- **Preparation**: See "What should I do in Week 7?" on SUNLearn.
- **Instructions**: Document all calculations, designs, and results properly. You do not have to hand in a practical report.
- **Note**: Study the Matlab/Simulink documentation by clicking on the *Help* menu if you do not know how to use a certain function/block.

- **Voorbereiding**: Sien "What should I do in Week 7?" op SUNLearn.
- **Instruksies**: Dokumenteer alle berekeninge, ontwerpe en resultate behoorlik. Jy hoef nie 'n praktiese verslag in te lewer nie.
- **Neem kennis**: Bestudeer die Matlab/Simulink-dokumentasie deur na die *Help*-kieslys te gaan indien jy nie weet hoe om 'n sekere funksie/blok te gebruik nie.

Background / *Agtergrond*

In Control Systems 314, you have learnt how to design a continuous classical controller to control a continuous plant. However, almost all controllers are implemented in digital processors, since it is much easier to implement and change complex controllers in a digital processor than in an analogue circuit, many sensors provide measurements as digital values, and many actuators accept digital commands. In this practical, we will investigate the effects of using a discrete controller on the performance of a control system, and compare it with the performance of a continuous controller.

For today's practical, we will use a simplified model of the electric robot vehicle of the extra example of Week 6. The input $u(t)$ of this plant is the voltage applied to the electric motor, and the output $y(t)$ is the angle of the back wheel. The (continuous) transfer function for this system is given by:

$$\frac{Y(s)}{U(s)} = G(s) = \frac{10}{s(s+2)}. \tag{1}$$

We will use a simple proportional controller to control this plant: we will first use a continuous-time controller, $D(s) = K$, and then use a discrete-time controller, $D(z) = K$.

In Beheerstelsels 314 het jy geleer hoe om 'n kontinue klassieke beheerder te ontwerp om 'n kontinue aanleg te beheer. Amper alle beheerders word egter geïmplementeer in digitale verwerkers, aangesien dit baie makliker is om komplekse beheerders in 'n digitale verwerker te implementeer en te verander as in 'n analoog-stroombaan, baie sensors gee metings uit as digitale waardes, en baie aktueerders neem digitale bevele. In hierdie prakties gaan ons ondersoek wat die effek op die gedrag van 'n beheerstelsel is wanneer ons 'n diskrete beheerder gebruik, en dit vergelyk met dié van 'n kontinue beheerder.

Vir vandag se prakties gaan ons 'n vereenvoudigde model van die elektriese robotvoertuig van die ekstra voorbeeld van Week 6 gebruik. Die intree $u(t)$ van hierdie aanleg is die spanning wat op die elektriese motor aangewend word, en die uittree $y(t)$ is die hoek van agterwiel. Die (kontinue) oordragsfunksie vir hierdie stelsel word gegee deur:

Ons gaan 'n eenvoudige proporsionele beheerder gebruik om die aanleg te beheer: ons gaan eers 'n kontinue-tyd beheerder, $D(s) = K$, gebruik, en dan 'n diskrete-tyd beheerder, $D(z) = K$, gebruik.

Assignment / *Voorskrif*

1. Draw the (continuous) root locus of the plant and proportional controller. Will the closed-loop system ever be unstable? Determine the value of K such that the closed-loop step response will have an overshoot of 20%. Determine the 2% settling time for the closed-loop system with the chosen value of K.

   Teken die (kontinue) wortellokus van die aanleg en proporsionele beheerder. Sal die geslotelusstelsel ooit onstabiel raak? Bepaal die waarde van K sodat die geslotelus-trapweergawe 'n oorskiet van 20% sal hê. Bepaal die 2%-wegsterftyd vir die geslotelus-stelsel met die gekose waarde van K.

2. Build the continuous control system in Simulink (green block in Figure 1). Verify that the closed-loop system satisfies the specifications. If the gain K changes, will the closed-loop system ever become unstable? Explain your answer by referring to the root locus. Verify your answer by adjusting the gain K in the Simulink model and observing the closed-loop response. Reset the value of K before the next question.

   Bou die kontinue beheerstelsel in Simulink (groen blok in Figuur 1). Bevestig dat die geslotelus-stelsel aan die spesifikasies voldoen. Indien die aanwins K verander, sal die geslotelus-stelsel ooit onstabiel raak? Verduidelik jou antwoord deur na die wortellokus te verwys. Bevestig you antwoord deur die aanwins K in die Simulink-model te verstel en die gedrag te beskou. Herstel die waarde van K voor die volgende vraag.

3. To implement a discrete controller in Simulink, we have to be able to convert analogue signals to discrete signals, and discrete signals to analogue signals[1]. We implement both conversions using the *Zero-Order Hold* Block (in the *Discrete* Folder). Now build the control system with the discrete controller in Simulink (blue block in Figure 1). Use the same controller gain K as for the continuous controller. Set the sample period of the *Zero-Order Hold* Blocks to 0.1 s [2]. Note that all signals in the red block of Figure 1 are discrete.

   Om 'n diskrete beheerder in Simulink to implementeer, moet ons analoog-seine kan omskakel na diskrete seine, en diskrete seine kan omskakel na analoog-seine[3]. Ons implementeer beide omskakelings deur die *Zero-Order Hold*-blok (in die *Discrete*-kieslys). Bou nou die beheerstelsel met die diskrete beheerder in Simulink (blou blok in Figuur 1). Gebruik dieselfde beheerder-aanwins K as vir die kontinue beheerder. Stel die monster-periode van die *Zero-Order Hold*-blokke na 0.1 s [4]. Let op dat alle seine in die rooi blok van Figuur 1 diskreet is.

[1] In Simulink, all signals are continuous (i.e. defined for all time t) – discrete signals are technically only defined at the sampling instants $t = kT$. To simulate a discrete signal in Simulink, the value of the signal is held constant over the sampling period, giving it a "stair"-like appearance.

[2] To make things easier, define a variable in Matlab (e.g. `T = 0.1`) and write the variable name (e.g. `T`) in the *Sample time* field, instead of the value (e.g. `0.1`). To change the sample time, all you have to do is to change the variable value in Matlab (e.g. `T = 0.2`).

[3] Alle seine in Simulink is kontinu (d.w.s. gedefinieer vir alle tyd t) – diskrete seine is tegnies slegs gedefinieer by die monster-oomblikke $t = kT$. Om 'n diskrete sein in Simulink te simuleer, word die waarde van die sein konstant gehou oor die monster-periode, wat 'n "trap"-voorkoms gee.

[4] Om dinge te vergemaklik, definieer 'n veranderlike in Matlab (bv. `T = 0.1`) en skryf die veranderlike-naam (bv. `T`) in die *Sample time*-veld, in plaas van die waarde (bv. `0.1`). Om die monster-periode te verander is al wat jy hoef te doen om die veranderlike-waarde in Matlab te verander (bv. `T = 0.2`).

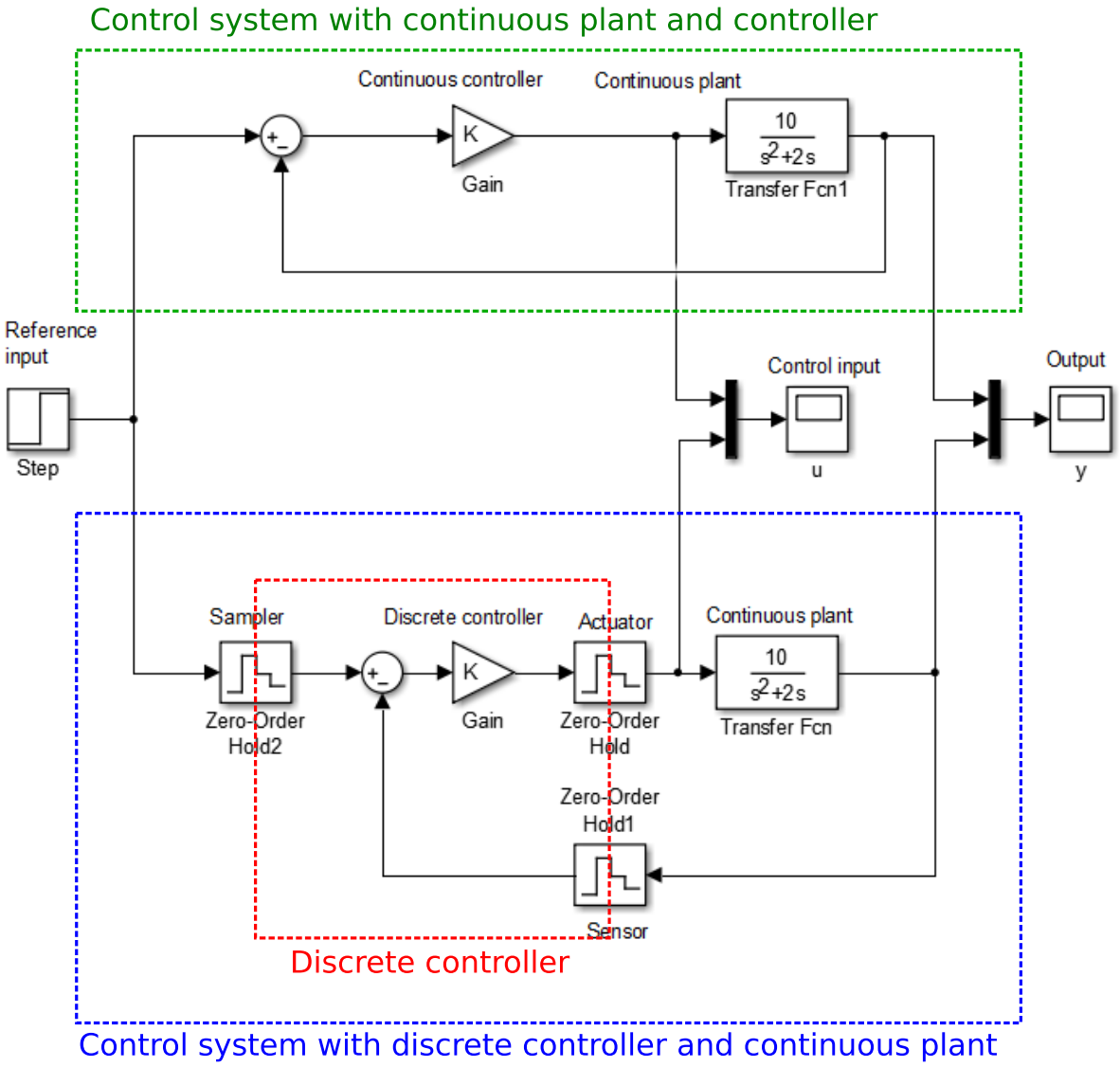

Figure 1: Simulink setup used in this practical

4. Simulate the control system with discrete controller and the one with the continuous controller together. How do the outputs differ? Does the discrete controller satisfy the specifications? Adjust the sample period to several values within the range from 0.01 s to 1 s and compare the outputs. What do you observe when the sample period is very short? How do the responses change when the sample period increases? At which sample period does the system with the discrete controller become unstable? Can you explain the effects? (Hint: Does the sample period introduce a delay?[5] What does a delay mean in terms of the phase of the loop gain?)

   Simuleer die beheerstelsel met die diskrete beheerder en die een met die kontinue beheerder saam. Hoe verskil die uittrees? Bevredig die diskrete beheerder die spesifikasies? Verstel die monster-periode na verskeie waardes in die bereik van 0.01 s tot 1 s en vergelyk die uittrees. Wat sien jy wanneer die monster-periode baie kort is? Hoe verander die gedrag wanneer die monster-periode toeneem? By watter monster-periode raak die stelsel met die diskrete beheerder onstabiel? Kan jy die effekte verklaar? (Wenk: Veroorsaak die monster-periode 'n vertraging?[6] Wat beteken 'n vertraging in terme van die fase van die lus-aanwins?)

5. Reset the sample period to 0.1 s. Increase and decrease the controller gain K of both the discrete and continuous controllers, and observe the change in the outputs. Can you explain the effects? (Hint: think of the effect of the gain on the bandwidth of the closed-loop system and the phase of the loop gain)

   Herstel die monster-periode na 0.1 s. Vermeerder en verminder die beheerder-aanwins K van beide die diskrete en kontinue beheerders, en beskou die verandering in die uittrees. Kan jy die effekte verklaar? (Wenk: dink aan die effek van die aanwins op die bandwydte van die geslotelus-stelsel en die fase van die lus-aanwins)

6. Reset the sample period to 0.1 s and the controller gain to the value designed in Question 1. Apply a sinusoidal signal to the reference input instead of a step signal. Adjust the frequency of the sinusoid to several values within the range from 1 rad/s to 100 rad/s, and observe the differences in the outputs in terms of frequency, relative phase, and amplitude[7]. At which point does the output frequencies start to differ? Can you explain this? (Hint: Nyquist frequency) How does an increase in frequency influence the relative phase of the outputs? Can you explain this?

   Herstel die monster-periode na 0.1 s en die beheerder-aanwins na die waarde soos ontwerp in Vraag 1. Wend 'n sinus-sein op die verwysings-intree aan in plaas van 'n trap-sein. Verstel die frekwensie van die sinus-sein na verskeie waardes in die bereik van 1 rad/s tot 100 rad/s, en beskou die verskille in die uittrees in terme van frekwensie, relatiewe fase, en amplitude[8] By watter punt begin die uittree-frekwensies verskil? Kan jy dit verklaar? (Wenk: Nyquist-frekwensie) Hoe beïnvloed 'n toename in frekwensie die relatiewe fase van die uittrees? Kan jy dit verklaar?

[5] To see this delay, connect both sides of the sampler of the output signal to a Scope; i.e., display both $y(t)$ and $y(k)$ on the same graph.

[6] Om hierdie vertraging te sien, verbind beide kante van die blok wat die uittree monster aan 'n "Scope"-blok, d.w.s. vertoon beide $y(t)$ en $y(k)$ op dieselfde grafiek.

[7] You might have to set the *Max step size* simulation parameter to a small value to ensure an adequate resolution at high frequencies.

[8] Jy sal dalk die *Max step size*-simulasie-parameter na 'n klein waarde moet stel om 'n voldoende resolusie te verseker by hoë frekwensies.