Crane control

The crane control problem for the pendulum-cart setup also refers to a SIMO plant description. The goal is to travel with the crane from one point to another keeping the pendulum swinging as little as possible (Figure 27).

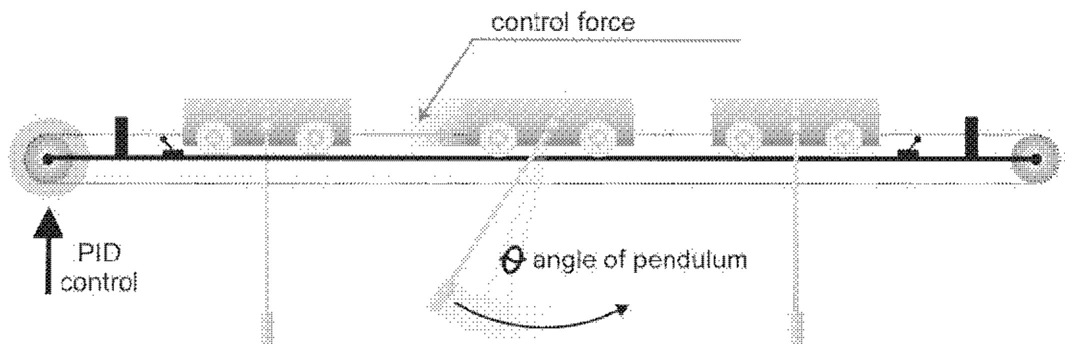

Figure 27 Crane control

The problem can be solved with many control engineering algorithms. Again the simplest method is presented here.

As with the inverted pendulum stabilisation, PD and LQ controllers can be used as they will actually have the same structure. They both use weighted errors of four state variables of the pendulum: x (cart position), $\dot{x}$ (cart velocity), θ (pendulum angle) and $\dot{\theta}$ (pendulum angular velocity). For both of the controllers, the linear model of the pendulum is used which has been obtained either from linearisation of the nonlinear model for $\theta = \pi$ radians or from identification of the crane model.

To simplify the design process the following exercises with Simulink demonstrations have been prepared, however more advanced users should be able to design the crane control system based on the information already given.

Exercise 13 – Crane control

Introduction

For this exercise you can use the *CraneModelStab.mdl* for model simulation and *CraneStab.mdl* for real time crane control. Make sure the control starts with the pendulum hanging freely and not moving.

Task

First run the offline simulation *CraneModelStab.mdl.* Start the simulation and observe the crane following the desired trajectory. Change the desired trajectory to see how the pendulum reacts.

Make sure the trajectory value is inside the range of cart movement.
Change the controller *P*, *D* and *I* gains to see how they influence the control action.

Now run the real-time model *CraneStab.mdl* and experiment with *P*, *D* and *I* gains to see how they influence the real system.

Example results and comments

Depending on the *P*, *D* and *I* gains you should observe the same reaction as described in the '*PID controller*' section.

Combined control techniques

This section combines the control techniques into a complete pendulum control application. The tasks are divided into exercises in order of increasing difficulty.

Swing up and hold

This application combines the swing up control algorithm with the pendulum LQ or PD control in the upright position. You can try to combine two of these control actions. Remember that excessive pendulum velocity when entering the stabilisation zone might cause the pendulum to over swing.

Exercise 14 – Swing up and hold control

Introduction

For this exercise you can use the *SwingHoldModel.mdl* for model simulation and *SwingHoldPendulum.mdl* for real time pendulum control. Make sure the control starts with the pendulum hanging freely and not moving.

Task

Run the offline model *SwingHoldModel.mdl.* Observe the swing-up controller bring the pendulum close to the stabilisation zone where the stabilisation controller takes over.

Change the controller *P*, *D* and *I* gains to see how they influence the stabilisation action.

Change the swing up parameters to see how that influences the whole control task.

Run the real-time model *SwingHoldPendulum.mdl* and observe the above effects on the real system.

Example results and comments

You may have noticed that the swing-up control sometimes lets the cart move out of bounds and it trips the limit switches. Exercise 15 will show how this can be improved. You can also use the improved real-time model *SwingHoldPendulumExtra.mdl* which has additional cart position control and observe how this improves the swing up. The extra position control block is described in Figure 42. Further information on how the additional position control is designed is given in the last section of the manual where the most advanced application is described *UpDownPendulum.mdl.*

Depending on the *P*, *D*, and *I* gains you should observe the same reaction as described in the '*PID controller*' section.

Changing the swing-up parameter may either cause the pendulum never to reach the upper zone or cause the pendulum to have excessive velocity when entering the stabilisation zone and thus cause problems for the stabilizing controller.

Up and down

This application adds fall and crane stabilisation of the pendulum to the previous swing up and hold controller. The pendulum will be swung up and stabilised in the inverted position for some time and then released to be stabilised and returned to the initial point where crane control is applied.

Exercise 15 – Up and down

Introduction

For this exercise you can use the *UpDownPendulum.mdl* for real time inverted pendulum control combined with crane control. Make sure the control starts with the pendulum hanging freely and not moving.

Task

Start the simulation and observe the swing-up controller bring the pendulum close to the stabilisation zone where the stabilisation controller takes over. After a period of time the stabilizing controller will be turned off and the pendulum will fall to be stabilised in the $x = 0$ and $\theta = \pi$ position. The cart is then made to follow a sinusoidal path (crane control). After a further period of time the control scheme is repeated until the time runs out.

You can experiment with the parameters of the control actions used within the model. Refer to the previous exercises for their explanation.

Example results and comments

Figure 28 shows the angle and the control signal of the pendulum during up and down task. Figure 29 shows the cart position.

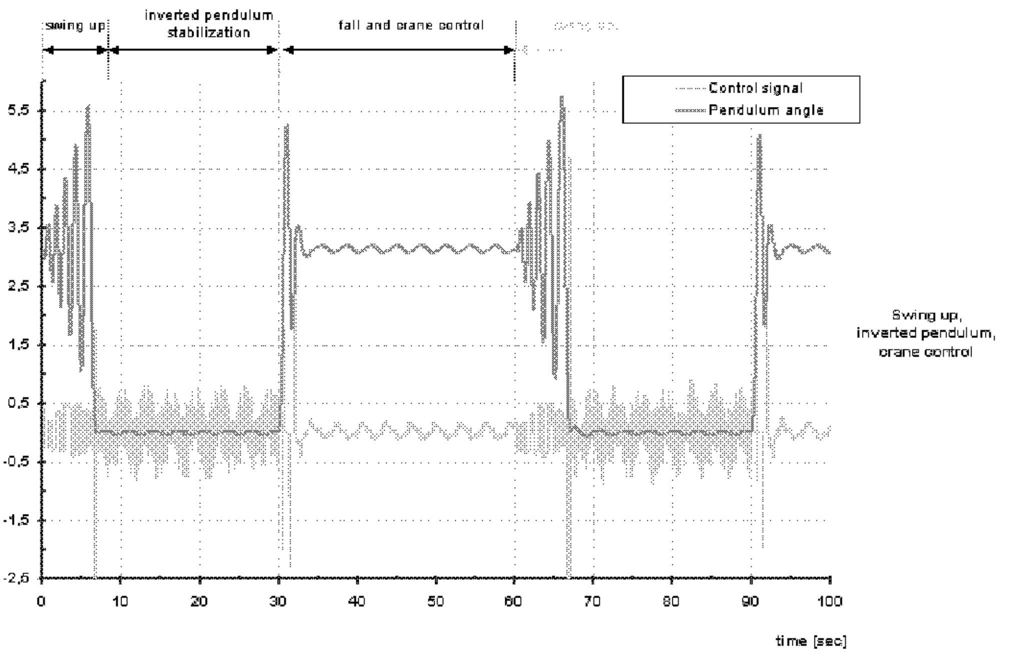

Figure 28 Control signal and pendulum angle

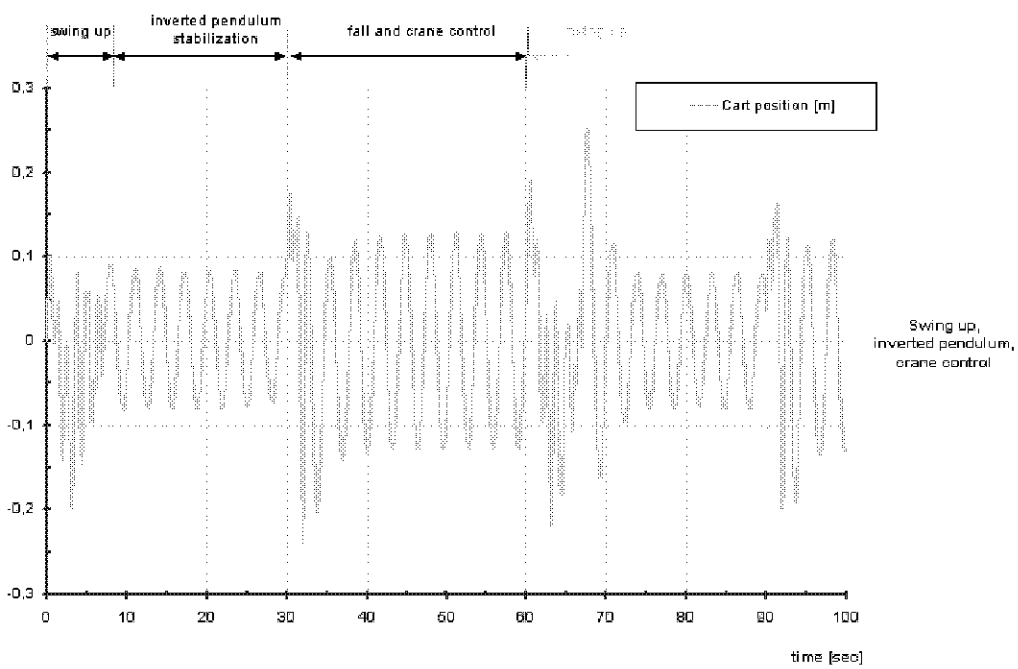

Figure 29 Cart position

The Up and Down demo includes some necessary control algorithm improvements resulting in robustness of the application. The improvements and the way each of the control block works is explained in the following subsection.

Detailed description of the Up/Down model

In case you have not designed all of the applications step by step on your own you might find this section particularly interesting. Here the blocks created in 'Up and Down' demonstration are described. Detailed information on each signal is given.

- Two most important blocks that have already been described in the "*Installation & Commissioning*" manual are the '*Feedback encoder for Pendulum*' block and the '*Feedback DAC for Pendulum*' block (Figure 30).

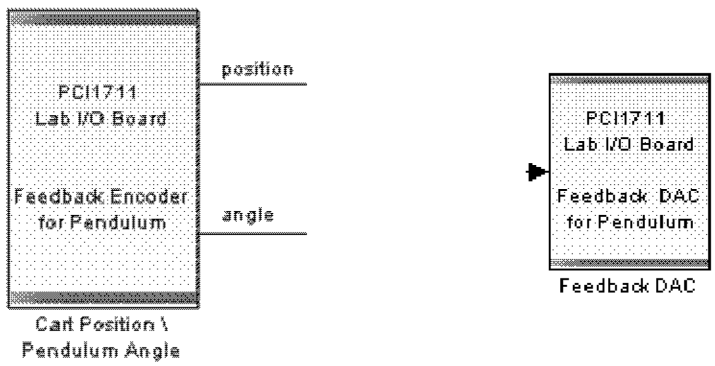

Figure 30 External equipment blocks

The outputs of the encoder block are the cart relative position in metres with respect to the initial starting point. The second is the pendulum relative angle in radians with respect to the initial starting point. The input of the '*Feedback DAC for Pendulum*' block is the control voltage for the pendulum motor. Inside the block, static friction compensation is added together with a safety block restricting the amplitude of the input control voltage to [-2.5 V .. +2.5 V]. The block converts the ±2.5V input signal to 0 to +5V when writing to the DAC as the PCI1711 I/O card has a unipolar (0V→+5V) DAC output.

- Another important block is the block presented in Figure 31.

Figure 31 Cart movement safety block

You can look inside this block and view the simple movement bounding function.

- The '*Task switching*' block (Figure 32) is used for switching the control actions. A pulse generator is used to switch between the two tasks: the swing up and stabilisation control, and the fall and crane control mode.

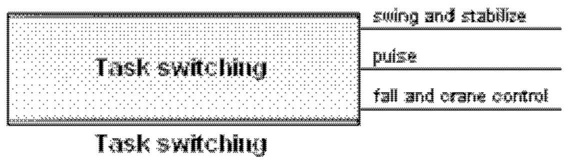

Figure 32 Task switching block

Block outputs: '*swing and stabilize*' when high, enables the swing and stabilize task, '*Pulse*' output for switching control signals, '*fall and crane control*' signal when high, enables the fall and crane control task.

- The '*Swing up and stabilisation control*' block (Figure 33) is responsible for swing up and inverted pendulum stabilisation. It is enabled by the 'Task switching' block.

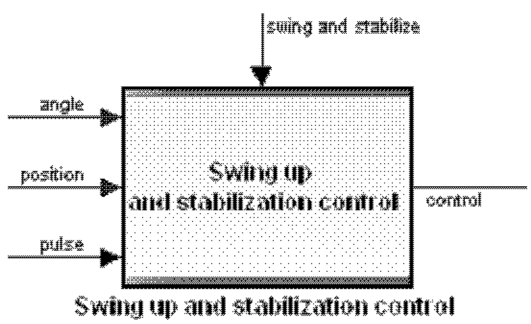

Figure 33 Swing up and stabilisation control block

Block inputs: *angle* of the pendulum, *position* of the cart, *pulse* (a delayed enabling signal for swing up initiation).
Block output: control signal.

Inside the '*Swing up and stabilisation control*' block two main control actions are designed, which are the swing up control (Figure 34) and the inverted pendulum stabilisation (Figure 35).

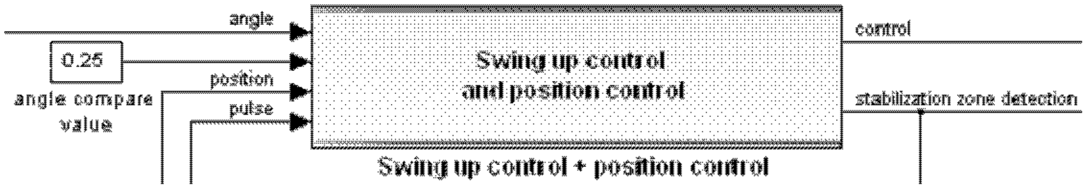

Figure 34 Swing up control

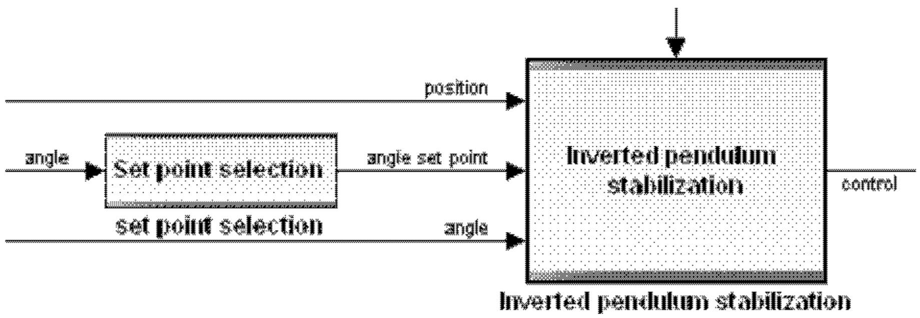

Figure 35 Inverted pendulum control

- '*Swing up control*' block (Figure 34).

  Block inputs: *angle* of the pendulum, *angle compare value* (explained below), *cart position, pulse (a* delayed enabling signal for swing up initiation).
  Block outputs: *control (*control voltage), *stabilisation zone detection* – goes high when the pendulum enters the stabilisation zone.

  Inside this block several control actions have been programmed.

  - The system presented in Figure 36 is responsible for the proper control signal application depending on the values of its inputs. It utilizes the swing up scheme described in the '*Swing up control*' section. The block is enabled by the *pulse* signal.

    Block inputs: *angular velocity* of the pendulum, *control voltage* – value of applied control voltage, *upper / lower zone detection* – high value signals upper half zone.
    Block outputs: *control* – control voltage.

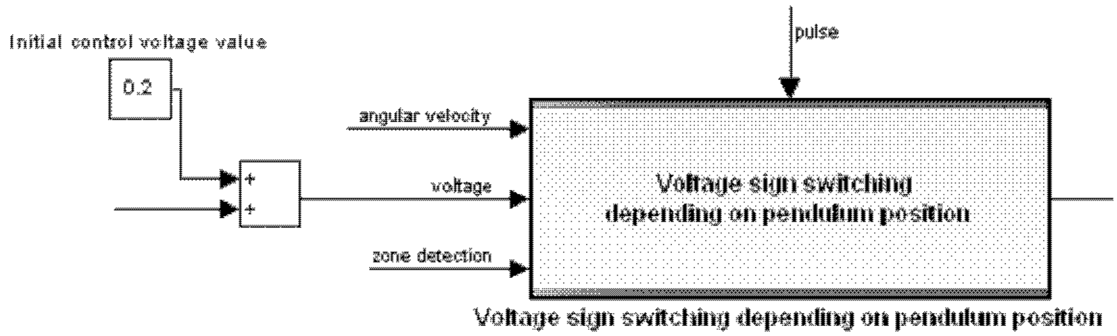

Figure 36 Voltage sign switching block for swing up

- Figure 37 presents the angular velocity calculation blocks. It consists of a derivative and low-pass filter block.

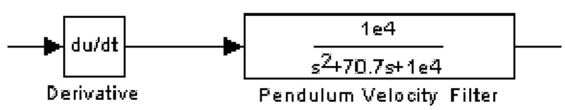

Figure 37 Angular velocity calculation

- The block presented in Figure 38 gives a high output signal when the pendulum enters the upper half zone.

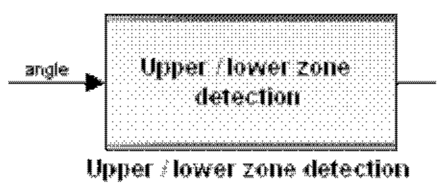

Figure 38 Upper zone detection

- If the pendulum angle is within the *angle compare value* of the vertical position, the block presented in Figure 39 gives a high output signal. This means the pendulum has entered the stabilisation zone.

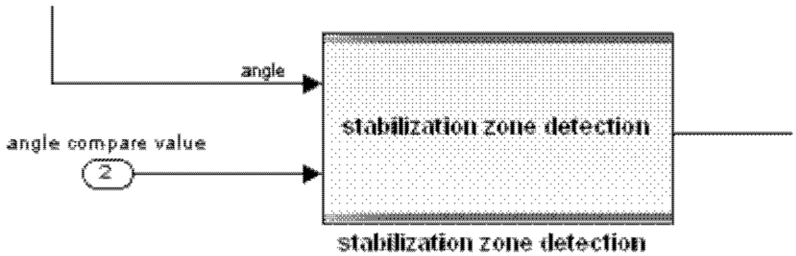

Figure 39 Stabilisation zone detection system

- First movement of the cart for the purpose of swing up is initiated by the '*First move*' block presented in Figure 40.

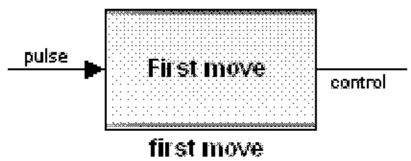

Figure 40 First move system

The next group of control subsystems has been created in the '*Swing up control*' block to improve the control performance and robustness.

- The subsystem presented in Figure 41 is responsible for control value calculation in the swing up control.

  Block inputs: *angle* – depending on its value the extra control signal value is generated. Its value is calculated by a sin(max|*angle*-π|) function. This assures a soft start and slow entry into the stabilisation zone. You may see the block functioning by observing its signals with Simulink scopes.

  Block outputs: *extra control* – extra control voltage.

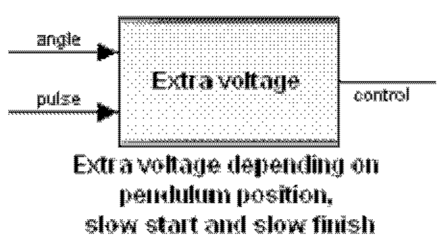

Figure 41 Extra voltage block

- To ensure that the cart will be kept within the admissible range, a PI controller is added to the swing up controller. The block responsible for the position control action is presented in Figure 42.

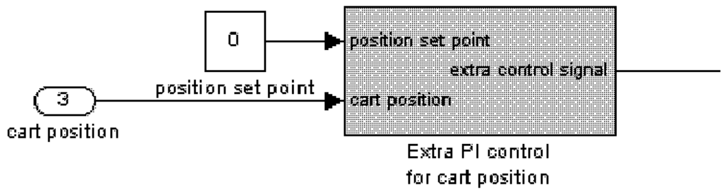

Figure 42 Position PI control block

- Inside the '*Inverted pendulum stabilisation*' block (Figure 35) two PID control algorithms are designed: one for the cart position control and the other for pendulum angle control. The desired value for the cart position controller is set to *0.* The set value for the angle is calculated by the '*set point selection*' block presented in Figure 43. The output of this block is either set to $2\cdot\pi$ or 0 depending on the side from which the pendulum has entered the stabilisation zone.

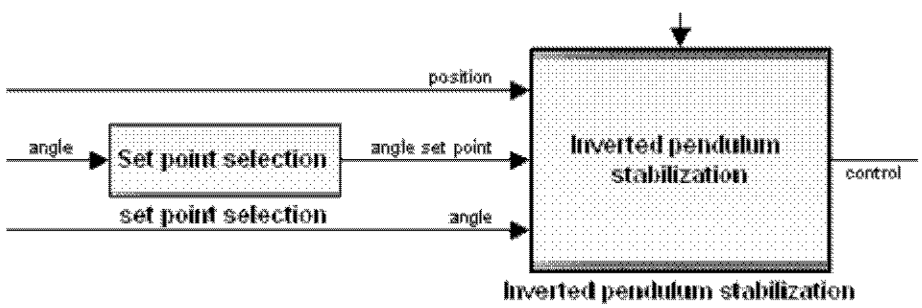

Figure 43 Vertical position knockout block

- In the main simulation window a '*knock from upright position*' block is placed (Figure 44).

- The block is enabled with the '*fall and crane control*' signal. Its inputs are the pendulum '*angle*' and the '*theta set point*'. The set point gives the value around which the inverted pendulum has been stabilised: either $2\cdot\pi$ or 0. The output is the control signal which is actually just a pulse of 0.5 V. Inside this block, the subsystem presented in Figure 44 gives a control voltage pulse dropping the pendulum in the direction which will ensure that the angle will again be $\theta=\pi$ in the lower vertical position. The falling is then initiated.

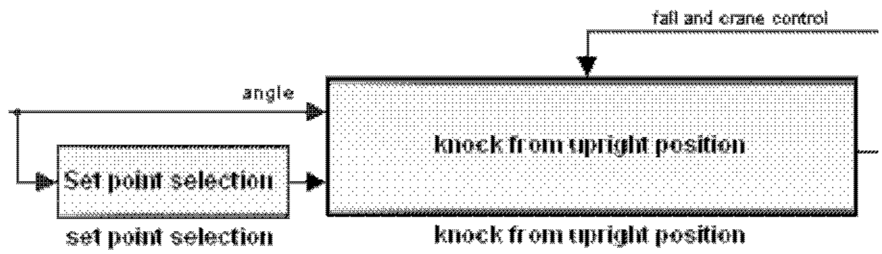

Figure 44 Pendulum fall initiation

- The '*Lower zone stabilisation and crane control*' block presented in Figure 45 provides pendulum fall control and crane stabilisation.

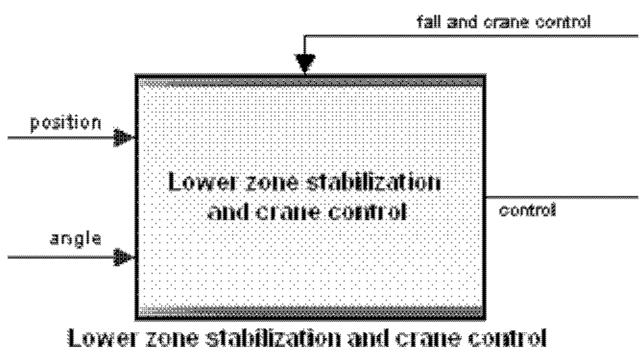

Figure 45 Lower zone and crane control block

Inside the block, two subsystems are designed: '*stabilisation zone detection*' (Figure 46) and '*crane control*' block (Figure 47).

- The '*stabilisation zone detection*' system gives a high output when the angle is within the '*angle compare value*' from the $\theta=\pi$ position.

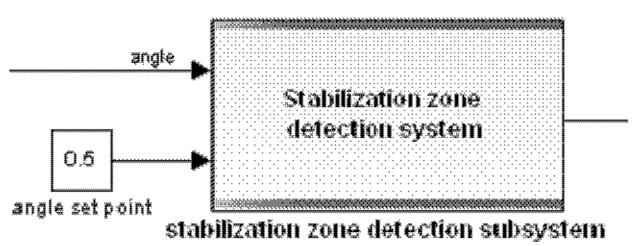

Figure 46 Stabilisation zone detection

- The '*crane control*' block utilizes the crane control algorithm. Two PID controllers are designed: one for pendulum angle and the other for cart position control.

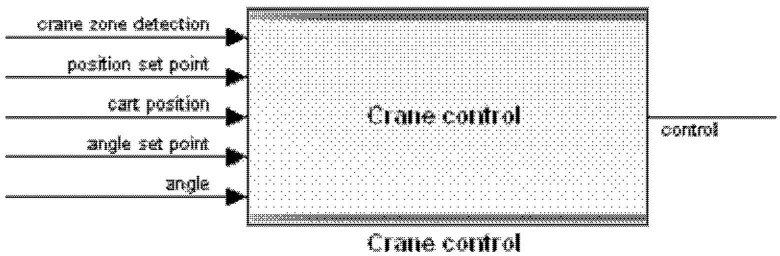

Figure 47 Crane control subsystem