HOW TO INITIALISE A SMS SERVO AT A SELECTED ZERO POSITION

Modified on: Mon, 12 Oct, 2015 at 3:51 PM

Decide a position from approximately which you will begin. Eg. a final 210degree position (or in can be something else) and an initial position (your 0 degree) position as they are in your structure and set-up. We will call these the "End Position". and "Initial Position". This position will be achieved exactly.

Use the SMS Commander via connectivity with **RS485** (http://01mechatronics.com/product/usb-rs-485)(see our download section @ http://goo.gl/LXdZAk)

- Power up your system in the End Position. DO NOT start the controllers.
- Execute a getAbsolutePosition for each controller. Note down the values on a piece of paper: END_POS_1
- Execute a resetIncrementalPosition for the controller.
- Take your setup by hand to the Initial Position and fix it there by some means.
- Execute a getPosition for each controller and also note them down. They are INIT_POS_1

These constants should be written in the code. We selected the INIT Position to be downwards and the END Position to be near 210degrees - counterclockwise. Our positive values on ticks movements give a counterclockwise direction. So these ticks were registered and reflected to the ino file void setup () section presented herewith:

C++

```
#include <ZerooneSupermodified.h>
#include <Wire.h>

long time;

ZerooneSupermodified motor(ZO_HW_WIRE);

#define END_POS_1 (32053)                       //the number of axis 1 from your notes
#define INIT_POS_1 (-19328)                     //the number from your notes

void setup() {
  // put your setup code here, to run once:

Serial.begin(57600);
  Serial.print("Started:");
  Serial.println();

int wakeUpPos1;
int zeroPos;

        motor.start(5);

        wakeUpPos1 = motor.getAbsolutePosition(5);

        //Move motors to Initial Position

        motor.profiledMoveToAbsolutePosition( 5, END_POS_1 - wakeUpPos1 + INIT_POS_1 );
        zeroPos=(END_POS_1 - wakeUpPos1 + INIT_POS_1);

        //after the above commands and provided that the actual end position is
        //within 1 turn of the noted END POSITION the setup must have assumed its
        //initial position.

}
void loop() {

//Your amazing SMS V3.0 code here

}
```