

VEX ROBOT ARM - REUSABLE CALIBRATION CODE

(port or name) needs: ^{sensors} lower limit switch port #, upper limit switch port, motor port, encoder port

Sensors & Motors: Lower limit switch, upper limit switch, motor port, encoder

Lower limit's degrees from origin, upper limit switch's degrees from origin

Struct ~~Motor~~ ^{MOTOR} SINGLE ARM CALIBRATION

~~set~~ everything from above, &

float - encoder degrees per step, float lower ^{encoder} offset.

boolean - has been calibrated,

function pseudocode: (struct SINGLE ARM MOTOR CALIBRATION ~~convert~~ convert)

calculate degrees ^{lower} move arm downward until limit switch.

traveled.

set lower offset = equal to current encoder reading.

(origin lower)

move arm upward to upper limit switch.

(origin upper) -

calculate # of encoder steps per degree of actual movement.

(origin - lower)

set has been calibrated to true.

end function.

this will work for everything but the lower -

Arm Motor functions

Arm Motor structure

motor port

motor encoder

steps per degree - set by calibration function

motor max angle

motor min angle

motor current angle (if not a function)

calibration substructure

motor advised max speed

calibrate(~~is~~)

get current angle();

calibrate (struct ~~Arm~~ Calibration);

float get current angle (struct Arm);

returns ^{relative} angle of arm as float.

goToAngle (struct Arm, speed (relative));

