/***

Module

ServoControl.c

Description

This module acts as the low level interface to initialize and control servos.

***/

Static module-level variables:

static uint16_t ServoSlidePotPosition;

static uint16_t ServoRotPotPosition;

static uint16_t ServoBeamPosition;

static uint16_t ServoTOTPosition;

static uint16_t ServoTOTBlockerPosition;

InitServos
Takes nothing; returns nothing

Initialize all needed PWM lines

Set the period for the PWM lines

End InitServos

SetServo
Takes the desired servo number and desired position in ticks for PWM width; returns nothing

If servo number corresponds to SERVO_SLIDE_POT

If DesiredPosition is within specified servo PWM width limits

Set PWM line to desired pulsewidth

Update ServoSlidePotPosition with DesiredPosition

Initialize no-user-interaction 30 second timer

If servo number corresponds to SERVO_ROT_POT

If DesiredPosition is within specified servo PWM width limits

Set PWM line to desired pulsewidth

Update ServoRotPotPosition with DesiredPosition

Initialize no-user-interaction 30 second timer

If servo number corresponds to SERVO_TOT

If DesiredPosition is within specified servo PWM width limits

Set PWM line to desired pulsewidth

Update ServoSlidePotPosition with DesiredPosition

If servo number corresponds to SERVO_BEAM

If DesiredPosition is below specified lower servo PWM width limit

Set PWM line to desired pulsewidth

Post CCWLimitReached event to ServoBeam.c

Else if DesiredPosition is above specified upper servo PWM width limit

Set PWM line to desired pulsewidth

Post CWLimitReached event to ServoBeam.c

Else

Set PWM line to desired pulsewidth

Update ServoBeamPosition with DesiredPosition

Initialize no-user-interaction 30 second timer

If servo number corresponds to SERVO_TOT_BLOCKER

Set PWM line to desired pulsewidth

Update ServoTOTBlockerPosition with DesiredPosition

End SetServo

GetServoPosition

Takes the desired servo number; returns integer value of PWM width in ticks for desired servo

If servo number corresponds to SERVO_SLIDE_POT

Return ServoSlidePotPosition

If servo number corresponds to SERVO_ROT_POT

Return ServoRotPotPosition

If servo number corresponds to SERVO_BEAM

Return ServoBeamPosition

If servo number corresponds to SERVO_TOT

Return ServoTOTPosition

If servo number corresponds to SERVO_TOT_BLOCKER

Return ServoTOTBlockerPosition

End GetServoPosition

AD2ServoTicks

Takes the desired servo number and an A/D value between 0 and 4095; returns integer of PWM width ticks

Use the upper/lower pulsewidth limit values for the specified servo

Convert 0-4095 value to corresponding value between lower/upper pulsewidth limits, in ticks

Calculate the center of the upper/lower pulsewidth limits for the specified servo

To flip direction of servo control:

If tick value is below center value

Return difference between center value and tick value, plus center value

Else if tick value is above center value

Return difference between center value and tick value, minus center value

Else

Return center value

End AD2ServoTicks