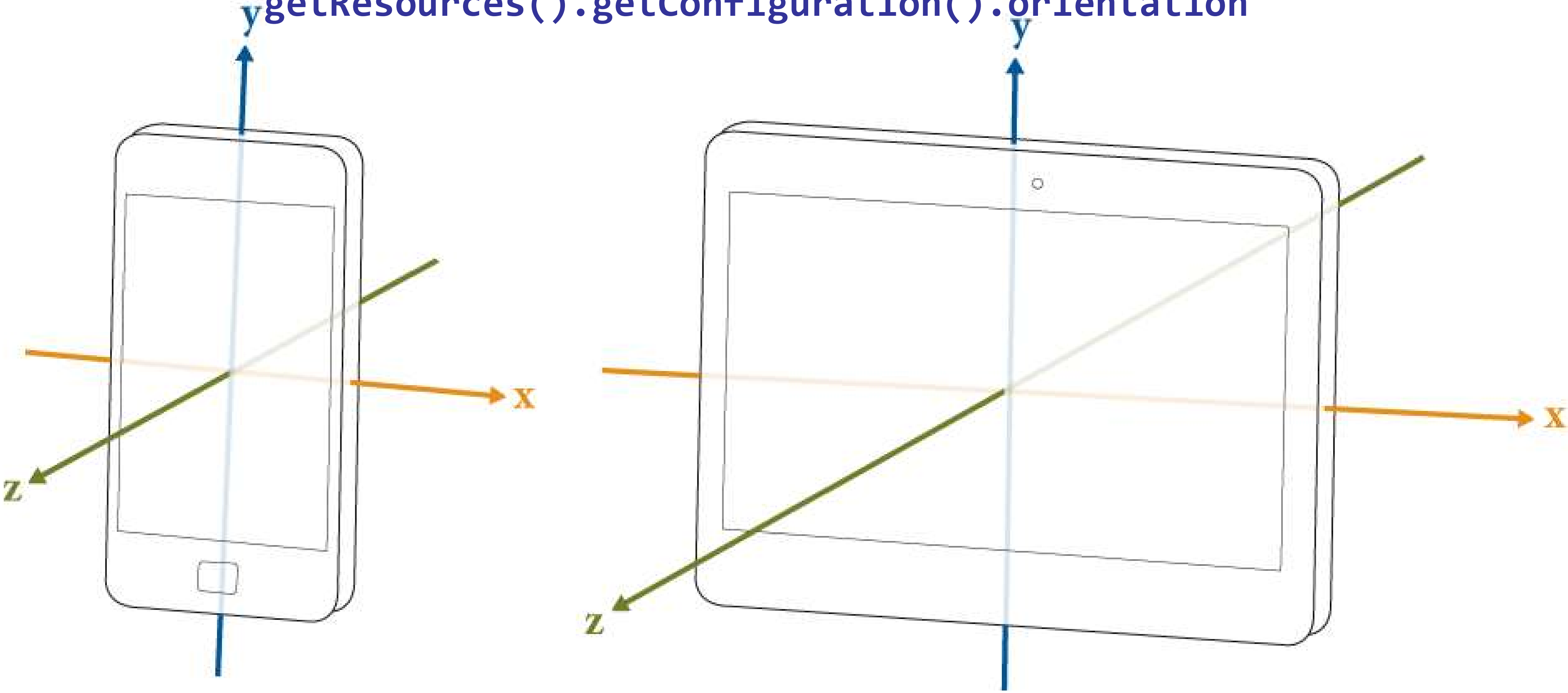


Project:SensorMaxAccel.zip

Screen, Sensor Orientation

Osi senzorov sa NEPRISPOSOBUJÚ orientácii aplikácie

`getResources().getConfiguration().orientation`



Project:SensorScreenOrientation.zip

Screen, Sensor Orientation

```
Display display
=((WindowManager)getSystemService(WINDOW_SERVICE)).getDefaultDisplay();
   int orientation = display.getOrientation();
   switch(orientation) {
     case Configuration.ORIENTATION_PORTRAIT:
        setRequestedOrientation
        (ActivityInfo.SCREEN_ORIENTATION_LANDSCAPE); break;
     case Configuration.ORIENTATION_LANDSCAPE:
       setRequestedOrientation
        (ActivityInfo.SCREEN_ORIENTATION_PORTRAIT); break;
   }
@Override
public void onConfigurationChanged(Configuration newConfig) {
  super.onConfigurationChanged(newConfig);
  if (newConfig.orientation == Configuration.ORIENTATION_LANDSCAPE) {
     Toast.makeText(this, "landscape", Toast.LENGTH_SHORT).show();
   } else ...
```

Project:SensorScreenOrientation.zip

A čo landscape/portrait display

AndroidManifest.xml:

- zakážeme ho:
  `android:screenOrientation="portrait"`
- povolíme všetky 4 orientácie display:
  `android:screenOrientation="fullSensor"`
- aby sa nám aplikácia neresetovala pri zmene zobrazenia:
  `android:configChanges="keyboardHidden|orientation|screenSize"`

```
public void onConfigurationChanged(Configuration newConfig) {
   super.onConfigurationChanged(newConfig);
   mScreenRotation =    // mScreenRotation * 90 degrees
        getWindowManager().getDefaultDisplay().getRotation();
}
```

ORIENTATION_PORTRAIT, ORIENTATION_LANDSCAPE
LANDSCAPE_REVERSE, PORTRAIT_REVERSE

Project:SensorSmallCompas.zip

Surface.Rotation
(azimut)

```
switch (mScreenRotation) {
   case Surface.ROTATION_0:
   break;
   case Surface.ROTATION_90:
        azimut+= Math.PI / 2f;
   break;
   case Surface.ROTATION_180:
        azimut= (float) (azimut > 0f ?
        (azimut- Math.PI) : (azimut+ Math.PI));
   break;
   case Surface.ROTATION_270:
        azimut-= Math.PI / 2;
   break;
}
```

bug (??) v prepínaní medzi 180<->0 a 270<->90

Project:SensorSmallCompas.zip

Surface.Rotation
(akcelerometer)

```
public void onSensorChanged(SensorEvent event) {
  if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER){
    switch (mDisplay.getRotation()) {
        case Surface.ROTATION_0:
            mSensorX = event.values[0];             // x     1  0
            mSensorY = event.values[1]; break;      // y     0  1
        case Surface.ROTATION_90:
            mSensorX = -event.values[1];            // -y    0 -1
            mSensorY = event.values[0]; break;      // x     1  0
        case Surface.ROTATION_180:
            mSensorX = -event.values[0];            // -x   -1  0
            mSensorY = -event.values[1]; break;     // -y    0 -1
        case Surface.ROTATION_270:
            mSensorX = event.values[1];             // y     0  1
            mSensorY = -event.values[0]; break;     //-x    -1  0
```

Project:SensorSmallCompas.zip

Prémia Vo výťahu
(súvisí s úvodným kvízom)

Za predpokladu, že máte v mobile tlakomer, resp. barometer na hodinkách, napíšte aplikáciu, ktorá zistí, na ktorom poschodí budovy sa nachádzate, ladiť vo výťahu ...

Potrebujete:

- odhadnúť výšku poschodia v budove, kde to ladíte (cca 3-5 metrov)
- mať možnosť základnej kalibrácie vašej apky, nastavenie tlaku, ...
- testability (matfyz 3.posch, FEI 7.-9.posch, STV 24.posch)

Hint (informatický):

Normálny atmosferický tlak je približne 1000 hPa (1bar, či 1.000 mbar)

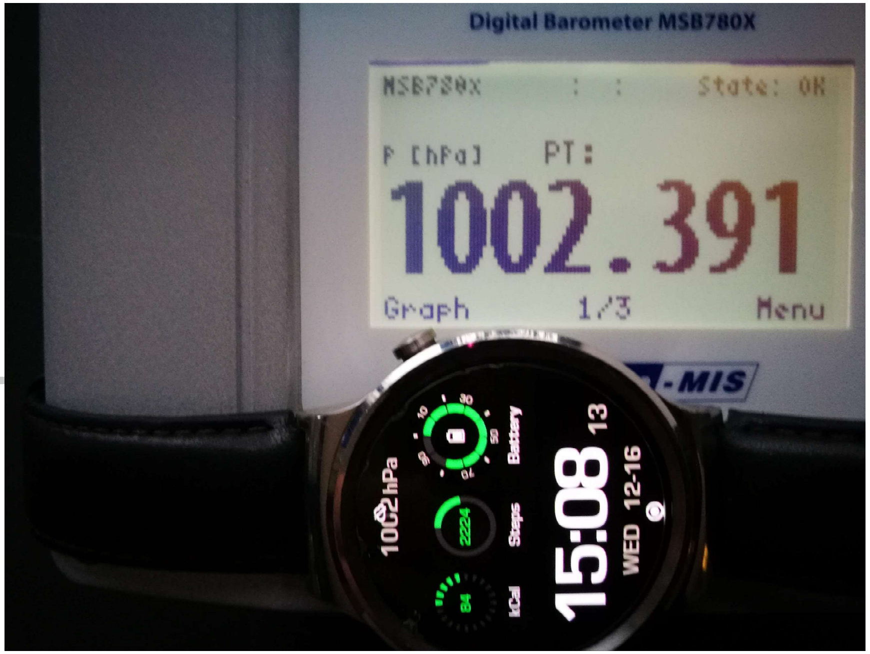
V bežných výškach (všetky budovy sveta) je tlak takmer lineárny od nadm.výšky teda len krátky čas (bez meteo-zmeny), resp. kým sa vyveziete výťahom ☺

Zjednodušene: 1000 Pa = 1kPa = 100 výškových metrov, 1hPa = 10 metrov.

Presnosť vášho mobilného tlakomera je rádovo 10 Pa=0.1hPa, t.j. cca 1 meter.

Presnosť leteckého tlakomera je v rádoch 1Pa, t.j. cca 10 cm.

Barometer


Digital Barometer MSB780X
State: OK
P [hPa]
PT:
1002.391
Graph
1/3
Menu
1002 hPa
Battery
Steps
2224
15:08
13
WED 12-16


MicroStep-MIS
999.238
Digital Barometer MSB780X

Atmosferický tlak

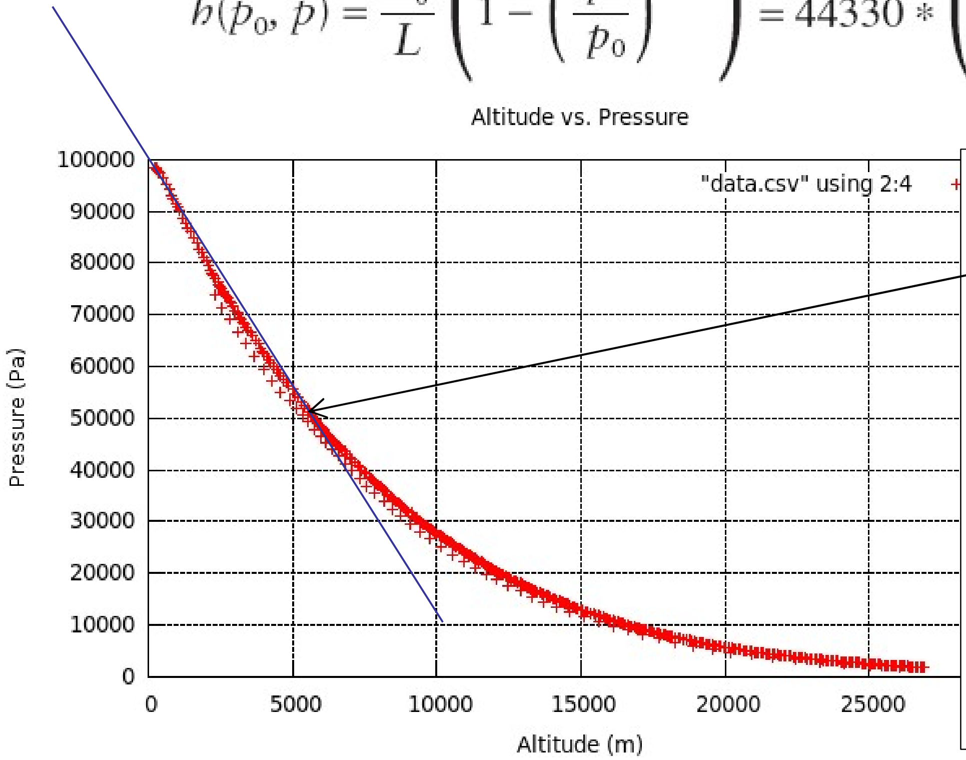
- TYPE_PRESSURE [1mbar=100Pa]

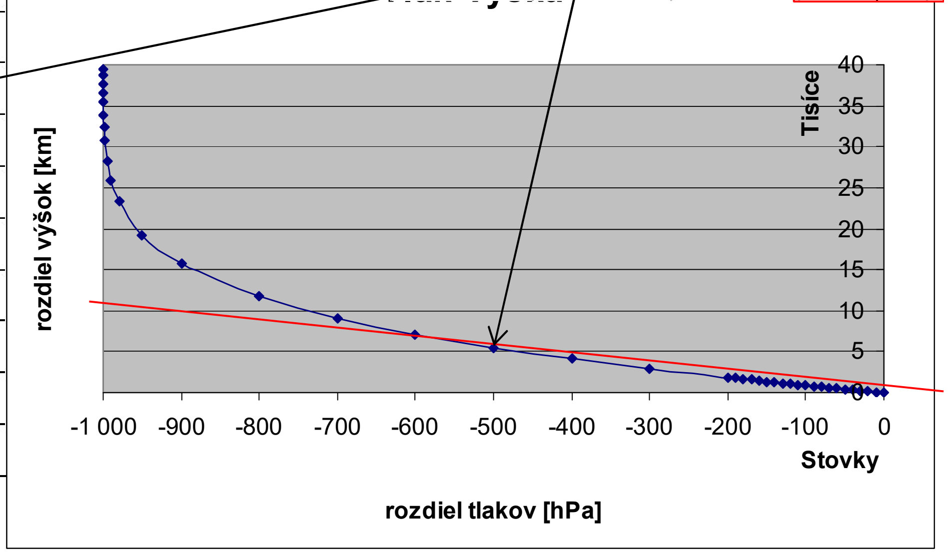
Výpočet výšky z rozdielu tlakov

$$h(p_0, p) = \frac{T_0}{L}\left(1-\left(\frac{p}{p_0}\right)^{\frac{RL}{gM}}\right) = 44330 * \left(1-\left(\frac{p}{p_0}\right)^{\frac{1}{5.255}}\right)$$

Project:tlak_vyska.xls

P	delta h	delta p	derivacia
100 000	0	0	
99 000	85	-1 000	-0,085
98 000	170	-2 000	-0,085
97 000	256	-3 000	-0,085
96 000	343	-4 000	-0,086
95 000	431	-5 000	-0,086
94 000	519	-6 000	-0,086
93 000	608	-7 000	-0,087
92 000	698	-8 000	-0,087
91 000	788	-9 000	-0,088
90 000	880	-10 000	-0,088
89 000	972	-11 000	-0,088
88 000	1 065	-12 000	-0,089
87 000	1 159	-13 000	-0,089
86 000	1 254	-14 000	-0,090
85 000	1 350	-15 000	-0,090
84 000	1 447	-16 000	-0,090
83 000	**1 544**	**-17 000**	-0,091
82 000	1 643	-18 000	-0,091
81 000	1 742	-19 000	-0,092
80 000	1 843	-20 000	-0,092
70 000	2 909	-30 000	-0,097
60 000	4 106	-40 000	-0,103
50 000	5 478	-50 000	-0,110
40 000	7 093	-60 000	-0,118



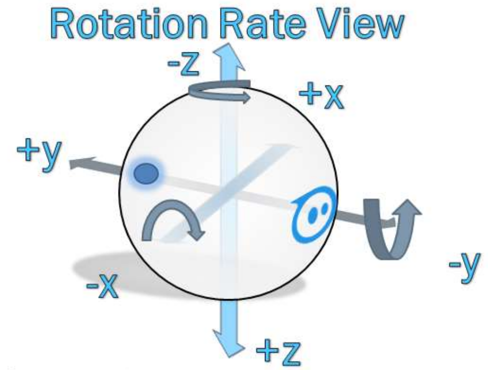


Tu už to nie je „lineárna“ fcia

ak dosadíte do vzorca, vo výške 39km je tlak 1-2 Pa, prakticky nič...

5	37 597
2	38 674
1	39 373

Gyroskop [radian/s]



- meria otáčanie okolo osí (v kľude ukazuje 0,0,0)
- funguje na princípe Coriolisovej sily

http://www.youtube.com/watch?v=_36MiCUS1ro
http://www.youtube.com/watch?v=Keyx8FwxnyM
fake

Katarina,USA

Austrália

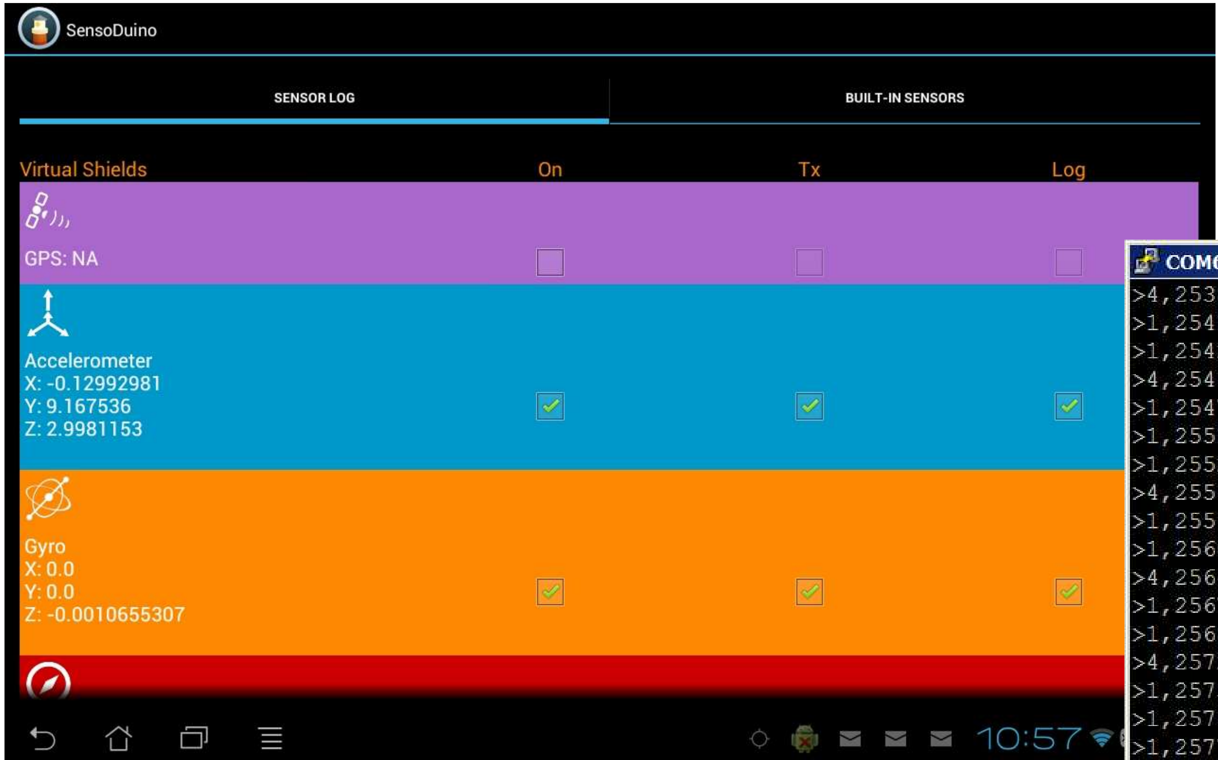
fake

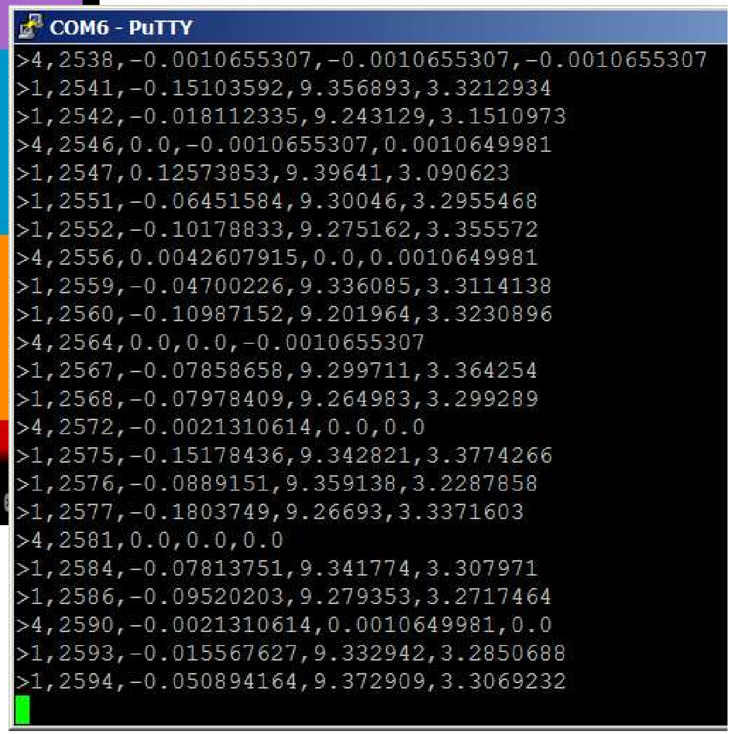
06x16 Bart vs.Austrálie

Prenos dát zo senzora

(cez BT/seriový com port)

- Bluetooth – Sensoduino výstup do BT, čo je v PC mapované na virtualny sériový port, napr. com11



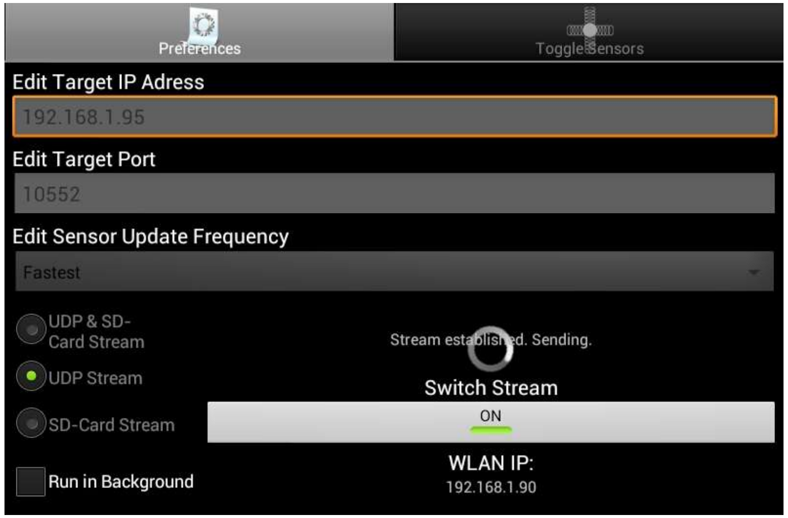


https://play.google.com/store/apps/details?id=com.techbitar.android.sensoduino&rdid=com.techbitar.android.sensoduino

Sensorstream
(cez TCP/IP, UDP)

- https://play.google.com/store/apps/details?id=de.lorenz_fenster.sensorstreamgps

```
import socket, traceback

host = ''
port = 5555
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
s.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
s.bind((host, port))

while 1:
    try:
        message, address = s.recvfrom(8192)
        print (message)
    except (KeyboardInterrupt, SystemExit):
        raise
    except:
        traceback.print_exc()
```



Sensorstream

(cez TCP/IP, UDP)

- https://play.google.com/store/apps/details?id=de.lorenz_fenster.sensorstreamgps

```
import socket, traceback

host = ''
port = 5555
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
```

Python 3.4.1 Shell

File Edit Shell Debug Options Windows Help

```
b'22168.51482, 3,   0.005,  9.150,  3.788, 4,  -0.004,  0.000,  0.000, 5, -22.906,-39.000,110.500'
b'22168.58437, 3,   0.035,  9.107,  3.795, 4,   0.006,  0.001,  0.000, 5, -22.906,-38.990,110.583'
b'22168.65526, 3,   0.062,  9.181,  3.761, 4,  -0.005,  0.000,  0.000, 5, -23.021,-39.010,110.448'
b'22168.72502, 3,   0.044,  9.158,  3.744, 4,   0.005,  0.000, -0.001, 5, -23.135,-39.010,110.500'
b'22168.79435, 3,   0.040,  9.176,  3.728, 4,  -0.005,  0.000, -0.002, 5, -23.135,-39.010,110.469'
b'22168.86769, 3,   0.043,  9.173,  3.693, 4,   0.003,  0.000,  0.001, 5, -22.969,-39.000,110.354'
b'22168.93447, 3,   0.034,  9.153,  3.707, 4,  -0.004,  0.000, -0.001, 5, -22.958,-39.000,110.385'
b'22169.00518, 3,   0.025,  9.156,  3.738, 4,   0.004,  0.000,  0.001, 5, -23.177,-39.000,110.417'
b'22169.14535, 3,   0.066,  9.144,  3.733, 4,   0.004, -0.001,  0.000, 5, -23.208,-38.969,110.458'
b'22169.21518, 3,   0.029,  9.154,  3.748, 4,  -0.003,  0.000,  0.002, 5, -23.302,-38.948,110.490'
b'22169.28522, 3,   0.014,  9.128,  3.789, 4,   0.000,  0.000,  0.001, 5, -23.281,-38.917,110.490'
b'22169.36171, 3,   0.023,  9.121,  3.732, 4,   0.001, -0.001,  0.002, 5, -23.333,-38.875,110.469'
b'22169.42470, 3,   0.031,  9.122,  3.743, 4,   0.000, -0.001,  0.000, 5, -23.510,-38.844,110.427'
b'22169.49543, 3,   0.021,  9.137,  3.814, 4,   0.001, -0.001,  0.001, 5, -23.573,-38.844,110.594'
b'22169.56537, 3,   0.014,  9.160,  3.746, 4,  -0.001,  0.001,  0.001, 5, -23.594,-38.844,110.781'
b'22169.63558, 3,  -0.013,  9.130,  3.772, 4,   0.005,  0.000,  0.000, 5, -23.656,-38.906,110.750'
b'22169.70542, 3,   0.013,  9.137,  3.777, 4,  -0.003, -0.001, -0.001, 5, -23.677,-38.948,110.781'
```

Lokalizácia akcelerometrom

- predstavte si, že máme dáta zo všetkých senzorov, okrem GPS.
- akceleračný (x,y,z)
- magneto / orientácia – kompas
- gyroskop
- čas

dá sa lokalizovať objekt/trasa z tracklogu ?

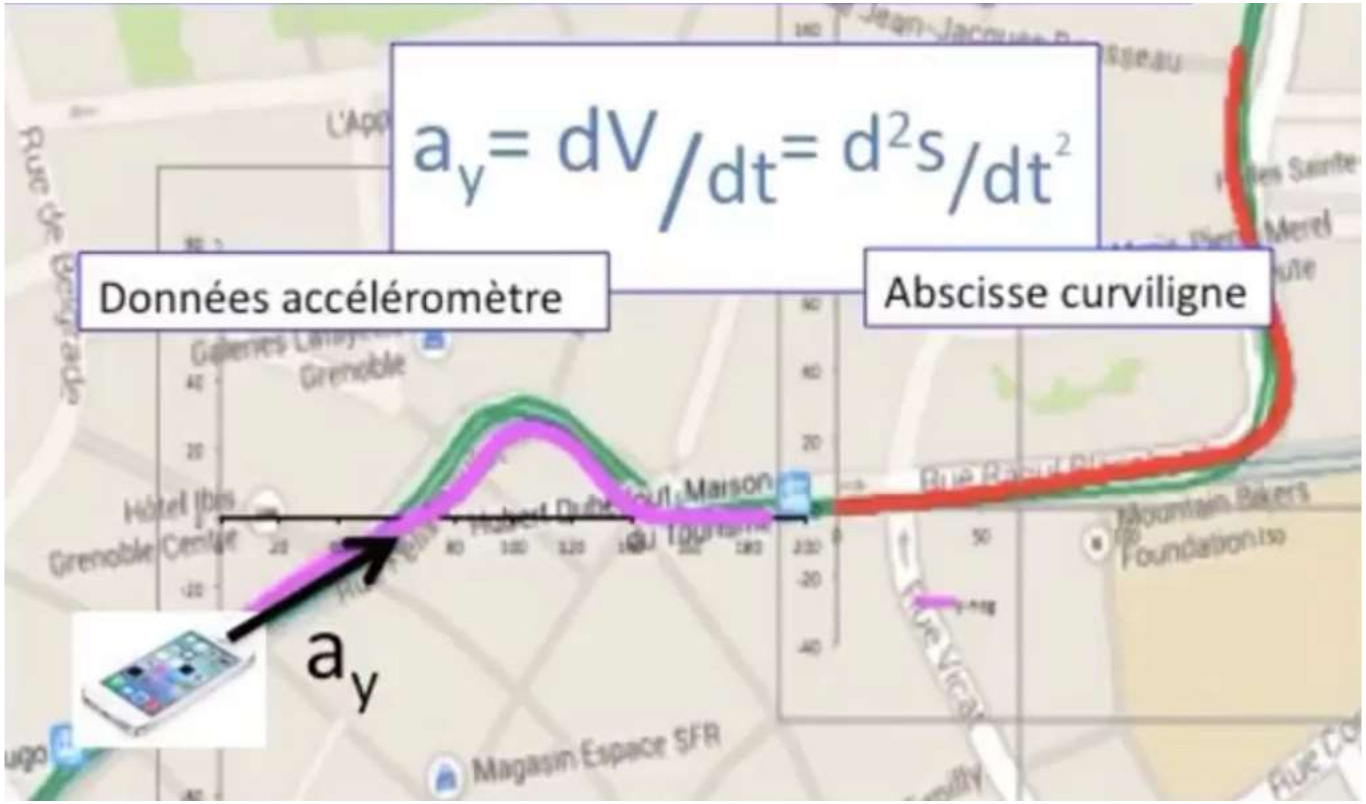
použil som aplikáciu Sensoduino- ~13min. log údajov pri ceste z domu do školy
http://dai.fmph.uniba.sk/courses/VMA/android/04Sensors/sensoduino/

je možné:

- rýchlosť či prejdenú dráhu v čase ? zistiť, ako ďaleko bývam ?
- zistiť, kde bývam ? … smer ?
- kedy oslavujem narodeniny ?

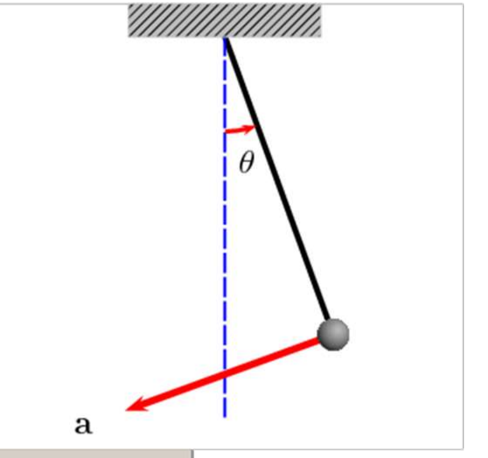
Tramway trajectory

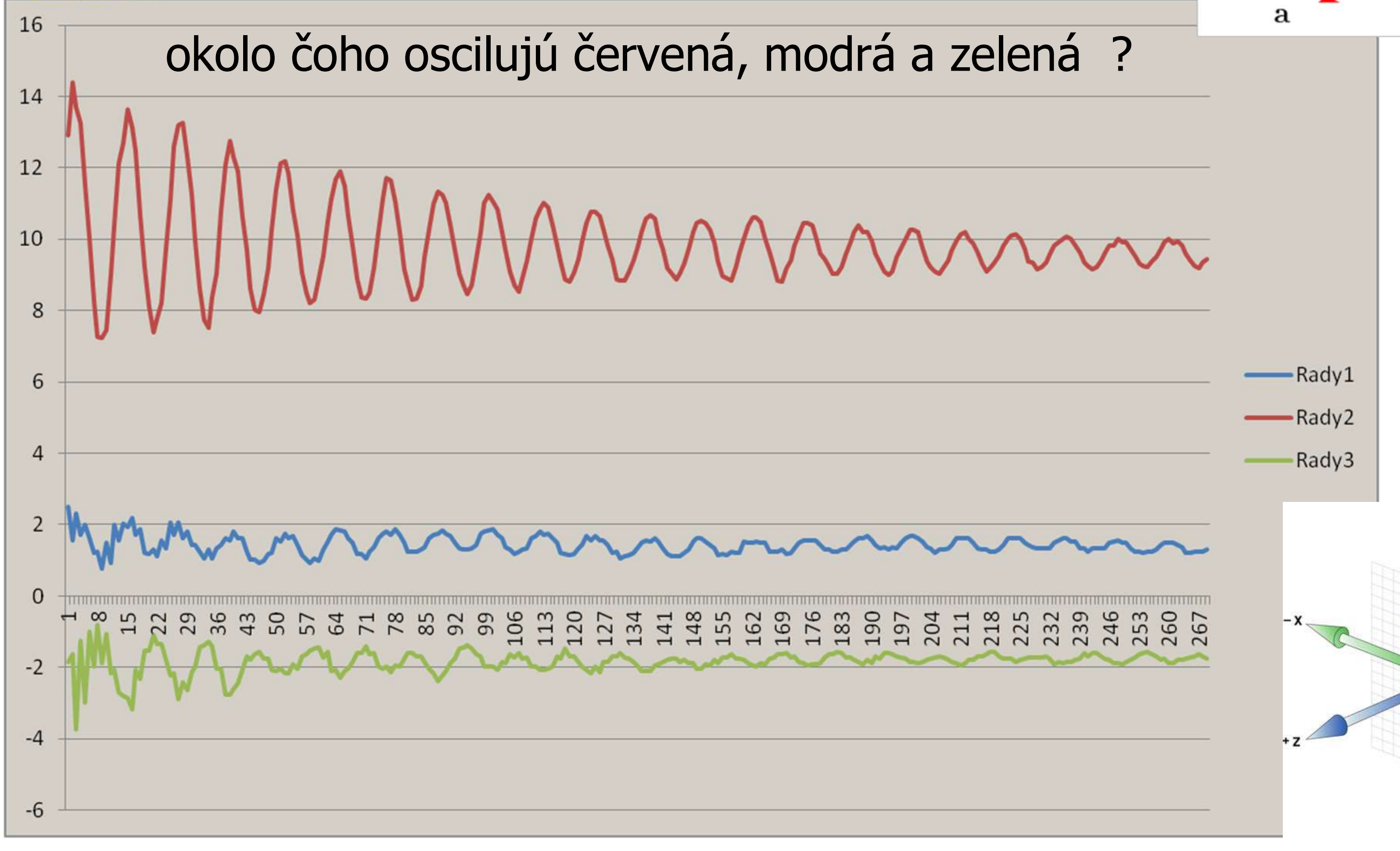
- https://sites.google.com/site/iprofmeca/session-8-tramway-trajectory
- jedna zastávka električkou (290m), iPhone bez kalibrácie senzorov
- zelená je google maps, červená/fialová sú aproximácie akcelerometrom



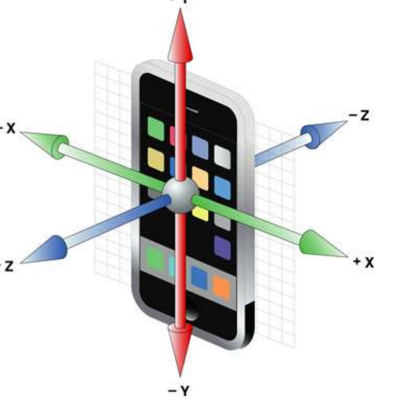
Pendulum

(akceleračný senzor – hw senzor)





Zelená a modrá sú vymenené oproti grafu

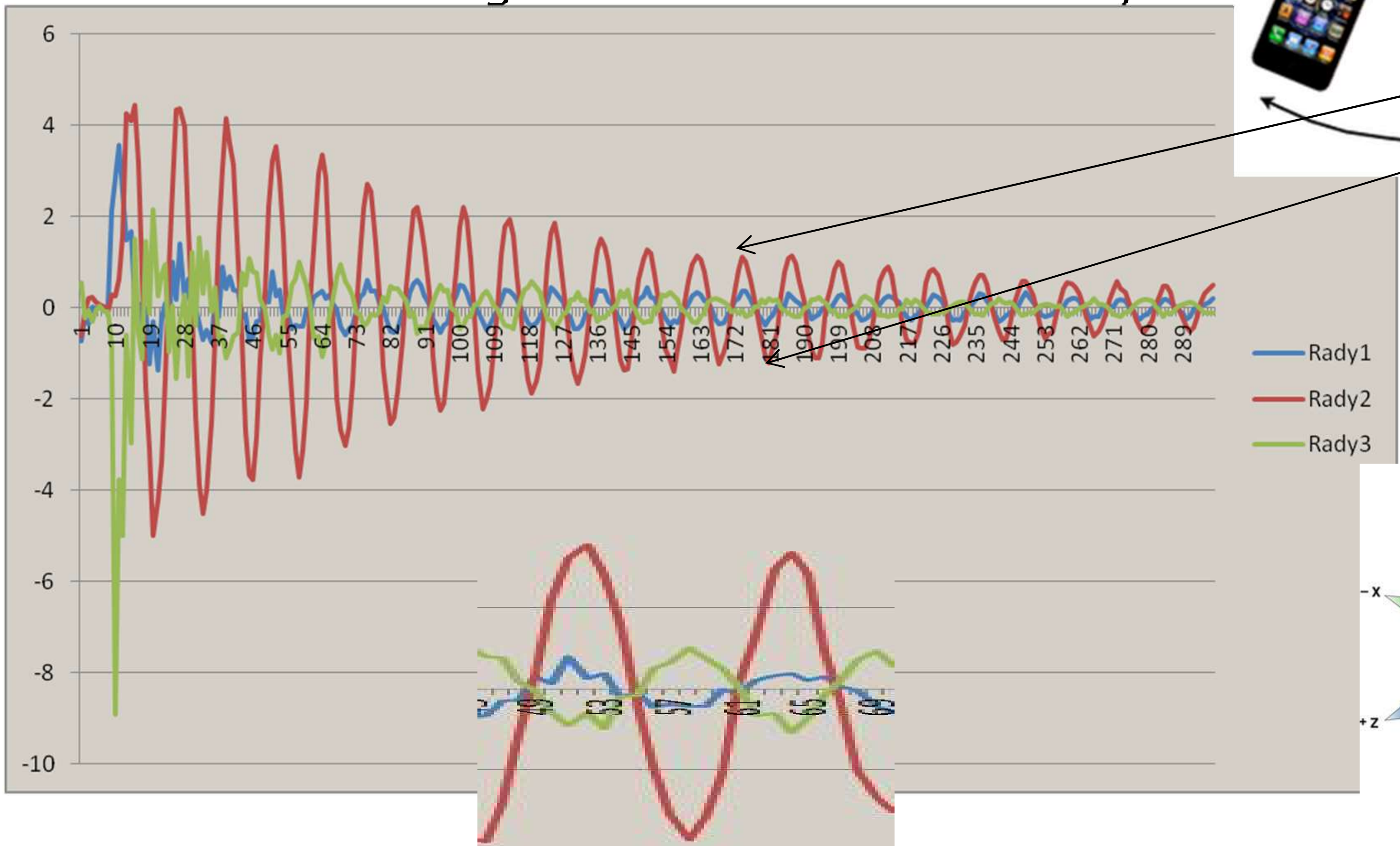


https://commons.wikimedia.org/wiki/File:Oscillating_pendulum.gif

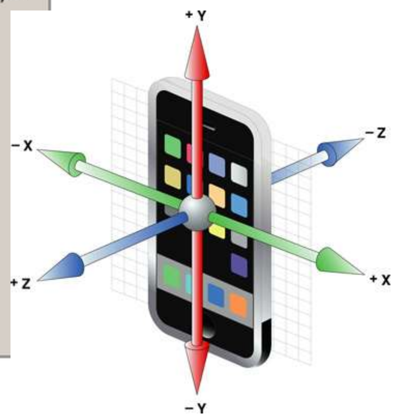
Pendulum

(lineárna akcelerácia – sw senzor)

- odstránená gravitačná konštanta 9.81 m/s^2



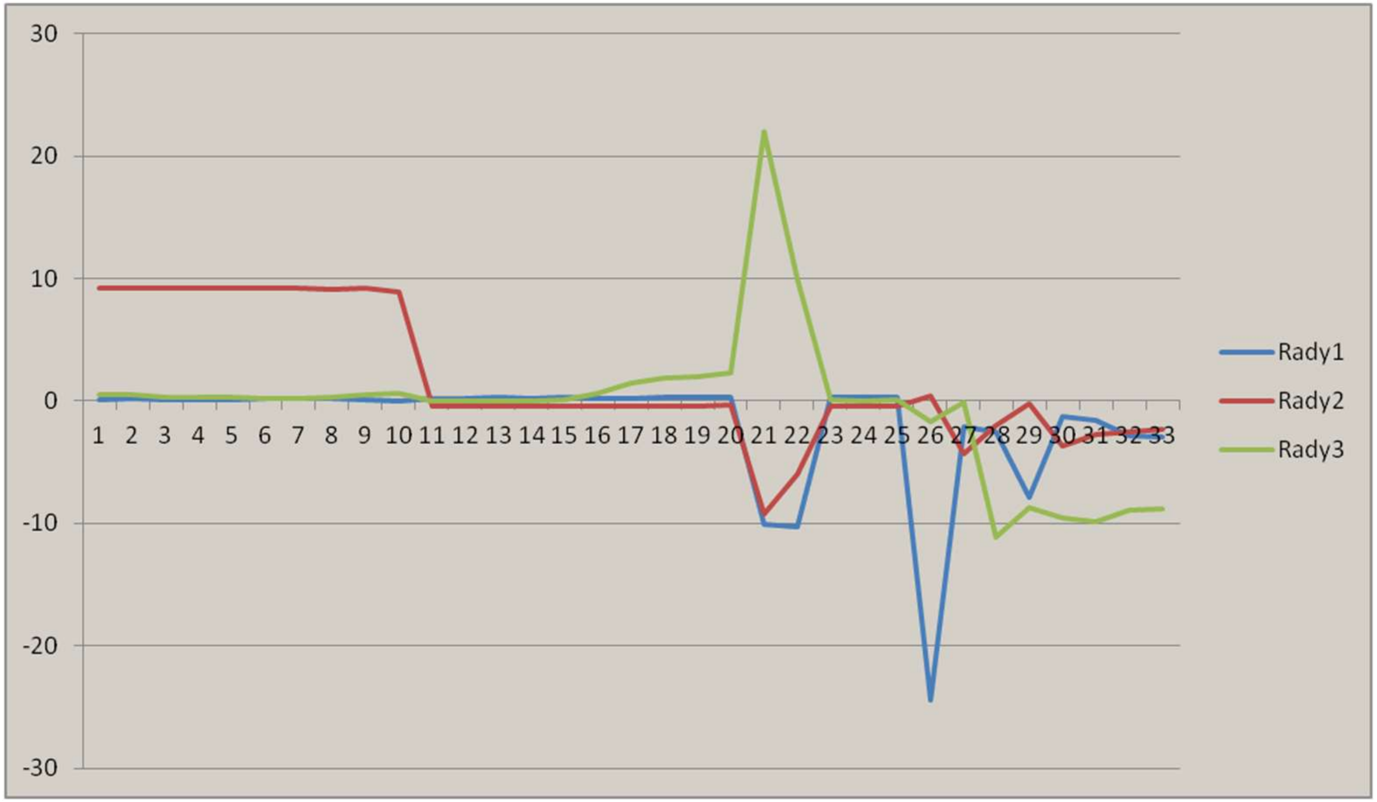
Zelená a modrá sú vymenené oproti grafu



https://commons.wikimedia.org/wiki/File:Oscillating_pendulum.gif

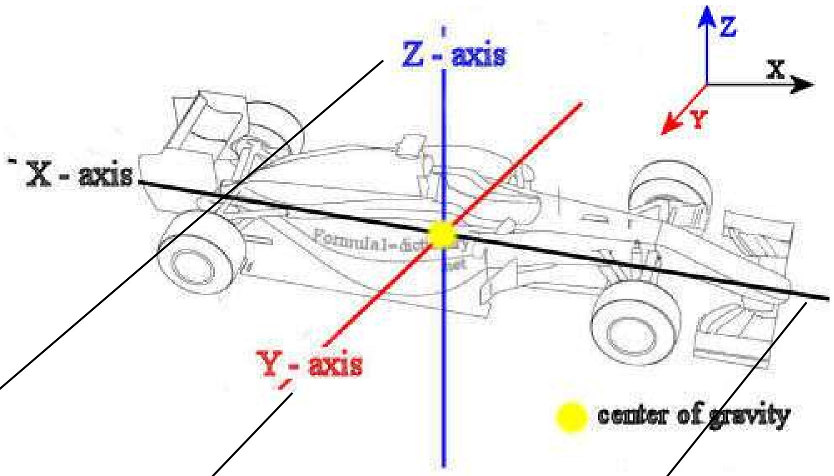
Voľný pád

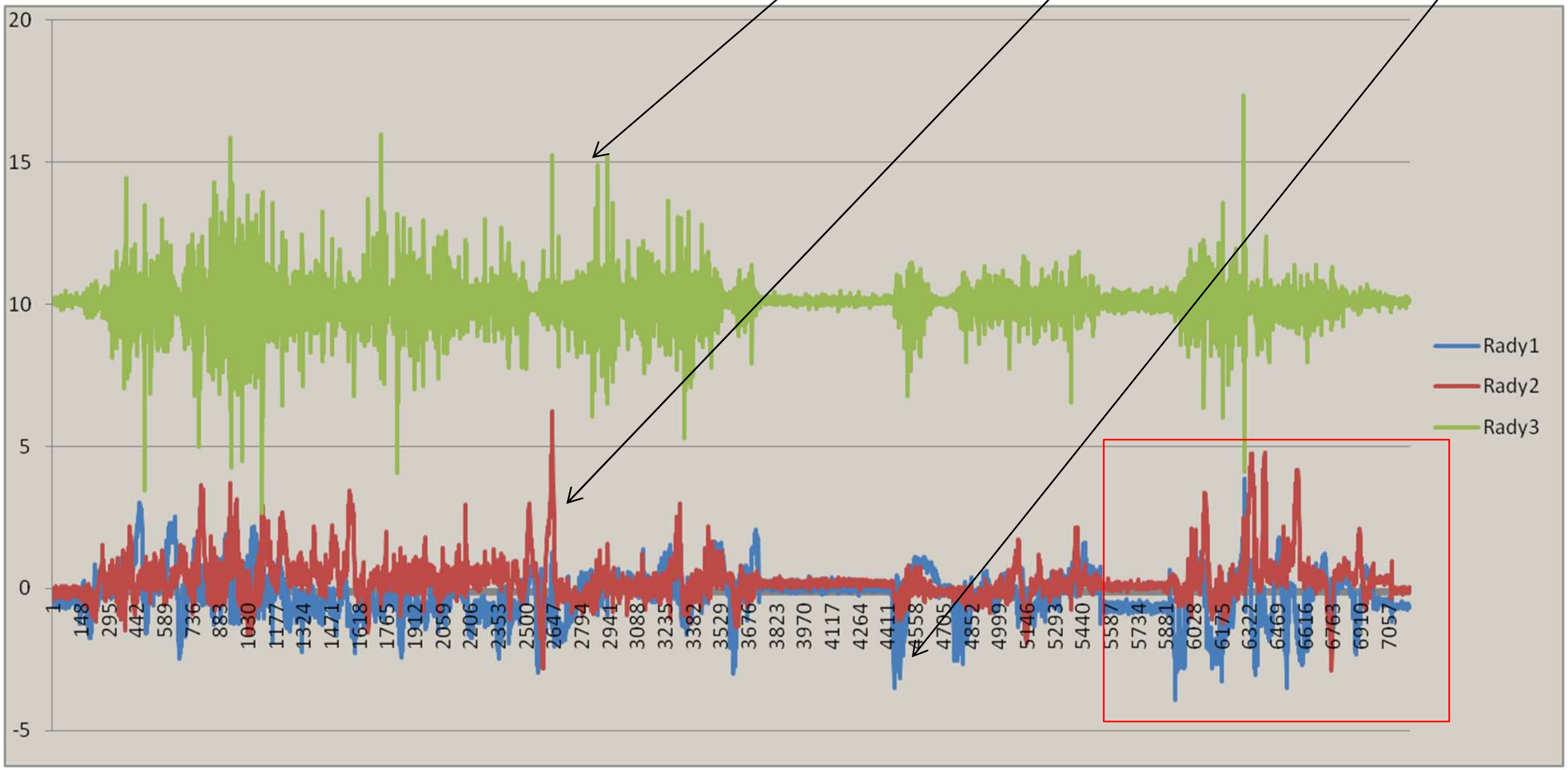
- Ak g=9.81m/s², po 1 s má mobil rýchlosť 9.81 m/s, ~35km/h
- priemerná rýchlosť je cca 4.9 m/s, takže preletí asi 4.9 m, ~ 1.5 poschodia
- dôležitá je preto vzorkovacia frekvencia, zvolíme 100Hz, t.j. 100 dát za 1s.



Acceleration

- analýza dát akcelerometra



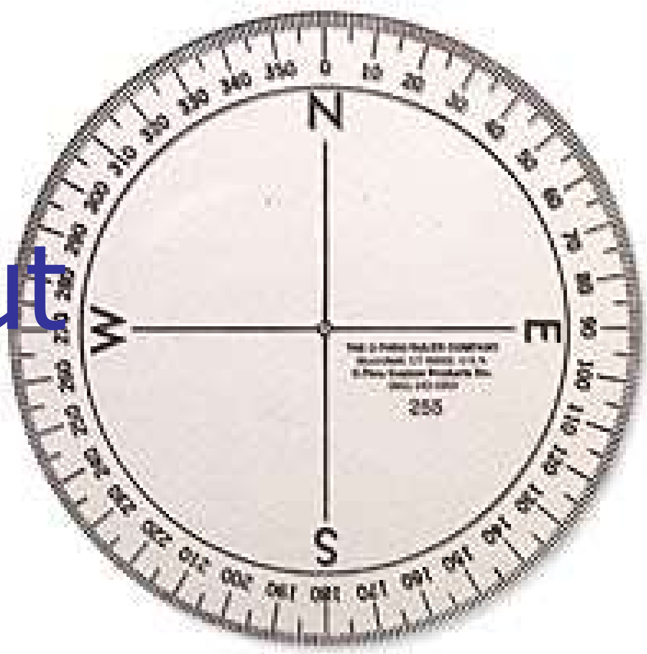


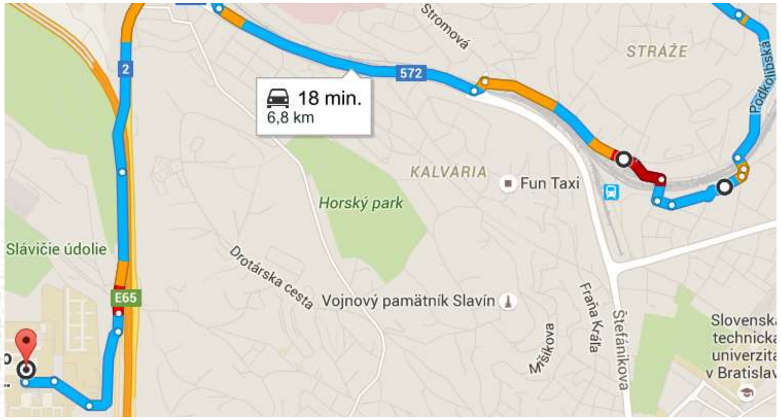
Magnetometer

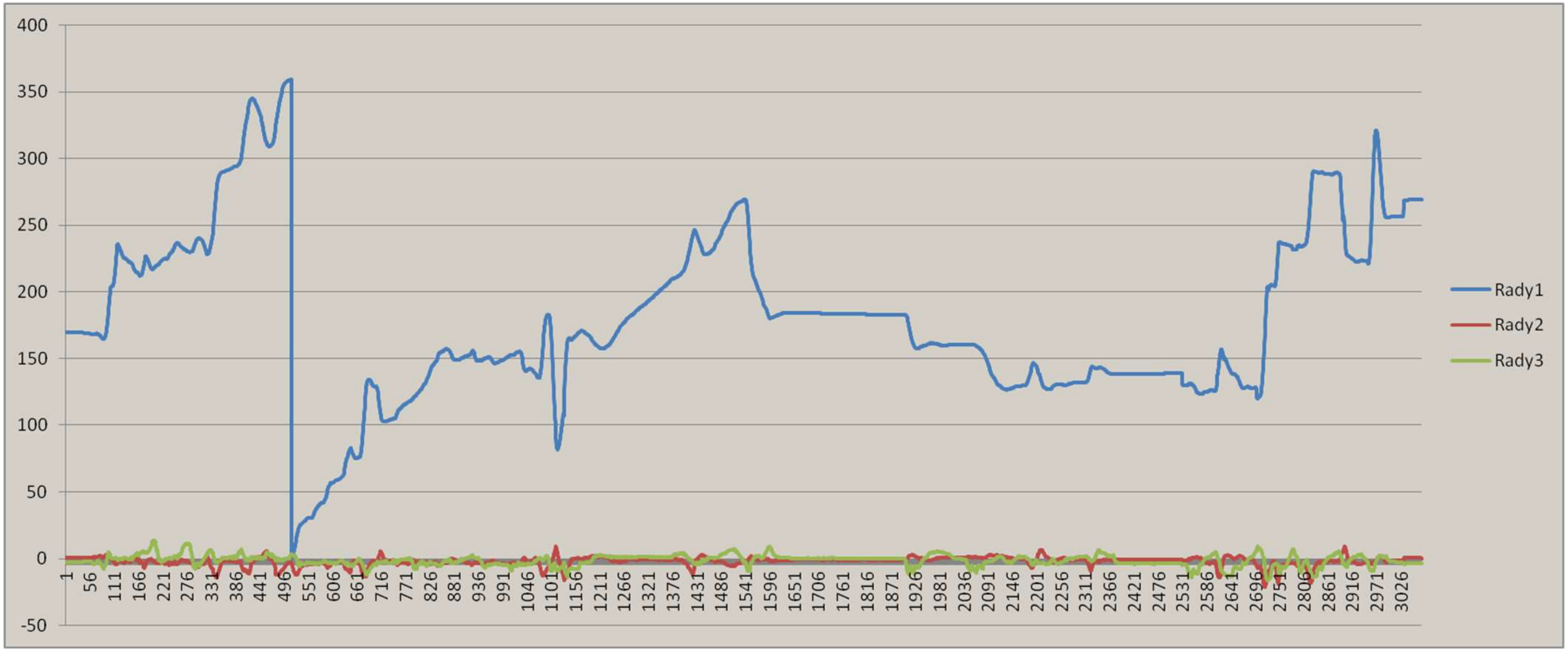
- analýza dát kompasu - to je hw senzor, ktorý meria silu mag.poľa v troch súradniciach, ale my by sme chceli azimut...

Azimut

Analýza dát orientačného senzora







Akcelerometer - rýchlosť [m/s]

- zrýchlenie je derivácia (zmena) rýchlosti

  takže
- zo zrýchlenia vieme určiť rýchlosť
  - integrovaním
  - až na konštantu

začneme pokus nulovou rýchlosťou akcelerometra

$$y = \int v\,dt$$

$$= \int (v_0 + at)\,dt$$

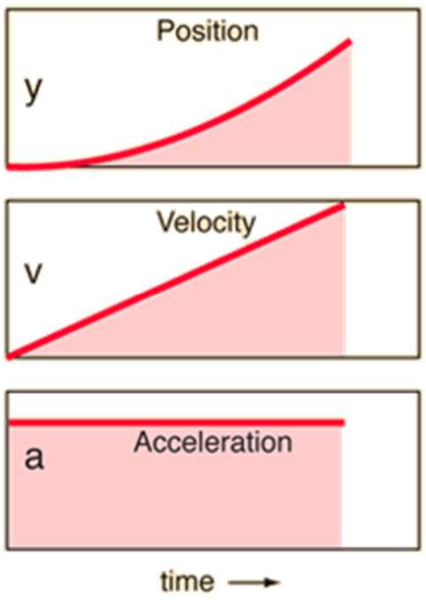
$$y = y_0 + v_0 t + \frac{1}{2}at^2$$

Integrate velocity to get position

$$v = \int a\,dt = v_0 + at$$

Integrate acceleration to get velocity

$$a = \text{constant}$$



Motion relationships in one dimension.

$$y = y_0 + v_0 t + \frac{1}{2}at^2$$

Derivative of position is velocity

$$v = \frac{dy}{dt}$$

$$v = v_0 + at$$

Derivitive of velocity is acceleration

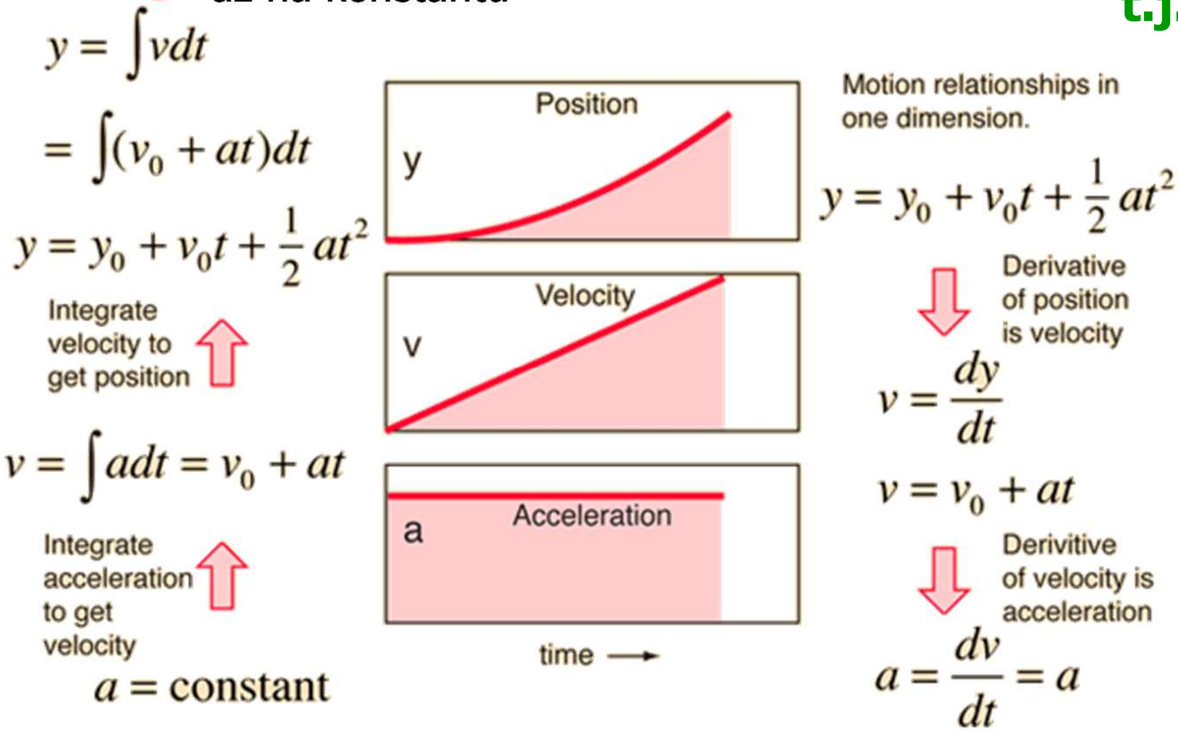
$$a = \frac{dv}{dt} = a$$

Akcelerometer - dráha [m]

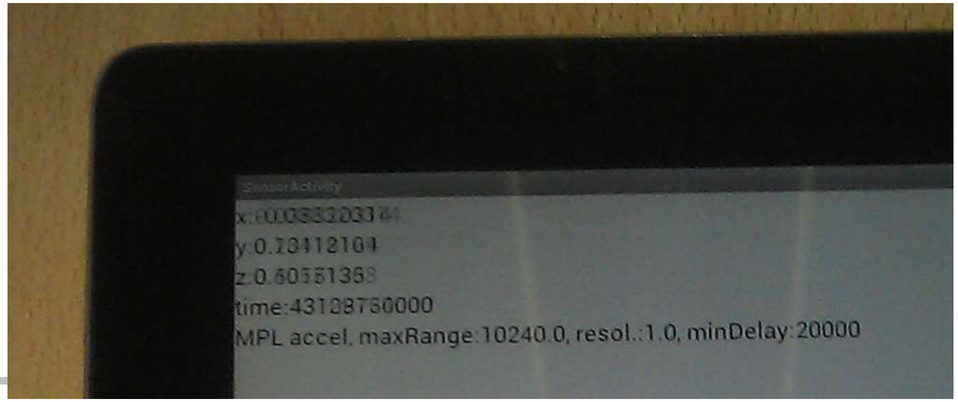
- rýchlosť je derivácia (zmena) prejdenej dráhy

  takže
- zo rýchlosti vieme určiť prejdenú dráhu
  - integrovaním
  - až na konštantu

začneme pokus nulovou dráhou, t.j. na štarte



Data Dump



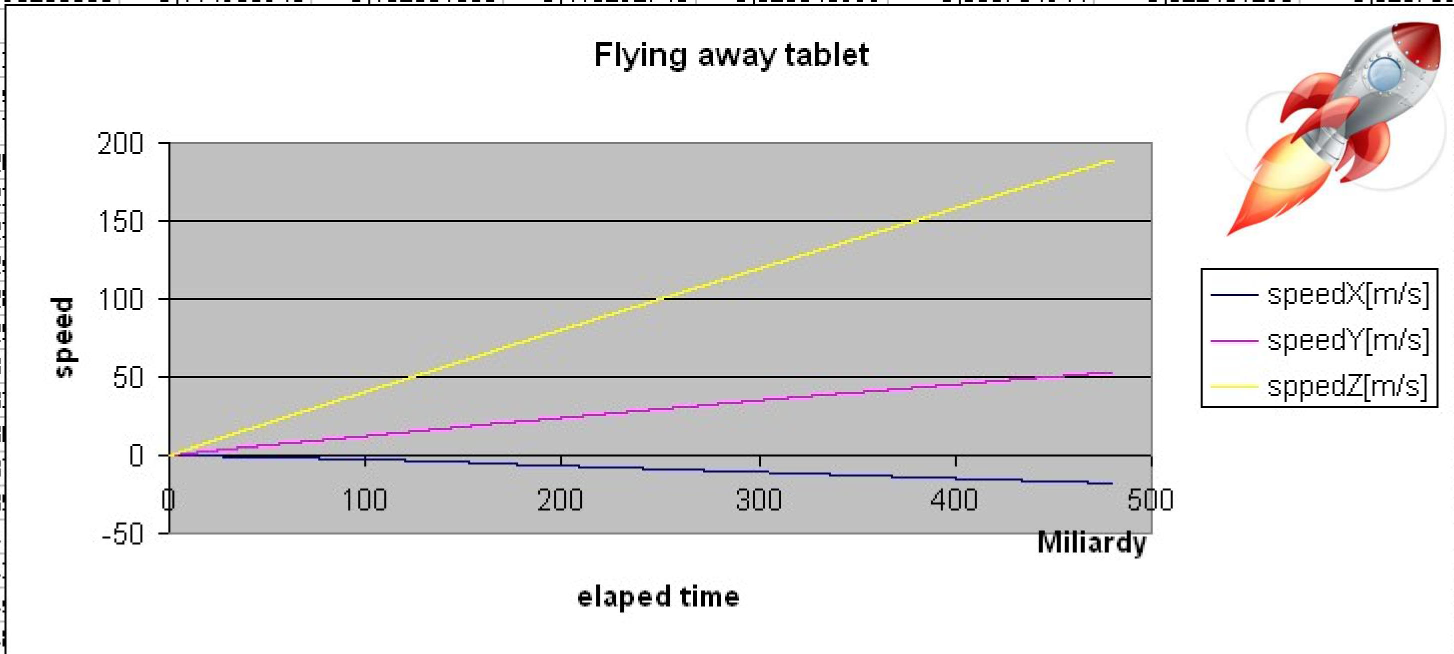

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
PrintStream ps;
protected void onResume() {
   ps = new PrintStream(new FileOutputStream(new File(
        Environment.getExternalStorageDirectory(), "accel.csv")));
   } catch (FileNotFoundException e) {

public void onSensorChanged(SensorEvent event) {
   if (ps != null)
      ps.print(
        (event.timestamp - startTime) +";"+ event.values[0] +";"+
        event.values[1] +";"+
        (event.values[2] - SensorManager.GRAVITY_EARTH) +"\n");

protected void onPause() {
   if (ps != null)  ps.flush();
```

Project:SensorAccel.zip

Excel Computing

accel_integral_s_grafom.xls

	A	B	C	D	E	F	G	H
1	elapsed time	x [m/s2]	y [m/s2]	z [m/s2]	delta time	speedX[m/s]	speedY [m/s]	speedZ[m/s]
2	0	-0,134121090	0,095800780	0,415292740	[s]	0,000000000	0,000000000	0,000000000
3	22489001	0,076640630	0,172441410	0,233271600	0,022489001	-0,000646339	0,003016249	0,007292782
4	38263000	0,095800780	0,411943350	0,300332070	0,015773999	0,000713706	0,007625292	0,011501314
5	58593000	0,210761730	0,268242180	0,166211130	0,020330000	=E5*((B5+B4)/2)+F4		0,016243726
6	78160001	-0,201181640	0,182021480	0,338652600	0,019567001	0,003923641	0,018944532	0,021183060
7	98206000	-0,114960940	0,162861330	0,415292740	0,020045999	0,000754944	0,022401293	0,028739854
8	1							811
9	1							214
10	1							916
11	1							847
12	2							790
13	2							743
14	2							833
15	2							255
16	2							845
17	2							743
18	3							662
19	3							367
20	3							583
21	3							768
22	3							902
23	4							031
24	4							103
25	4							273
26	4							483
27	499827000	0,028740235	0,114960940	0,386552800	0,017864000	-0,023344471	0,086640397	0,181182879



Project:**accel_raw.xls – surové dáta, accel_integral.xls – prvý integrál**

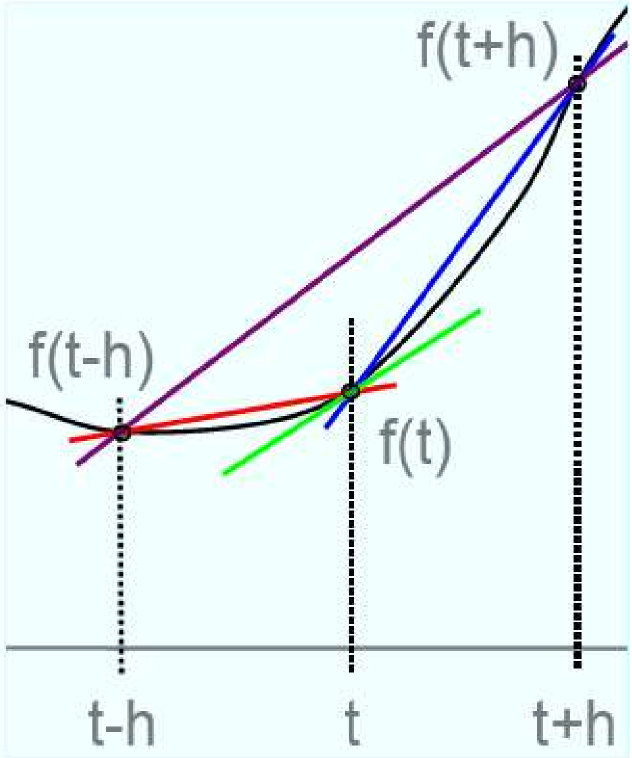
Je problém integrovanie

Definition: $$\frac{df(t)}{dt} = \lim_{dt\to 0}\left[\frac{f(t+dt)-f(t)}{dt}\right]$$

Forward estimate: $$\frac{df(t)}{dt} \approx \frac{f(t+h)-f(t)}{h}$$

Backward estimate: $$\frac{df(t)}{dt} \approx \frac{f(t)-f(t-h)}{h}$$

Central estimate: $$\frac{df(t)}{dt} \approx \frac{f(t+h)-f(t-h)}{2\cdot h}$$



Je problém Excel ?

```
speedX +=  (event.timestamp - lastTime)/1e9*event.values[0];
distX += (event.timestamp - lastTime)/1e9*speedX;
lastTime = event.timestamp;
```

Je problém frekvencia vzorkovania ?

Project:SensorAccelIntegral.zip

SensorActivity

x:-0.10538086
y:0.009580079
z:0.37697315
speedX:-4.159321536326763
speedY:-0.22854352503080377
speedZ:22.29868326828103
distanceX:-126.17289747592432
distanceY:0.8540353066052624
distanceZ:679.0257414396833
time:61.126801
MPL accel, maxRange:10240.0, resol.:1.0, minDelay:20000, rate[/ms]:0

SENSOR_DELAY_FASTEST

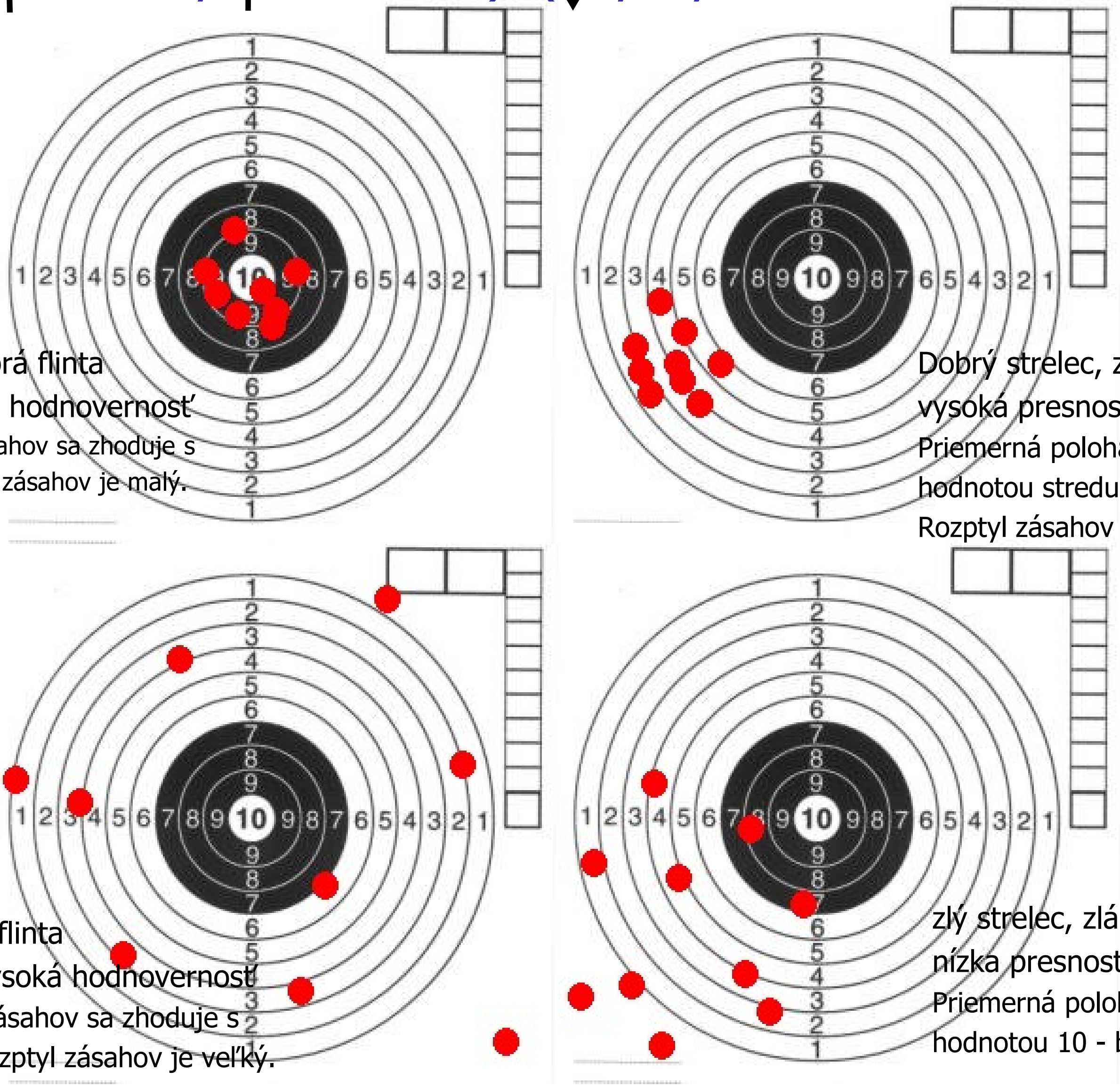
SensorActivity

x:-0.09580078
y:0.028740235
z:0.3865528
speedX:-3.766435234487395
speedY:1.596941474028592
speedZ:21.90616844880135
distanceX:-109.9939287890509
distanceY:56.90706030389773
distanceZ:660.9154906517665
time:60.260081
MPL accel, maxRange:10240.0, resol.:1.0, minDelay:20000, rate[/ms]:2

SENSOR_DELAY_UI

Hodnovernosť/Presnosť

(↑Accuracy/↑Precission) (↓vychýlenie= drift=bias /↓ disperzia/noise)



Dobrý strelec, dobrá flinta
vysoká presnosťaj hodnovernosť
Priemerná poloha zásahov sa zhoduje s hodnotou 10. Rozptyl zásahov je malý.

Dobrý strelec, zlá flinta
vysoká presnosť, nízka hodnovernosť
Priemerná poloha zásahov sa nezhoduje s hodnotou stredu terča. Odchýlka sa nazýva bias. Rozptyl zásahov je malý.

Zlý strelec, dobrá flinta
nízka presnosť, vysoká hodnovernosť
Priemerná poloha zásahov sa zhoduje s hodnotou 10, ale rozptyl zásahov je veľký.

zlý strelec, zlá flinta
nízka presnosť, nízka hodnovernosť
Priemerná poloha zásahov sa nezhoduje s hodnotou 10 - bias. Rozptyl zásahov je veľký.

Chyby senzora

- šum (noise) – low pass filter (ukážeme si...)
- vychýlená muška (drift) – adaptujeme hodnoty o nejaký offset, kalibrácia
- zero offset – sensor nedáva 0 ani v kľude

Riešenia:

- nastavením nuly – kalibračným tlačidlom
- filtrovaním – low pass a high pass filter, frame filter (tzv.plávajúci priemer)
- kombináciou hodnôt rôznych (súvisiacich) senzorov

  !!! odporúčam si pozrieť prednášku:

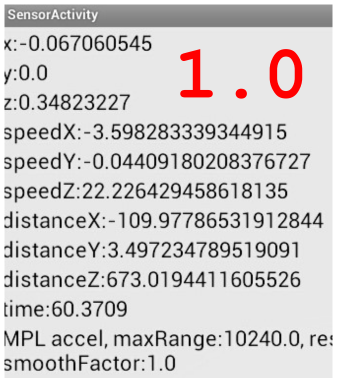
  Sensor Fusion on Android Devices, David Sachs, GoogleTechTalks, 2010

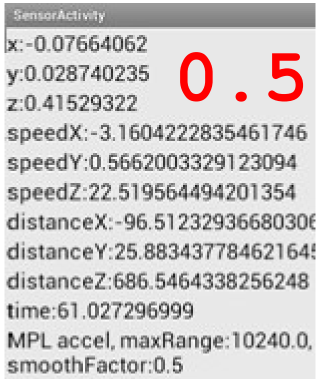
  http://www.youtube.com/watch?v=C7JQ7Rpwn2k

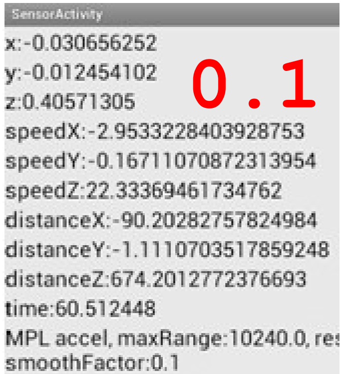
Low Pass Filter

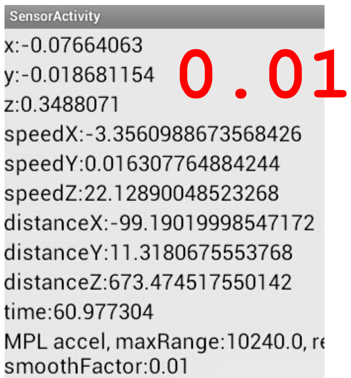
Low (frequency) pass – pomaly meniace dáta filtrom prejdú, náhle zmeny nie

```
// weighted smoothing váhované žehlienie
newValueX = weightedSmoothing(event.values[0], lastValueX);
lastValueX = event.values[0];
SpeedX += (event.timestamp - lastTime)/1e9* newValueX;

float smoothingFactor    = 1.0f;         // nič nežehlíme
                         = 0.5f;         // a.priemer posebeidúcich
                         = 0.1f;         // vysoké žehlenie
float weightedSmoothing(float newVal, float oldVal) {
  return  newVal*smoothingFactor + (1.0f-smoothingFactor)*oldVal;
```





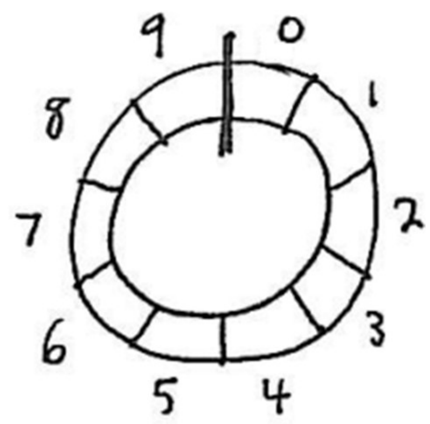




Project:SensorAccelLowPas.zip

Plavajúci priemer

```
public PlavajuciPriemer(int frameSize) {
   this.frameSize = frameSize;
   frame = new float[frameSize];
   count = indexLast = 0; average = 0;
}
public float add(float val) {
   if (count++ == 0) { // prázdny frame
        for (int i = 0; i < frameSize; ++i)
                frame[i] = val; // všetky val
        average = val;
   }
   float lastValue = frame[indexLast];   // von z frame
   average += (val - lastValue) / frameSize; // update average
   frame[indexLast] = val;                   // val do frame
   indexLast = (++indexLast) % frameSize;// kto ďalší
   return average;
}
```



Project:SensorAccelFrameAveragezip

Plavajúci priemer - výsledky

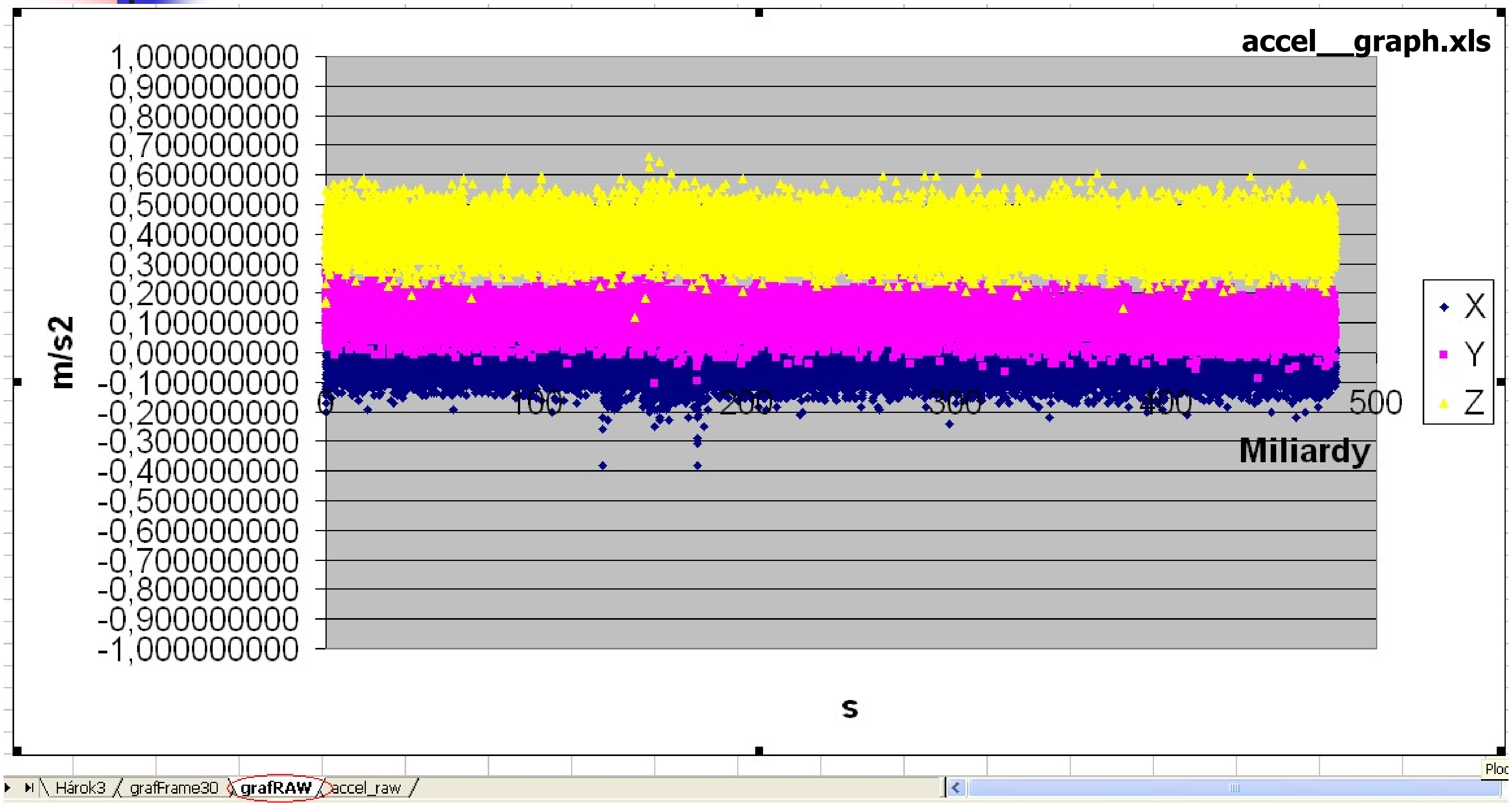
SensorActivity	SensorActivity	SensorActivity	SensorActivity
10	100	1000	1000
x:-0.05364839	x:-0.052307297	x:-0.054309484	x:-0.1084768
y:0.01820214	y:0.023183823	y:0.018623622	y:-0.029221334
z:0.35781258	z:0.3762068	z:0.37230647	z:0.35190618
speedX:-3.173261267247902	speedX:-3.196427873657978	speedX:-3.6067348168384923	speedX:-6.792806623407397
speedY:1.3546656983323895	speedY:1.871421845249238	speedY:1.447715156406039	speedY:-2.055527222109543
speedZ:22.593706507333316	speedZ:22.58608023113226	speedZ:22.721005969227377	speedZ:21.286434300840696
distanceX:-95.78163654227322	distanceX:-97.90495042045065	distanceX:-117.56637838957712	distanceX:-208.70946669447403
distanceY:47.81902766555779	distanceY:69.43149261678886	distanceY:50.03727618448285	distanceY:-65.33244522836334
distanceZ:685.9978794660202	distanceZ:691.0564856941342	distanceZ:698.701919773617	distanceZ:646.466551073196
time:60.628649001	time:60.992929	time:62.305015001	time:60.805509
MPL accel, maxRange:10240.0, re	MPL accel, maxRange:10240.0,	MPL accel, maxRange:10240.0, re	MPL accel, maxRange:10240.0, re
frameSize:10	frameSize:100	frameSize:1000	frameSize:10000

Kontrolná otázka: koľko buffrov potrebujeme ?

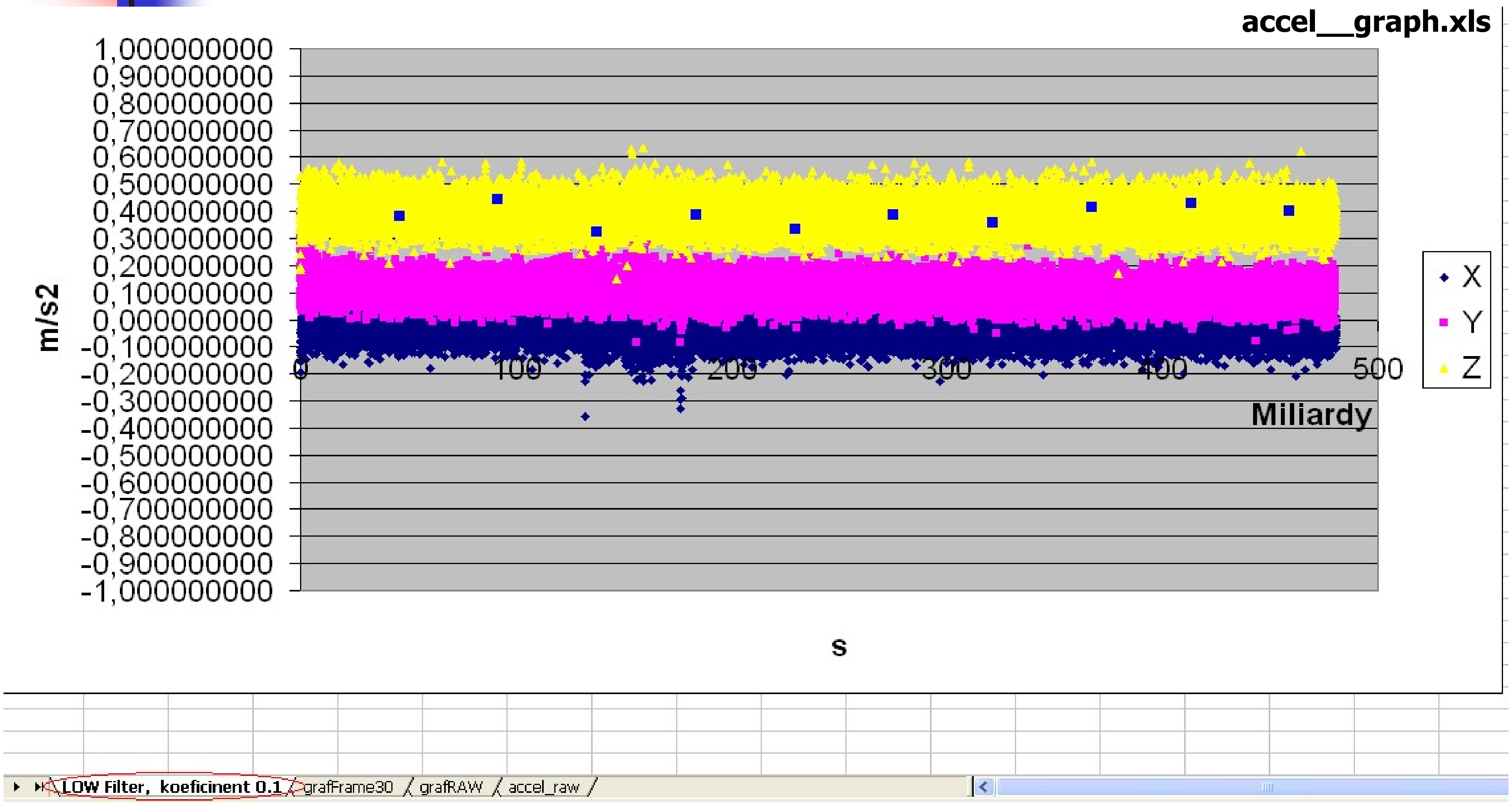
```
ppX = new PlavajuciPriemer(SIZE);
ppY = new PlavajuciPriemer(SIZE);
ppZ = new PlavajuciPriemer(SIZE);
```

Skúsime *TYPE_GRAVITY ...*

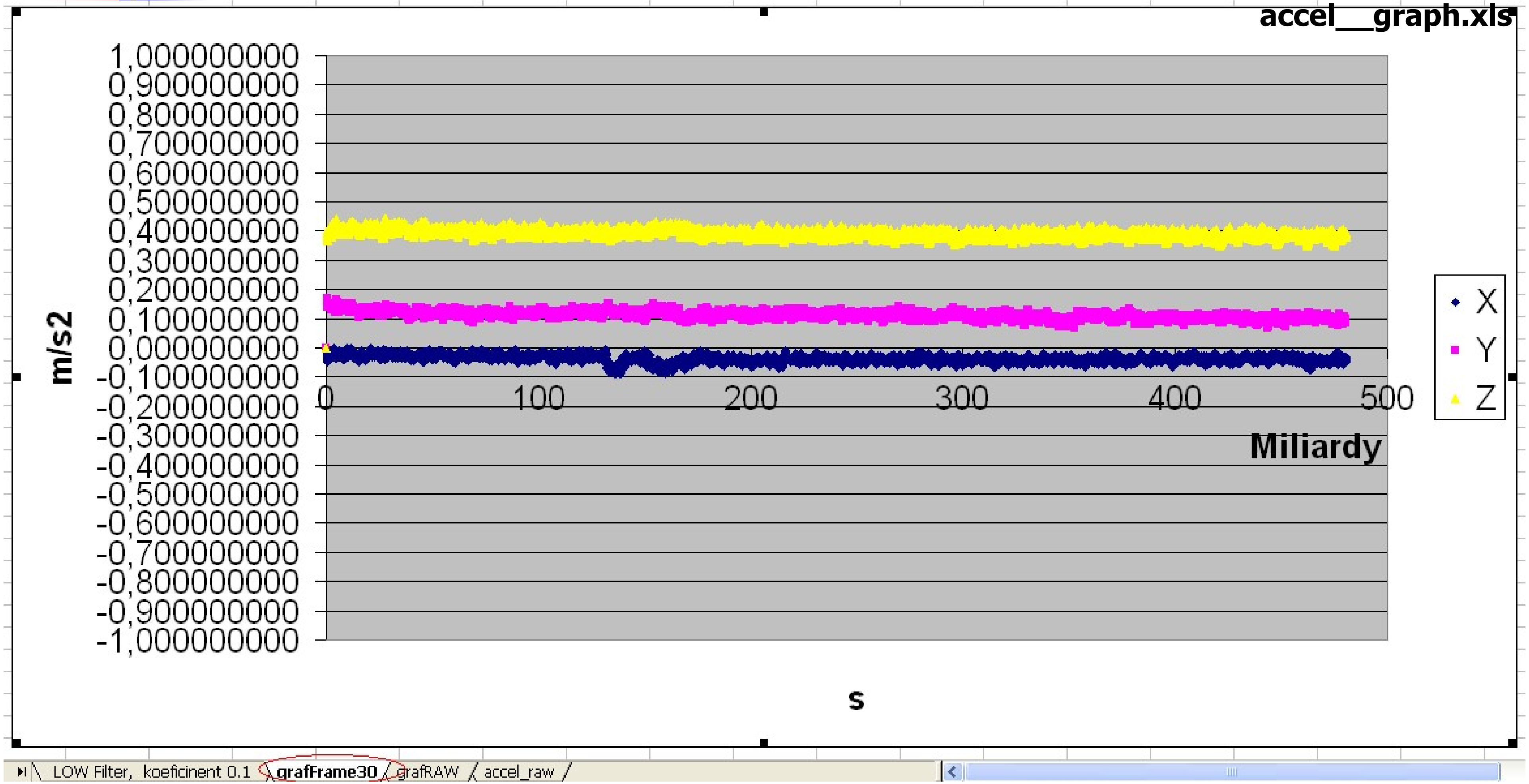
Excel graph – surové hodnoty



Žehlenie s koeficientom 0.1



Plávajúci priemer, frame = 30



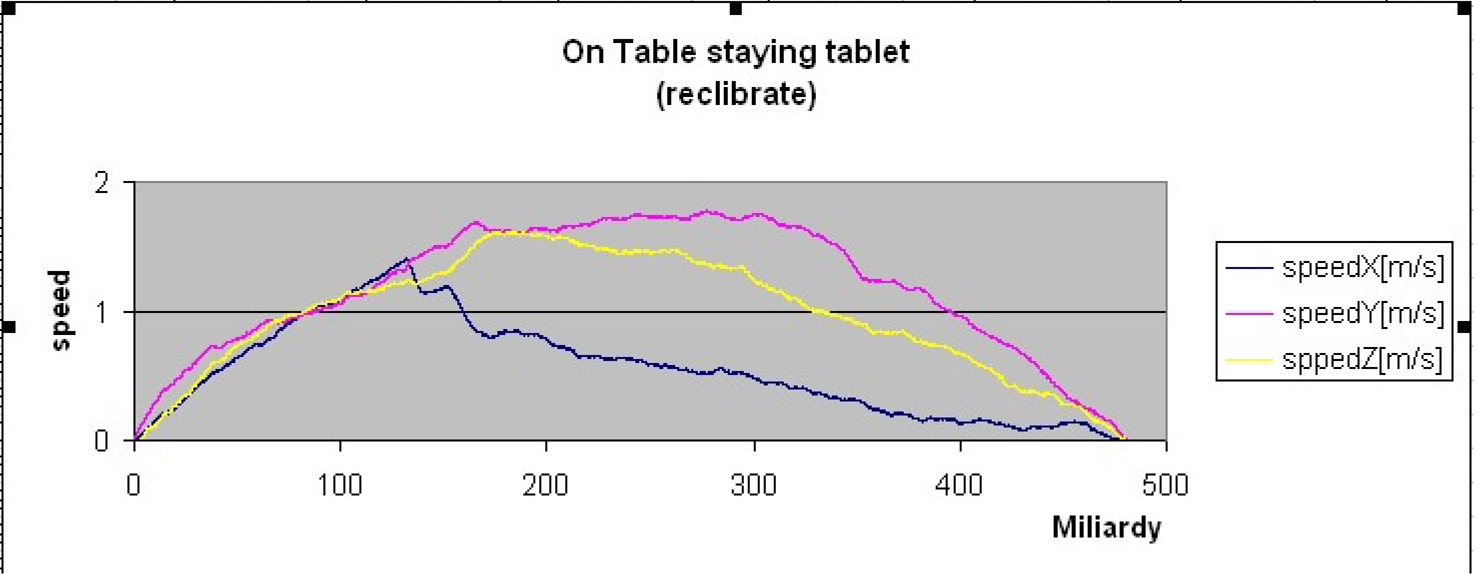
Kalibrujeme senzor

accel_recalibrate.xls

Ak pri meraní boli podmienky, aby výsledky boli (0,0,0), odčítame od hodnôt dlhodobé priemery.

23929	480 588 254 000	-0,047900390	0,067060545	0,367392540	0,021734000	-0,001873977	-0,001085352	0,011551637
23930	480 606 652 000	0,000000000	0,134121090	0,386552800	0,018398000	-0,001189746	-0,000653854	0,011420319
23931		**-0,037190513**	**0,110667539**	**0,393690470**	**480,606652000**			

elapsed time	x [m/s2]	y [m/s2]	z [m/s2]	delta time	speedX[m/s]	speedY [m/s]	speedZ[m
0	-0,134121090	0,095800780	0,415292740	[s]	0,000000000	0,000000000	0,000000
22 489 001	0,076640630	0,172441410	0,233271600	0,022489001	0,002559949	0,001389233	-0,003607
38 263 000	0,095800780	0,411943350	0,300332070	0,015773999	0,004657753	0,006141557	-0,005080
58 593 000	0,210761730	0,268242180	0,166211130	0,020330000	0,009698622	0,009345049	-0,009704
78 160 001	-0,201181640	0,182021480	0,338652600	0,019567001	0,006489808	0,010741232	-0,010781
98 206 000	-0,114960940	0,162861330	0,415292740	0,020045999	0,004930822	0,011787509	-0,010348



Problém vzdialenosti je vážny

Je možné navigovať polohu/zmenu polohy vo vnútorných priestoroch bez

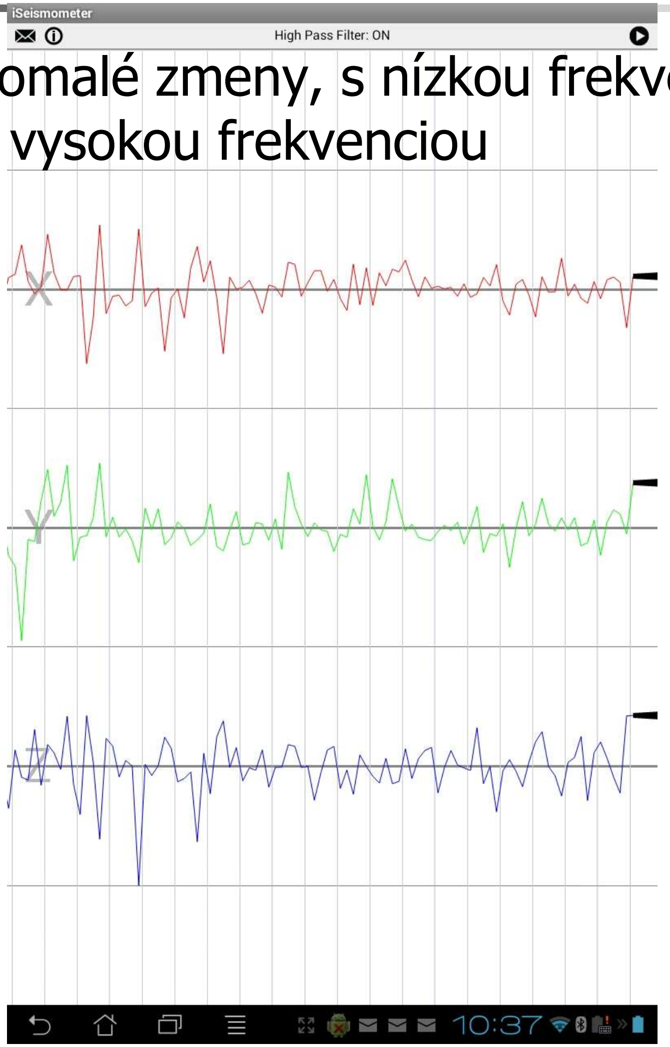
- GPS,
- WiFi, BT, RFID

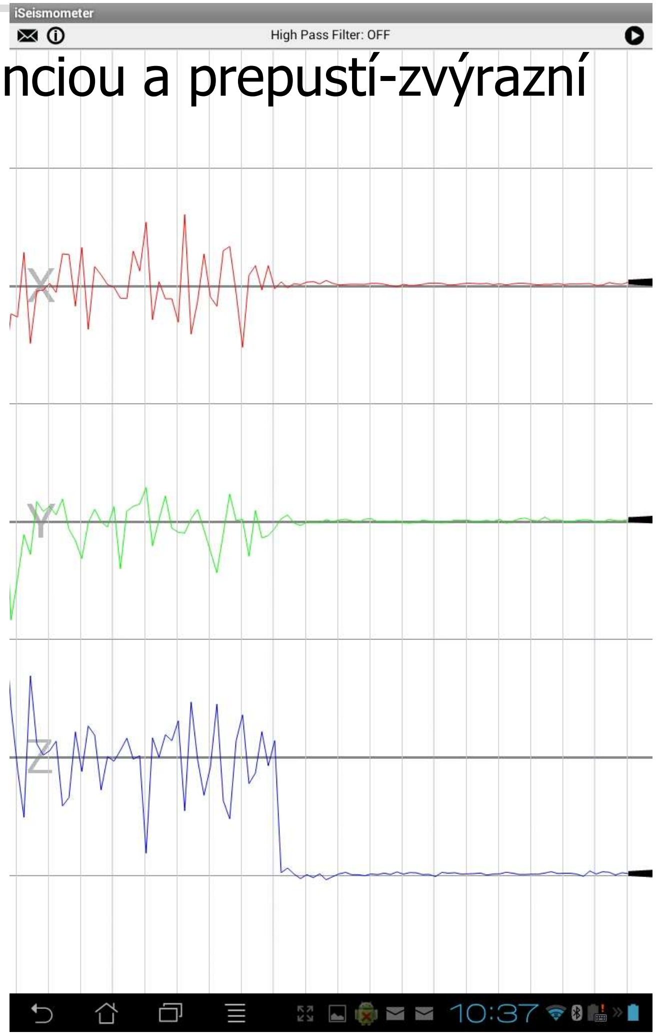
len pomocou akcelerometra ?

- Oliver J. Woodman, "An introduction to inertial navigation", PhD thesis, 2010
  http://www.cl.cam.ac.uk/research/dtg/www/people/ojw28
- **pekné, ilustratívne, programátorsky ladené, a pochopíte úlohu filtrov, frame (windowing) aj kalibráciu, odporúčam si pozrieť pred DÚ:**
  Implementing Positioning Algo using Accelerometer
  http://www.freescale.com/files/sensors/doc/app_note/AN3397.pdf
- Finding position via an accelerometer is all but impossible...
  http://forums.anandtech.com/showthread.php?t=2076227
- You get position by integrating the linear acceleration twice but **the error is horrible. It is useless in practice.**
  http://stackoverflow.com/questions/7829097/android-accelerometer-accuracy-inertial-navigation/7835988#7835988
- Robotics - Measuring Distance with an Accelerometer
  http://kristoph.minchau.com/Robotics/Resources/distance_with_accelerometer.html

Príklad na HighPass Filter

od filtruje pomalé zmeny, s nízkou frekvenciou a prepustí-zvýrazní rýchle, s vysokou frekvenciou





Signal processing

To, čo sa snažíme objaviť sa volá spracovanie signálu, v praxi existuje množstvo rôznych filtrov

http://www.falstad.com/dfilter/

http://www.falstad.com/mathphysics.html