Supplementary Material to
Benchmarking Torque Control Strategies for a Torsion based Series Elastic Actuator

Barkan Ugurlu, *Member, IEEE,* Emre Sariyildiz, *Member, IEEE,* Ahmet Talha Kansizoglu, Erim Can Ozcinar, *Student Member, IEEE,* and Sinan Coruk *Student Member, IEEE*

***Abstract*—This is a supplementary material to *Benchmarking Torque Control Strategies for a Torsion based Series Elastic Actuator*. The document includes further technical details regarding the series elastic actuator modeling and the synthesis of the controllers. Furthermore, it provides sufficient information regarding the simulation models (both CJT-based and script-based) and as well as the construction of the hardware setup.**

***Index Terms*—series elastic actuator, torque control, benchmarking.**

I. Modeling of an SEA

An SEA can be represented as a two-inertia system as it is constituted of motor and link ports that are connected via a torsional spring [1]. A reduction gear is placed between the motor side and torsional spring to increase the torque output. The output torque of the actuator is directly related to the spring deflection, therefore, the torque control problem of an SEA can be interpreted as the deflection control problem. The block diagram of an SEA is displayed in Fig. 1; see [2].

In this figure, (J_m, B_m) and (J_l, B_l) are motor and link side inertia and viscous friction coefficients, respectively. Motor side link side angles are θ_m and θ_l, while θ_s is torsional spring deflection. Motor and output torque values are represented by τ_m and τ_s. Torsional spring stiffness, environmental stiffness and damping are denoted with K_s, K_{env} and B_{env}, respectively. The gear ratio is symbolized with N. The motor side and link side disturbances are defined as d_m and d_l. The Laplace operator is s. Using this diagram, the mathematical model of an SEA can be represented via a transfer function written in Laplace domain or in a state-space form. While they both give the same system responses evidently, they differ in the sense of controller design procedures.

A. SEA Modeling in Laplace Domain

Since we have a two-inertia system, motor side and link side dynamics can be represented by using two distinct transfer functions. To this end, the motor angle can be described as below:

$$\theta_m(s) = \frac{1}{J_m s^2 + B_m s}(\tau_m(s) - \frac{K_s}{N}\theta_s - \frac{d_m}{N}). \quad (1)$$

Using (1), the motor side transfer function can be yielded:

$$P_m(s) = \frac{\theta_m(s)}{\tau_m(s) - K_s N^{-1}\theta_s(s)} = \frac{1}{J_m s^2 + B_m s}. \quad (2)$$

In (2), P_m relates θ_m to θ_s and τ_m. By the same token, the link side ankle can be expressed as in the following:

$$\theta_l(s) = \frac{1}{J_l s^2 + (B_l + B_{env})s + K_{env}}(K_s\theta_s - d_l). \quad (3)$$

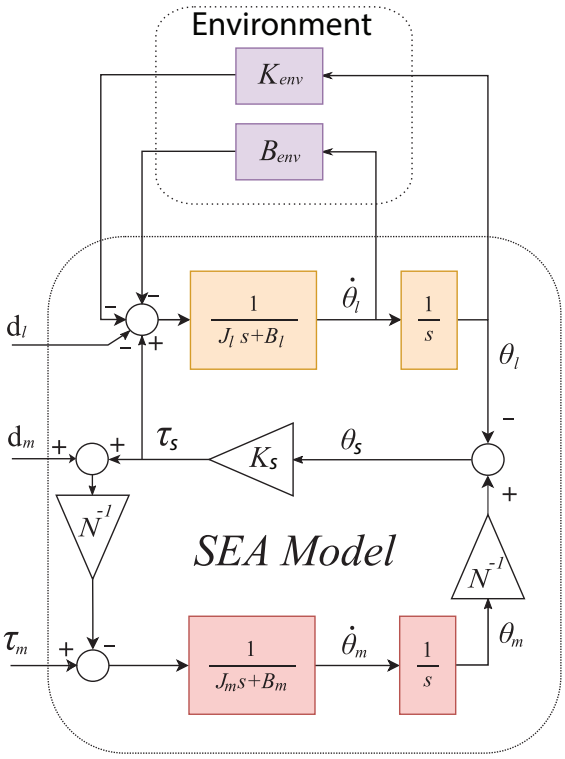

Fig. 1. Block diagram of an SEA unit.

B. Ugurlu, E. C. Ozcinar, S. Coruk, and A. T. Kansizoglu are with the Dept. of Mechanical Engineering, Ozyegin University, 34794 Istanbul, Turkey. (e-mail: {barkan.ugurlu}@ozyegin.edu.tr)

E. Sariyildiz is with the School of Mechanical, Materials, Mechatronics and Biomedical Engineering, University of Wollongong, NSW 25522 Wollongong, Australia. (e-mail: emre@uow.edu.au)

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK), with the project 215E138.

Using (3), the link side transfer function can be obtained:

$$P_l^*(s) = \frac{\theta_l(s)}{K_s\theta_s(s)} = \frac{1}{J_l s^2 + (B_l + B_{env})s + K_{env}} \quad (4)$$

$P_l^*(s)$ is a transfer function which relates θ_l to $K_s\theta_s$. In practice, environmental impedance values are not known and this uncertainty introduces another disturbance source. Thus, we will use the transfer function $P_l(s)$ which is expressed below

$$P_l(s) = \frac{\theta_l(s)}{K_s\theta_s(s)} = \frac{1}{J_l s^2 + B_l s} \quad (5)$$

To obtain the transfer function from the motor torque τ_m to the deflection θ_s, we can simply rewrite the relationship between the deflection and the output torque as follows,

$$\theta_s(s) = \theta_m(s)N^{-1} - \theta_l(s). \quad (6)$$

By substituting the eqs. (2) and (5) into (6), the following are yielded:

$$\begin{aligned} \theta_s(s) &= \frac{\left(\tau_m(s) - K_s N^{-1}\theta_s(s) - d_m N^{-1}\right)P_m(s)}{N} \\ &- (K_s\theta_s(s) - \tilde{d}_l)P_l(s) \quad (7) \\ &= \tau_m(s)\frac{P_m(s)N^{-1}}{1 + K_s N^{-2}P_m(s) + K_s P_l(s)} \\ &- d_m(s)\frac{P_m(s)N^{-2}}{1 + K_s N^{-2}P_m(s) + K_s P_l(s)} \\ &+ \tilde{d}_l(s)\frac{P_l(s)}{1 + K_s N^{-2}P_m(s) + K_s P_l(s)}, \quad (8) \end{aligned}$$

where $\tilde{d}_l$ is the new link side disturbance which additionally includes the omitted environmental effects; $\tilde{d}_l = d_l + K_{env}\theta_l + B_{env}\dot{\theta}_l$. Using (8), the transfer function between the motor torque and the torsional deflection is obtained:

$$P_n(s) = \frac{\theta_s(s)}{\tau_m(s)} = \frac{P_m(s)N^{-1}}{1 + K_s N^{-2}P_m(s) + K_s P_l(s)} \quad (9)$$

B. SEA Modeling in State Space Representation

The dynamics of a general linearized system can be represented in a state space form as follows,

$$\dot{x} = Ax + Bu - F \quad and \quad y = Cx + Du \quad (10)$$

In (10), A is the state matrix, B is the input matrix, C is the output matrix, D is the feed-forward matrix and F is the disturbance matrix. Furthermore, x is the state vector, y is the output vector. The motor torque τ_m is represented by the input u. The states of the system can be chosen as below:

$$x^\intercal = [\theta_m \quad \dot{\theta}_m \quad \theta_l \quad \dot{\theta}_l]. \quad (11)$$

To obtain the deflection as an output, C matrix can be written as follows

$$y = Cx = [N^{-1} \quad 0 \quad -1 \quad 0]x. \quad (12)$$

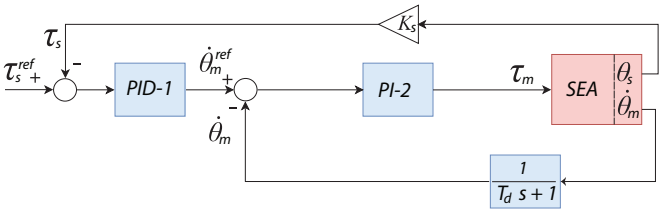

Fig. 2. Block diagram of the Cascaded PID Control with an inner velocity loop.

If we rewrite the dynamic equations, the state matrix A can be obtained as in the following:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K_s}{J_m N^2} & -\frac{B_m}{J_m} & \frac{K_s}{J_m N} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K_s}{J_l N} & 0 & -\frac{K_s}{J_l} & -\frac{B_l}{J_l} \end{bmatrix} \quad (13)$$

The input vector B is displayed as below.

$$B^\intercal = \begin{bmatrix} 0 & \frac{1}{J_m} & 0 & 0 \end{bmatrix}, \quad (14)$$

We may observe that the disturbances d_m and d_l have an effect on the motor and link angles as expressed below.

$$F^\intercal = \begin{bmatrix} 0 & \frac{d_m}{J_m N} & 0 & \frac{d_l + K_{env}\theta_l + B_{env}\dot{\theta}_l}{J_l} \end{bmatrix} \quad (15)$$

The environmental impedance parameters are usually not known. Therefore, they were not included in the state matrix as they were considered in F.

II. Methods: Robust Control Synthesis for an SEA

In this section, we disclose the synthesis of the five torque controllers that are chosen for experimental benchmarking.

A. Cascaded PID Controller

The cascaded controller configuration can be seen in Fig. 2, where the superscript ref denotes the reference. K_{p1}, K_{i1}, and K_{d1} are the proportional, integral and derivative gains of the outer loop respectively. T_d is the time constant of the motor velocity filter and ω_d is the low pass filter frequency for obtaining the rate change of torque measurement. K_{p2} and K_{i2} are the proportional and integral gains of the inner loop.In this control configuration, the outer loop has the the output torque feedback and a PID controller. The output of the PID controller generates the motor velocity reference. The PI controller achieves the velocity control at the inner loop for a given calculated velocity reference. To determine the inequality constraints regarding the controller gains, refer to the stability analysis provided in [3].

B. Cascaded PID Controller with DOB

The performance of the cascaded PID controller is directly linked to the inner loop tracking performance. As the inner loop is mainly responsible for motor velocity tracking, its tracking performance can be increased with the help of a DOB that is implemented to the motor side [4], [5]. The cascaded controller with a motor side DOB configuration can be seen in Fig. 3, where $\hat{d}_m$ is the estimated motor side disturbance and g is the low pass filter of the DOB. The tuning of the controller follows the same rules as in the previous method.

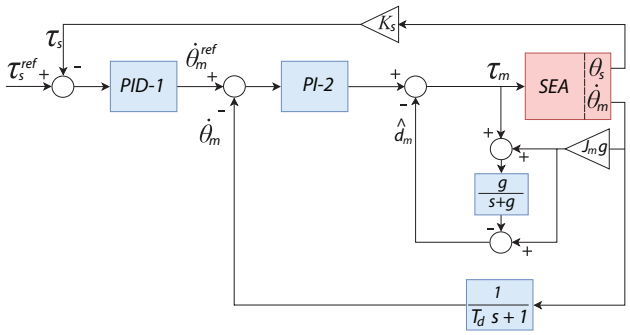

Fig. 3. Block diagram of the Cascaded PID Control with inner velocity loop and DoB.

C. PID with model based feed-forward and DOB

In contrast to the previous method in which DOB is configured solely at the motor side, one alternative way is based on considering the overall SEA transfer function given in eq. (**??**). With the addition of feed-forward and feedback terms to this DOB block, Oh and Kong proposed a robust model-based control algorithm in [2]. High precision tracking is addressed by means of spring deflection feedback. The controller scheme can be seen in Fig. 4.

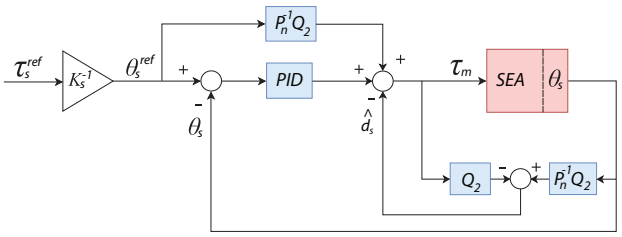

Fig. 4. Block diagram of the PID with model based feed-forward and DoB

In this figure, P_n is the nominal model given in (9). Q_2 is a second-order low-pass Butterworth filter:

$$Q_2(s) = \frac{\omega_c^2}{s^2 + \sqrt{2}\omega_c s + \omega_c^2} \tag{16}$$

Both the feed-forward term and the DOB contain the inverse of the nominal model. To realize P_n^{-1}, a second-order Butterworth filter with a cut-off frequency ω_c is used [6]. The feedback term is the PID controller that is tuned with respect to the nominal plant. In this approach, the disturbance acting on the spring deflection (d_s) can be expressed using (8) and (9):

$$d_s = \frac{\tilde{d}_l P_l(s) - d_m P_m(s) N^{-2}}{P_m(s) N^{-1}} \tag{17}$$

To suppress d_s, a disturbance observer was designed using the inverse of the nominal model (P_n^{-1}) to obtain the estimated deflection disturbance $\hat{d}_s$; see Fig 4. To realize P_n^{-1}, a second-order Butterworth filter with a cut-off frequency ω_c is used; see eq. (16) [6]. Refer to [2] for further details.

D. Sliding mode control with DOB

A sliding mode controller is known to be highly suitable for controlling the systems that suffer from modeling uncertainties and disturbances [7]. Fundamentally, an SMC defines a sliding surface that is constructed by using system states, and the aim of the controller is defined as confining the sliding variable to a certain close neighborhood [8].

Depending on the sign of the sliding variable, the control action switches between different values within a discrete signal profile. The switching action sensitivity and the decay rate of the sliding trajectory can be determined by controller parameters by considering a well-known problem: chattering [9]. To that end, Sariyildiz et al. proposed a sliding mode controller with a disturbance observer to overcome the chattering phenomenon [10]; see Fig. 5.

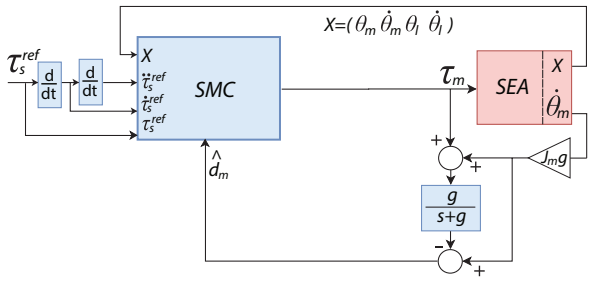

Fig. 5. Block diagram of the Sliding Mode Controller with DOB.

In an attempt to characterize the measured deflection, we slightly modified Sariyildiz's proposed controller scheme so as to associate link torque feedback:

$$\tau_s^{ref} = K\theta_s^{ref}, \tag{18}$$

In (18), τ_s^{ref} and θ_s^{ref} are the desired output torque and deflection. The error can be defined as follows:

$$e = \theta_s^{ref} - \theta_s \tag{19}$$

The first and the second derivatives of the error are obtained to construct the sliding surface.

$$\dot{e} = \dot{\theta}_s^{ref} - \dot{\theta}_s \tag{20}$$

$$\ddot{e} = \ddot{\theta}_s^{ref} - \ddot{\theta}_s \tag{21}$$

While the second derivative of the reference signal ($\ddot{\theta}_s^{ref}$) is easy to yield via differentiating τ_s^{ref}, $\ddot{\theta}_s$ may be obtained from the dynamic equations:

$$\ddot{\theta}_s = \frac{\ddot{\theta}_m}{N} - \ddot{\theta}_l \tag{22}$$

If we rewrite the equation (22), the following are yielded.

$$\begin{aligned}\ddot{\theta}_s &= -\left(\frac{K_s}{J_m N^2} + \frac{K_s}{J_l}\right)\theta_s - \left(\frac{J_l B_m - J_m B_l}{J_m J_l N}\right)\dot{\theta}_m \\ &+ \frac{\tau_m}{J_m N} - \frac{d_m}{J_m N^2} + \frac{\tilde{d}_l}{J_l} - \frac{B_l}{J_l}\dot{\theta}_s \end{aligned} \tag{23}$$

The sliding surface can be defined by using e and $\dot{e}$ as follows:

$$\sigma = \dot{e} + ce \tag{24}$$

$$\dot{\sigma} = \ddot{e} + c\dot{e} \tag{25}$$

In defining the surface, c denotes the convergence rate. Plugging (25) and (25) into (23) $\dot{\sigma}$ can be derived:

$$\dot{\sigma} = \ddot{\theta}_s^{ref} - \ddot{\theta}_s + c(\dot{\theta}_s^{ref} - \dot{\theta}_s) \tag{26}$$

$$\begin{aligned}\dot{\sigma} &= \ddot{\theta}_s^{ref} + \left(\frac{K_s}{J_m N^2} + \frac{K_s}{J_l}\right)\theta_s + \left(\frac{J_l B_m - J_m B_l}{J_m J_l N}\right)\dot{\theta}_m \\ &- \frac{\tau_m}{J_m N} + c(\dot{\theta}_d^{ref} - \dot{\theta}_s) + \frac{B_l}{J_l}\dot{\theta}_s + \frac{d_m}{J_m N^2} - \frac{\tilde{d}_l}{J_l}\end{aligned} \tag{27}$$

Finally, the control action τ_m can be designed by using eq. (27) as in the following, where ρ is the controller gain:

$$\begin{aligned}\tau_m &= J_m N\left(\rho sign(\sigma) + \ddot{\theta}_s^{ref} + \left(\frac{K_s}{J_m N^2} + \frac{K_s}{J_l}\right)\theta_s \right. \\ &+ \left(\frac{J_l B_m - J_m B_l}{J_m J_l N}\right)\dot{\theta}_m + \frac{B_l}{J_l}\dot{\theta}_s + c(\dot{\theta}_s^{ref} - \dot{\theta}_s) \\ &+ \left.\frac{\hat{d}_m}{N}\right)\end{aligned} \tag{28}$$

In (28), $\hat{d}_m$ indicates the estimated motor side disturbance. In this case $\dot{\sigma}$ takes the following form:

$$\dot{\sigma} = -\rho sign(\sigma) + \frac{d_m - \hat{d}_m}{J_m N^2} - \frac{\tilde{d}_l}{J_l} \tag{29}$$

In (29), the second term stands for the motor side estimation error, while the third term indicates the aggregated link side disturbance including the environmental effects.

E. Differential Flatness Control with DOB

Differential flatness control was proposed by Fliess et al. mainly for motion planning applications of the systems that are differentially flat. A system is called differentially flat if all system states, inputs and outputs are expressible with the different combinations of flat output variables and its derivatives that are not differentially coupled [11]. While useful in handling nonlinearities, differential flatness control itself may not possess the robustness property as it is usually sensitive to uncertainties [12].

In order to benefit the useful properties of differential flatness while handling uncertainties, Sariyildiz and Yu proposed a high-order disturbance observer in state space and combined it with a DF-based controller design [12]; see Fig. 6. The general dynamic model of the SEA can be expressed in the state-space:

$$\dot{x} = f(x, u) \tag{30}$$

where x is the state vector and u is the system input. If the system is differentially flat, there exists a function ϕ such that an output vector can be expressed as in the following.

$$y_{DF} = \phi(x, u, \dot{u}, \ddot{u}, ..., u_n). \tag{31}$$

In (31), y_{DF} is the differentially flat output variable, and n is a finite integer. The flatness theory states that the input of the system and the state vector could be expressed as a combination of differentially flat output variable and its derivatives as in the following, where r is a finite integer [12].

$$x = \kappa_x(y_{DF}, \dot{y}_{DF}, \ddot{y}_{DF}, ..., y_{DF}^{(r)}) \tag{32}$$

$$u = \kappa_x(y_{DF}, \dot{y}_{DF}, \ddot{y}_{DF}, ..., y_{DF}^{(r+1)}) \tag{33}$$

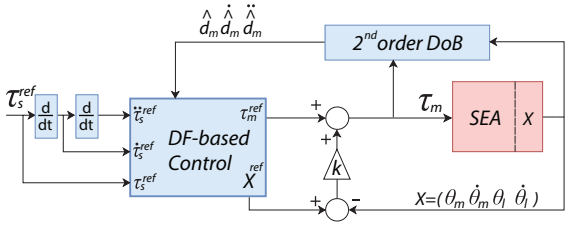

Fig. 6. Block diagram of the DF+DoB controller. To estimate the disturbances d_m and d_l and their derivatives, a second order DOB was implemented; see [12].

Since a DF-based controller is highly sensitive to uncertainties, we additionally considered nominal plant dynamics in the model. Thus, final equations of the DF controller are as below,

$$x^{ref} = \begin{bmatrix} J_l\ddot{y}_{DF} + \mu_1\dot{y}_{DF} + \mu_2 y_{DF} + \mu_3 \\ KN^{-2}y_{DF} \end{bmatrix} \tag{34}$$

$$u^{ref} = \rho_1\dddot{y}_{DF} + \rho_2\ddot{y}_{DF} + \rho_3\dot{y}_{DF} + \rho_4\dot{y}_{DF} + d_{total} \tag{35}$$

where

$$\mu_1 = (B_l+B_{env}) \quad (36)$$
$$\mu_2 = N^{-1}(K+K_{env}) \quad (37)$$
$$\mu_3 = d_l N^2 K^{-1} \quad (38)$$
$$\rho_1 = J_m J_l \quad (39)$$
$$\rho_2 = J_m(B_l+B_{env})+J_l B_m \quad (40)$$
$$\rho_3 = KN^{-2}J_l+(K+K_{env})N^{-1}J_m + B_m(B_l+B_{env}) \quad (41)$$
$$\rho_4 = KN^{-2}(B_l+B_{env})+N^{-1}(K+K_{env})B_m \quad (42)$$
$$d_{total} = d_m+d_l+B_m N^2 K^{-1}\dot{d}_l + J_m N^2 K^{-1}\ddot{d}_l \quad (43)$$

Desired torque reference is as below:

$$\tau_s^{ref} = K\big[J_l N^{-1}\ddot{y}_{DF} + (B_l+B_{env})N^{-1}\dot{y}_{DF} + K_{env}N^{-2}y_{DF}\big] \quad (44)$$

Therefore, the desired differentially flat output reference can be defined as a function of the desired output torque τ_s^{ref} by using the eqs. (34)-(43).

$$\ddot{y}_{DF}^{ref} = NJl^{-1}\big[K^{-1}\tau_s^{ref} - d_l N K^{-1} - N^{-2}K_{env}y_{DF} - (B_l+B_{env})N^{-1}\dot{y}_{DF}\big] \quad (45)$$

The motor torque command τ_m can be calculated via a state feedback with a pole placement in the original controller design. Alternatively, the state feedback can be replaced with PD controllers as depicted in Fig. 7.

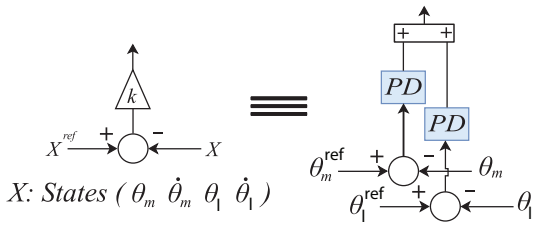

Fig. 7. State feedback versus double PD controller

Therefore, the state feedback can be tuned with a proportional K_p and derivative gain K_d. The motor and link side PD controller gains are respectively defined as K_p^m, K_d^m and K_p^l, K_d^l. The relationship between these parameters is given as below.

$$K_p^m = -K_p; \quad K_d^m = K_d \quad (46)$$
$$K_p^l = NK_p; \quad K_d^l = -NK_d \quad (47)$$

III. Simulation Setups

A. Compliant Joint Toolbox Simulation Model

A realistic model of our hardware setup was constructed using the Compliant Joint Toolbox (CJT) [13]. Furthermore, all five controllers were also synthesized using CJT for our constructed SEA model. All the necessary files and a short guideline can be accessed via the following GitHub repository:

https://github.com/new

B. Script-based Simulation Model

Another simulation model of our hardware setup was constructed using a MATLAB script with no Simulink file. The main motivation behind this approach was to place the model and the controllers into Code Ocean[1] such that potential readers could examine and experiment with the controllers in question rather easily, without downloading any files and/or programs. These files can be accessed via the following Code Ocean repository:

https://doi.org/10.24433/CO.7392529.v1

IV. Hardware Setup

The general structure of electronic hardware is illustrated in Fig. 8. The main controller is a Raspberry Pi 3 Model B, that has a real-time operating system achieved via Preempt-RT patch [2]. A Maxon 70/10 motor drive was employed with current control configuration. Analog motor drive commands were sent via an D/A converter (Waveshare 12-bit) which can communicate with the Raspberry Pi unit via SPI (Serial Peripheral Interface). The two 23-bit encoders (Broadcom AS38-H39E-S13S), namely position encoder and torque encoder, were utilized to measure motor side angle and spring deflection, respectively. A custom-made PCB was used to transfer encoder data to Raspberry Pi 3 unit via SPI. The circuit schematic and PCB layout were included as the supplementary material.

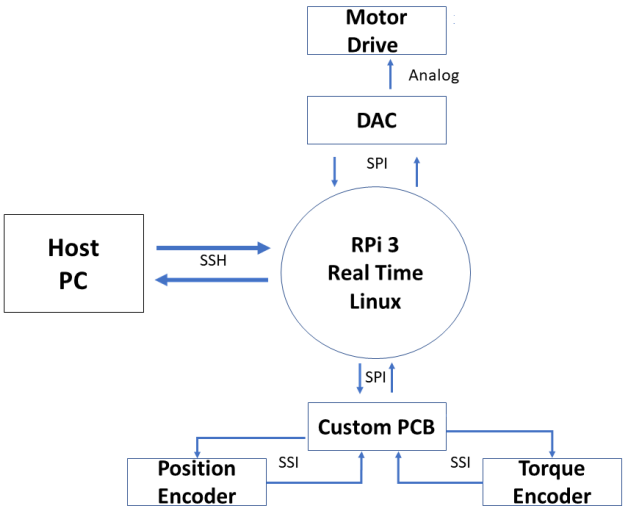

Fig. 8. The block diagram and communication interfaces used in our hardware setup.

[1] https://codeocean.com/
[2] https://rt.wiki.kernel.org/

REFERENCES

[1] Y. Yokokura and K. Ohishi, "Fine load-side acceleration control based on torsion torque sensing of two-inertia system," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 1, pp. 768–777, Jan. 2020.

[2] S. Oh and K. Kong, "High-precision robust force control of a series elastic actuator," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 71–80, Feb. 2017.

[3] H. Vallery, R. Ekkelenkamp, H. van der Kooij, and M. Buss, "Passive and accurate torque control of series elastic actuators," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2007.

[4] K. Ohnishi, M. Shibata, and T. Murakami, "Motion control for advanced mechatronics," *IEEE/ASME Transactions on Mechatronics*, vol. 1, no. 1, pp. 56–67, Mar. 1996.

[5] W. Roozing, J. Malzahn, D. G. Caldwell, and N. G. Tsagarakis, "Comparison of open-loop and closed-loop disturbance observers for series elastic actuators," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2016.

[6] J. S. Mehling, J. Holley, and M. K. O'Malley, "Leveraging disturbance observer based torque control for improved impedance rendering with series elastic actuators," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Sep. 2015.

[7] J. Bae, K. Kong, and M. Tomizuka, "Gait phase-based smoothed sliding mode control for a rotary series elastic actuator installed on the knee joint," in *Proceedings of the 2010 American Control Conference*. IEEE, Jun. 2010.

[8] V. Utkin, "Sliding mode control design principles and applications to electric drives," *IEEE Transactions on Industrial Electronics*, vol. 40, no. 1, pp. 23–36, Feb. 1993.

[9] A. Calanca, L. Capisani, and P. Fiorini, "Robust force control of series elastic actuators," *Actuators*, vol. 3, no. 3, pp. 182–204, Jul. 2014.

[10] E. Sariyildiz, R. Mutlu, and H. Yu, "A sliding mode force and position controller synthesis for series elastic actuators," *Robotica*, vol. 38, no. 1, pp. 15–28, Apr. 2019.

[11] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, "Differential flatness and defect: an overview," *Banach Center Publications*, vol. 32, no. 1, pp. 209–225, 1995.

[12] E. Sariyildiz and H. Yu, "A robust force controller design for series elastic actuators," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Sep. 2017.

[13] J. Malzahn, W. Roozing, and N. Tsagarakis, "The compliant joint toolbox for matlab: An introduction with examples," *IEEE Robotics & Automation Magazine*, vol. 26, no. 3, pp. 52–63, Sep. 2019.