

Ansteuerung Versuchsstand Einfachpendel

1 Echtzeitverarbeitungssystem

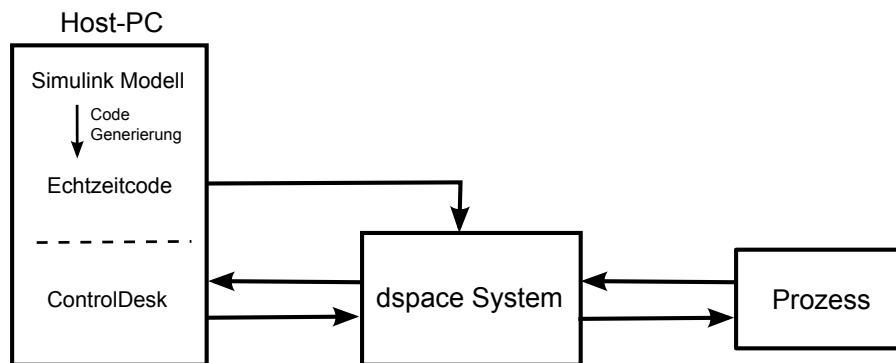


Abbildung 1: Struktur der Echtzeitansteuerung.

Als separates Gerät ausgeführt, übernimmt das dSpace-System, im Folgenden auch Prozessrechner genannt, die gesamte Echtzeitverarbeitung¹ des Prozesses (siehe Abbildung 1). Die Steuerung des Prozessrechners erfolgt über einen Host-PC, der via Ethernet mit ersterem kommuniziert. Über die bestehende Ethernet-Verbindung können mittels der auf dem Host-PC laufenden Software, *ControlDesk*, Prozessdaten in Echtzeit visualisiert werden (siehe Abbildung 2). Darüber hinaus können Reglerparameter im laufenden Betrieb variiert und die unmittelbare Reaktion auf den Prozess beobachtet werden. Ferner lassen sich auf diesem Wege auch Versuchsdaten aufzeichnen und auswerten. Der größte Vorzug des dSpace-Systems besteht jedoch in der automatischen Code-Generierung mit Hilfe des Werkzeugs *Code Generation* von Matlab/SIMULINK, so dass der Anwender keine prozessorspezifische Programmiersprache beherrschen muss.

2 Inbetriebnahme des Versuchsstands

Schalten Sie folgende Geräte ein:

- Computer/Host-PC
- dSpace-Box
- Versuchsstand

3 Login

Loggen Sie sich unter dem Benutzerprofil *praktikum* auf dem Host-PC ein.

¹Die Echtzeitverarbeitung umfasst im Wesentlichen die deterministische Abarbeitung der Aufgaben: Messdatenerfassung, Reglerberechnung, Stellsignalausgabe.

Ansteuerung Versuchsstand Einfachpendel

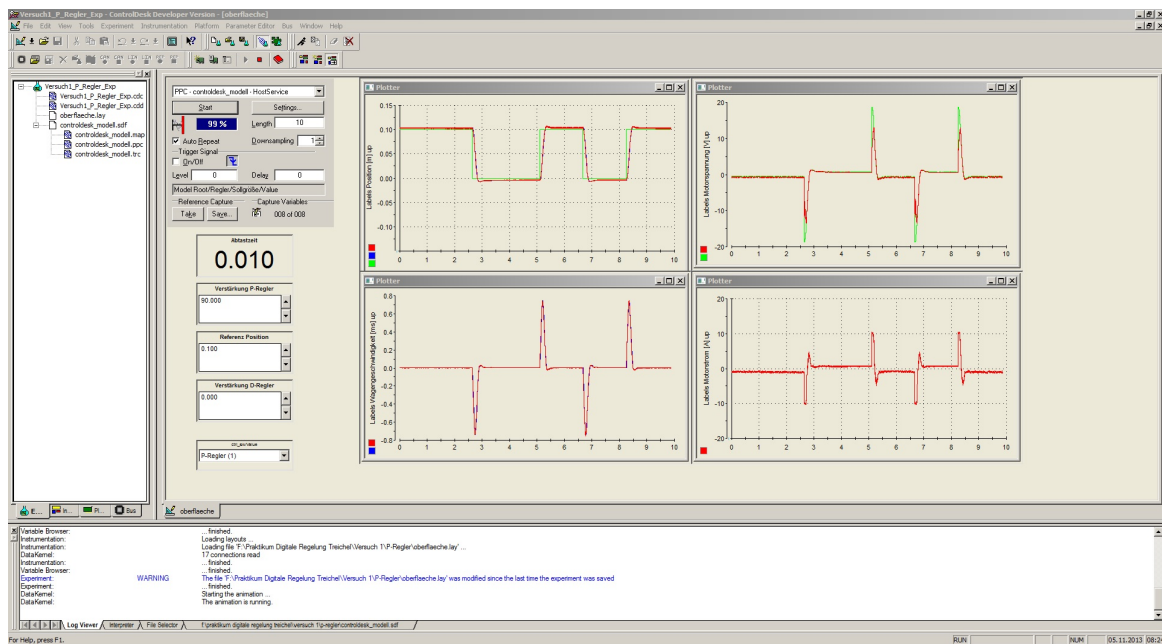


Abbildung 2: Software-Tool *Control-Desk* zur Visualisierung und Steuerung von Prozessvariablen.

4 Matlab/Simulink

Starten Sie nun das Programm Matlab über die Desktopverknüpfung *MATLAB R2011b 32 BIT DSPACE*. Nach kurzer Zeit erscheint ein Fenster mit dem Titel *Select dSPACE RTI Platform Support*, zur Auswahl der verwendeten Echtzeitplattform. Klicken Sie hier auf den Button *RTI1103*. Matlab steht Ihnen nun zur Benutzung bereit. Die für die Praktika DR-1 und DR-2 benötigten Simulink-Modelle befinden sich in den Pfaden:

- F:\Praktikum Digitale Regelung Treichel\Versuch 1\P-Regler bzw.
- F:\Praktikum Digitale Regelung Treichel\Versuch 2\Zustandsregler.

Geben Sie je nach Versuch einen der beiden Pfade als Matlab-Pfad² an. Starten Sie nun das Simulink-Modell *ControlDesk_Modell.mdl*. Daraufhin öffnet sich das Fenster aus Abbildung 3.

Dieses Modell stellt die Schnittstelle zum Prozess dar. Die Blöcke *Messsignale* und *Stellsignale* dienen zur Erfassung der Sensorsignale bzw. zur Ausgabe der korrekten Reglerstellsignale. Dort laufen im Wesentlichen A/D und D/A Wandlungen sowie Skalierungsprozesse ab. Diese Blöcke sind bereits vorgefertigt und müssen nicht bearbeitet werden.

Zu Anschauungszwecken werden im Modell zwei verschiedene Abtastzeiten mit den Variablenbezeichnungen T_s und T_a verwendet. Diese müssen vorab vom Benutzer über das *Command Window* vorgegeben werden. Während $T_s=1e-3$ im Wesentlichen zur Abtastung der Messsignale verwendet

²Der Matlab-Pfad kann über die grafische Oberfläche im Dropdown Menü *Current Folder* oder über den Befehl `cd('F:\Praktikum Digitale Regelung Treichel\...')` eingestellt werden.

Ansteuerung Versuchsstand Einfachpendel

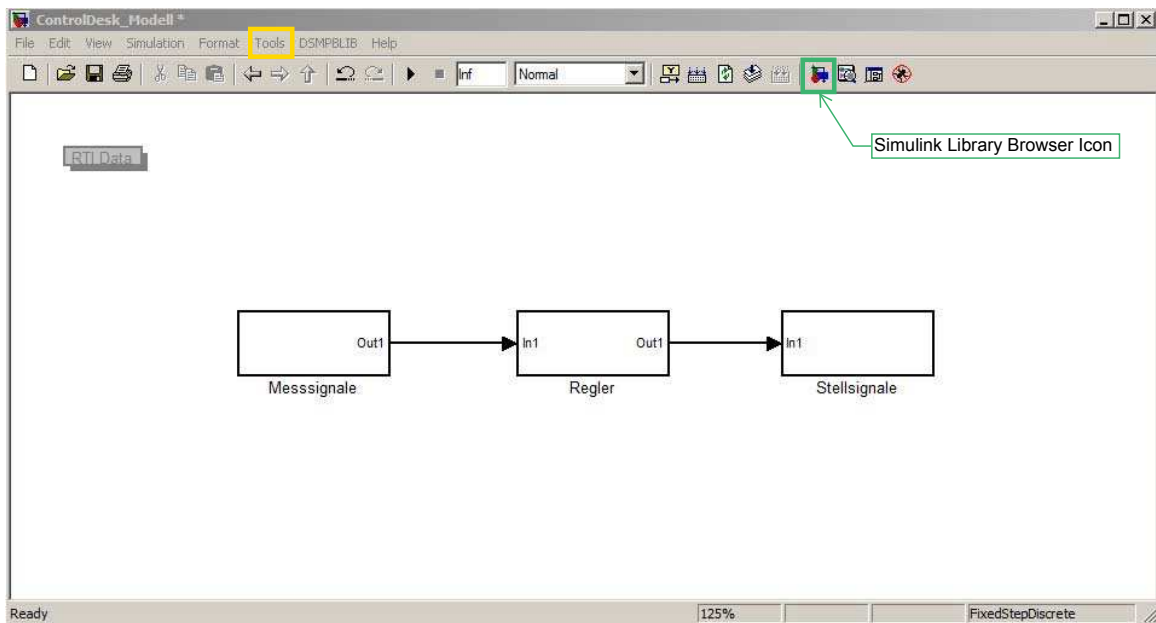


Abbildung 3: Simulink-Modell als Schnittstelle zum Prozess.

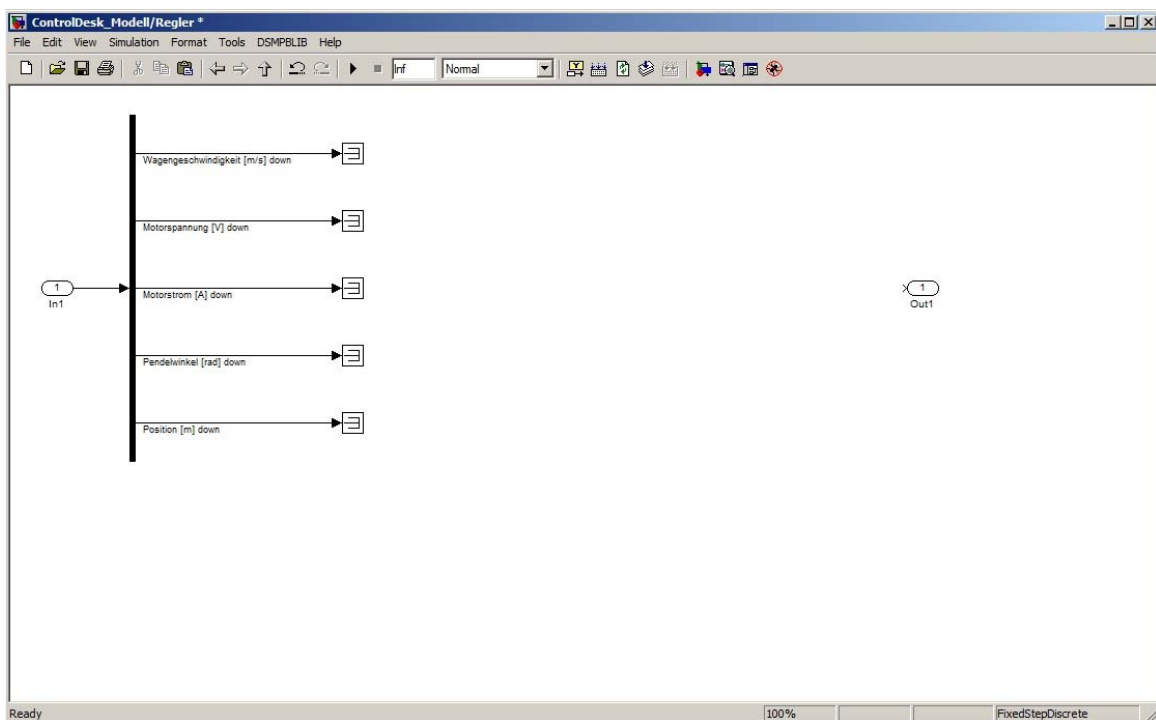


Abbildung 4: Vorgefertigter Regler-Block für den eigenen Reglerentwurf mit Messsignalen und einem Ausgangsport.

wird, stellt T_a die Abtastzeit des diskreten Regelkreises dar. Letztere kann den entsprechenden Versuchsanleitungen entnommen werden.

Öffnen Sie nun den Block *Regler* durch einen Doppelklick. Im Regler-Block werden Messsignale zur Verfügung gestellt, die für eine Regelung genutzt werden können (siehe hierzu Abbildung 4)—der

Ansteuerung Versuchsstand Einfachpendel

Regler fehlt jedoch. Nutzen Sie die benötigten Messsignale und implementieren Sie Ihren Regler lt. Aufgabenstellung.³ Nutzen Sie weiterhin ihre Matlab Skripte aus der Versuchsvorbereitung zum Entwurf der jeweiligen Regler/Reglerparameter. Beachten Sie, dass jedem der vom Regler verwendeten Blöcke die Abtastzeit T_a zugewiesen werden muss.⁴

Nachdem die Abtastzeit sowie die Reglerparameter definiert/berechnet wurden und im Matlab *Workspace* vorliegen, können Sie nun das Simulink-Modell gemäß des Workflows aus Abbildung 1 in echtzeitfähigen C-Code übersetzen. Nutzen Sie hierzu das *Code Generation* Werkzeug unter *Tools* → *Code Generation* → *Build Model* oder alternativ das Tastenkürzel *Ctrl+B*. Hierdurch wird das Modell übersetzt und vollautomatisch auf das dSpace-System geladen. Die einzelnen Schritte des Übersetzungsprozesses lassen sich im Matlab *Command Window* beobachten.

Achten Sie bitte stets darauf, dass Sie sich während des Übersetzungsprozesses im richtigen Dateipfad befinden. Andernfalls können Probleme mit der im Folgenden beschriebenen Visualisierung der Prozessdaten mittels *ControlDesk* auftreten.

5 ControlDesk

Öffnen Sie das Programm *dSPACE ControlDesk 3.7.3* über das gleichnamige Shortcut auf dem Desktop. Gehen Sie dann auf *File* → *Open Experiment* und wählen Sie die Datei

- *Versuch1_P_Regler_Exp.cdx* bzw.
- *Versuch2_Zustandsregler_Exp.cdx*

aus den zugehörigen Pfaden

- `F:\Praktikum Digitale Regelung Treichel\Versuch 1\P-Regler`
- `F:\Praktikum Digitale Regelung Treichel\Versuch 2\Zustandsregler`

aus. Anschließend erscheint die Bedienoberfläche aus Abbildung 5. Klicken Sie auf den Knopf *Animation Mode* (siehe Abbildung 5), um die Visualisierung der Prozessdaten zu starten.

6 Hardware Freigabe

Betätigen Sie den grünen Knopf auf der Bedienoberfläche des Leistungsverstärkers zur Freigabe der Reglerstellsignale. Der Prozess kann nun durch die ControlDesk-Oberfläche gesteuert werden.

³Entnehmen Sie die benötigten Blöcke dem *Simulink Library Browser* (siehe Abbildung 4) und verbinden Sie den Reglerausgang mit dem Ausgangsport *Out1*.

⁴Dies geschieht durch einen Doppelklick auf den jeweiligen Block und das Eintragen der Variable T_a im Eingabefeld *Sample Time*.

Ansteuerung Versuchsstand Einfachpendel

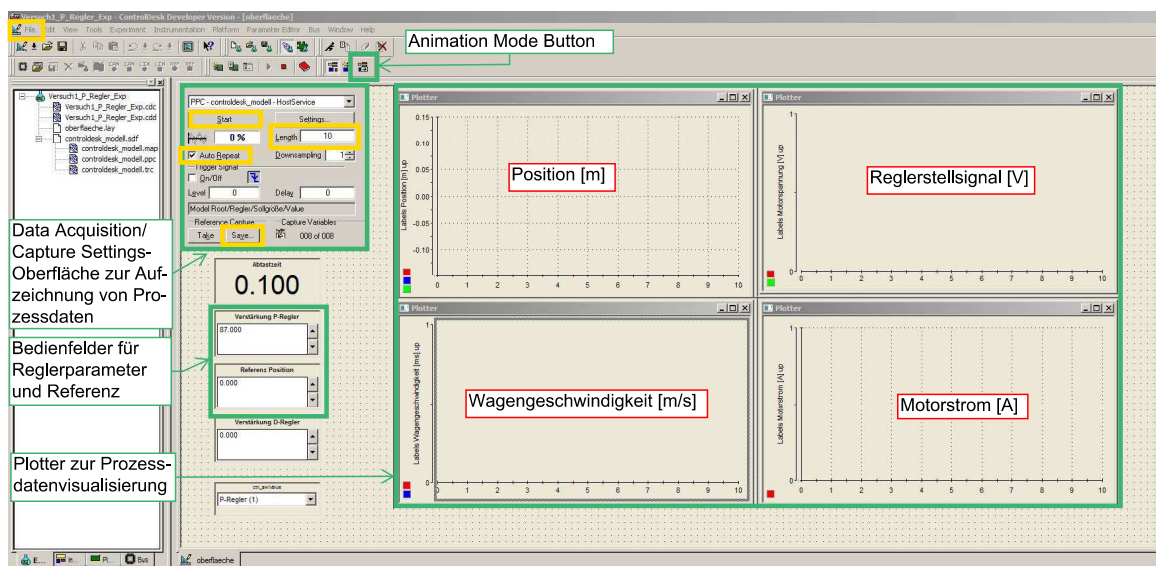


Abbildung 5: ControlDesk-Oberfläche nach dem Öffnen des Experiments *Versuch1_P_Regler_Exp.cdx*.

7 Prozessteuerung

Mit Hilfe von vorgefertigten Eingabefeldern (siehe Abbildung 5) können Parameter wie Sollwerte und Reglerverstärkungen im laufenden Betrieb geändert werden und deren Auswirkung auf den Prozess unmittelbar beobachtet werden.

8 Datenerfassung

Mit Hilfe des *Data Acquisition/Capture Settings*-Bedienfelds (siehe Abb. 5) kann die Aufzeichnung von Messdaten erfolgen. Deaktivieren Sie hierzu zunächst die *Auto Repeat* Funktion und geben Sie eine sinnvolle Versuchsdauer z.B. 10 s im Eingabefeld *Length* ein.

Über den Button *Start*, starten Sie die Aufzeichnung der Messdaten, die anschließend über *Save* als Matlab-File exportiert werden kann. Speichern Sie die Versuchsdaten im Ordner

- `.\Versuchsdaten`.

Verwenden Sie anschließend die vorgefertigte Matlab-Funktion *DataPreProc.m*, die eine Vorverarbeitung der Versuchsdaten vornimmt und diese zusammen mit den im Vorfeld angefertigten Simulationen vergleicht.