
Electronic Thesis and Dissertation Repository

1-11-2021 2:15 PM

Design, Development, and Evaluation of Customized Electronics for Controlling a 5-DOF Magneto-Rheological Actuator Collaborative Robot

Ziqi Yang, *The University of Western Ontario*

Supervisor: Kermani, Mehrdad R., *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Master of Engineering Science degree in Electrical and Computer Engineering

© Ziqi Yang 2021

Follow this and additional works at: <https://ir.lib.uwo.ca/etd>



Part of the [Controls and Control Theory Commons](#), [Electrical and Electronics Commons](#), [Electromagnetics and Photonics Commons](#), [Electro-Mechanical Systems Commons](#), [Electronic Devices and Semiconductor Manufacturing Commons](#), and the [VLSI and Circuits, Embedded and Hardware Systems Commons](#)

Recommended Citation

Yang, Ziqi, "Design, Development, and Evaluation of Customized Electronics for Controlling a 5-DOF Magneto-Rheological Actuator Collaborative Robot" (2021). *Electronic Thesis and Dissertation Repository*. 7622.

<https://ir.lib.uwo.ca/etd/7622>

This Dissertation/Thesis is brought to you for free and open access by Scholarship@Western. It has been accepted for inclusion in Electronic Thesis and Dissertation Repository by an authorized administrator of Scholarship@Western. For more information, please contact wlsadmin@uwo.ca.

Abstract

In recent years, Magneto-Rheological (MR) fluids has been used in various fields such as robotics, automotive, aerospace, etc. The most common use of the MR fluids is within a clutch-like mechanism, namely an MR clutch. When mechanical input is coupled to the input part of the MR clutch, the MR clutch provides a means of delivering this mechanical input to its output, through the MR fluids. The combination of the mechanical input device and the MR clutch is called an MR actuator. The MR actuator features inherently compliance owing to the characteristic of the MR fluids while also offering higher torque-to-mass and torque-to-inertia ratios over common actuators. As such, MR actuators are suitable candidates for human-safe and collaborative robots.

The goal of this study is to design, develop and test customized electronic drivers that are compact and powerful to enable effective low-level control of the robot joints. The electronic drivers are responsible for sensor data processing, between-joint communication, supplying electric power, and executing control actions. The hardware design is optimized to handle transient current and voltage, and dissipate heat generated by components. Moreover, software development is based on $\mu C/OS-II$ real-time operating system to handle multiple time-critical tasks and to guarantee the stability and effectiveness of robot control system. A series of experiments are conducted to validate the designed hardware and software systems, and evaluate their performance.

Keywords: Collaborative Robot, Magneto-Rheological Actuator, MR Clutch, Real-time Operating System, $\mu C/OS - II$

Summary for Lay Audience

Magneto-Rheological (MR) fluids are smart materials that have increasingly been used in developing essential components of the robot, automotive, aerospace, etc. Its rheological properties change fast and reversible under the effect of an external magnetic field. In the field of robot development, MR fluids is commonly used to develop the MR actuator. It can deliver the mechanical input to its output through the MR fluids. The difference between the MR actuator and the commonly seen electric actuator is that the MR actuator features higher performance and inherently compliance. As such, MR actuators are suitable candidates for human-safe collaborative robots. Based on that, we are developing 5 degrees of freedom collaborative robot.

This dissertation focuses on design a compact and powerful electronic driver to enable effective low-level control on the joints of a human-robot-safe collaborative robot. The hardware design is optimized to handle extreme situations. The software design is based on a real-time operating system to guarantee the instantaneity and effectiveness of the robot control system. The control system for controlling the entire robot is constructed by building the connection between electronic drivers. Experiments are conducted to evaluate the performance of the robot and software/hardware systems.

Dedication

I dedicate this dissertation to my family, especially to my grandfather, Ruichun Yang, who has been my mentor since I was a child. Always stay true to the mission and forge ahead.

Acknowledgments

I would like to express my gratitude to my supervisor Dr. Mehrdad R. Kermani for giving me the opportunity to be a part of his research team and I would like to thank his patience and his expertise in control and robotics. I would also like to thank my lab mates, Sergey Pisetskiy, Shuwei Qiu, Navid Feizi, Moteaal Asadi Shirzi for sharing their experience and knowledge with me.

Contents

Abstract	ii
Summary for Lay Audience	iii
Dedication	iv
Acknowledgments	v
List of Figures	ix
List of Tables	xii
1 Introduction	1
1.1 Background	1
1.2 Motivation	4
1.3 Objectives	7
1.3.1 Main Contributions	9
1.4 Thesis Outlines	10
2 Design and Development of Embedded Electronic Drivers for a 5-DOF MR Clutch Based Collaborative Robot	12
2.1 Introduction	13
2.2 Magneto-Rheological Clutch	15
2.3 Electronic Design	16
2.3.1 Component Selection	19
2.3.2 Circuit Design	20

Transient-voltage-suppression Diode	21
Protection Capacitors	24
2.3.3 Hardware Verification	29
2.3.4 Schematic and Thermal Performance	33
2.3.5 Sensors	37
2.4 Conclusions	38
3 Firmware Development of a 5-DOF MR Clutch Based Collaborative Robot	40
3.1 Introduction	41
3.2 STM32 and RTOS Platform	43
3.3 μ C/OS-II Real-time Operating System	44
3.3.1 Clock System	45
3.3.2 Task Structure	45
3.3.3 Startup Process of the RTOS	45
3.3.4 Task Scheduling	46
3.4 The Firmware Design of Driver Unit	51
3.4.1 System Operation Process	52
3.4.2 Tasks	52
3.4.3 Data Processing and Control Algorithms	55
Hall Sensor	55
Encoder	57
ADC	60
Communication Protocol	60
Position Control	66
Current Control	68
Trajectory Generation	69
3.5 Experimental Results	70
3.5.1 Position Control Experiments	71
Sinusoidal Signal Tracking	71
Trajectory Tracking	73

3.5.2	Open-loop Torque Experiments	75
	Step response	75
	Controllability of Output Torque	77
3.6	Conclusions	78
4	Conclusion and Future Works	79
4.1	Conclusion	79
4.2	Future Works	80
	Construction of MR Clutch Model and Closed-loop Torque Control . . .	80
	Overall Control of the 5-DOF Robot	82
	Collaborative Tasks	82
	Learn to Collaborate from Human Demonstrations	82
	Bibliography	83
	Curriculum Vitae	91

List of Figures

1.1	MR fluid representation (a)without and (b) with the applied magnetic field. . . .	2
1.2	MR fluids states; (a) off-state and (b) on-state.	2
1.3	Relationship between yield stress and magnetic field.	3
1.4	KUKA LBR iiwa 14 Robot.	5
1.5	Schematic of the electronics of the 5-DOF collaborative robot.	8
1.6	Visualisation of the 5-DOF robot.	9
2.1	Taxonomy of failures	13
2.2	DASA actuation concept.	14
2.3	Cross-section of a multidisk MR clutch.	16
2.4	The location of the electronic driver inside the housing of Joint 2.	19
2.5	Schematic of voltage change.	21
2.6	Cross-section of the coil.	22
2.7	Transient voltage suppression diode I/V curve.	24
2.8	Frequency characteristic of ceramic capacitors.	27
2.9	ESR frequency characteristic of capacitors.	27
2.10	Host-to-board connection.	29
2.11	Ripple voltage of voltage conversion system; (a) 36V PSU ripple voltage, (b) 5V regulator's ripple voltage, (c) 3.3V regulator's ripple voltage.	30
2.12	Startup procedure of voltage conversion system; (a) 36V PSU, (b) 5V regulator, (c) 3.3V regulator.	31
2.13	The response of voltage conversion system when supply the coil with a 5 Hz, 1 A square wave; (a) 36V PSU, (b) 5V output, (c) 3.3V output.	32
2.14	Hardware diagram.	34

2.15	Three dimension view of the electronic driver; (a) Top layer, (b) Bottom layer. .	34
2.16	2-layer PCB junction-to-ambient thermal resistance vs copper area.	36
2.17	Zonal section view of Joint 2.	38
2.18	Pictures of the electronic driver; (a) Top layer, (b) Bottom layer.	39
3.1	Principle of Preemptive kernel.	42
3.2	Clock system of STM32.	44
3.3	Task structure.	46
3.4	Task states.	47
3.5	Concept of task scheduling.	47
3.6	External interrupt.	48
3.7	Task-wise communication.	49
3.8	Time delay.	50
3.9	Procedures of task scheduling.	50
3.10	Real-time system architecture.	51
3.11	System flowchart.	53
3.12	Task, (a) program sequence chart, (b) program context.	56
3.13	SENT Frame.	57
3.14	ASII Data Frame.	58
3.15	CRC-16 algorithm.	59
3.16	Differential output.	60
3.17	CAN bus.	62
3.18	CAN data frame.	62
3.19	Bus arbitration.	62
3.20	Identifier mask mode.	64
3.21	CAN bus communication network.	65
3.22	CAN bus data frame.	65
3.23	Current control scheme.	68
3.24	Cubic polynomial trajectory.	70
3.25	Transmission mechanism.	70

3.26	Joint 2 position tracking of a 0.25Hz sinusoidal reference signal.	71
3.27	Joint 2 position tracking of a 0.5Hz sinusoidal reference signal.	72
3.28	Joint 2 position tracking of a 1Hz sinusoidal reference signal.	72
3.29	Joint 1 trajectory tracking.	73
3.30	Joint 2 trajectory tracking.	74
3.31	Joint debugging of Joint 1 and Joint 2 trajectory tracking.	74
3.32	Error signal of joint debugging of Joint 1 and Joint 2 trajectory tracking.	75
3.33	Experiment setup.	76
3.34	Diagram of the current control experiments.	76
3.35	Transient performance of the joint 2 MR clutch when subjects to 0 A to 0.4 A input current.	77
3.36	Current tracking of a 0.5 Hz sinusoidal reference signal.	77
3.37	Current tracking of a 1 Hz sinusoidal reference signal.	78
4.1	Hysteresis behaviour of the MR clutch when applying a 1 Hz sinusoidal refer- ence signal.	80
4.2	Hysteresis behaviour of the MR clutch when applying a 5 Hz sinusoidal refer- ence signal.	81
4.3	Proposed control scheme of the torque control.	81

List of Tables

1.1	General properties of MR fluids	3
2.1	Design requirements of electronic driver	17
2.2	Key components table	20
2.3	Information of the second joint coil	23
2.4	Parameters of thermal performance	36
2.5	Thermal performance tests	37
3.1	Task information	54
3.2	Features of communication protocols	63

Chapter 1

Introduction

1.1 Background

Magneto-Rheological (MR) fluids are smart materials characterized by fast, tuneable and reversible changes of their rheological properties under application of magnetic fields [36]. MR fluids consist of dispersions of micron-sized particles of magnetizable materials dispersed in a liquid, usually a type of oil. It often contains surfactants, oleic acid, tetramethylammonium hydroxide, for example. These surfactants serve to decrease the rate of ferroparticle settling, of which a high rate is an unfavorable characteristic of MR fluids. The ideal MR fluid would never settle, but developing this ideal fluid is as highly improbable as developing a perpetual motion machine according to our current understanding of the laws of physics [73]. The application of a magnetic field results in the magnetization of the dispersed particles, which consequently experience attractive forces, giving rise to the formation of particle structures that resist to the flow [37]. This means without magnetic field action, the fluid behaves as a regular fluid with properties near to those of the carrier fluid of which it is made of. In the presence of a magnetic field, the MR fluids start behaving like a semisolid as a function of the field intensity [17, 26] (see Fig. 1.1 and 1.2).

This change is completely reversible: when the magnetic field is removed, the MR fluid behaves like the carrier fluid again. The condition in which the MR fluid is subjected to the action of an external magnetic field is called “on-state”, while the condition in which the MR fluid is not activated by the magnetic field is called “off-state” [42]. Both the activation and the

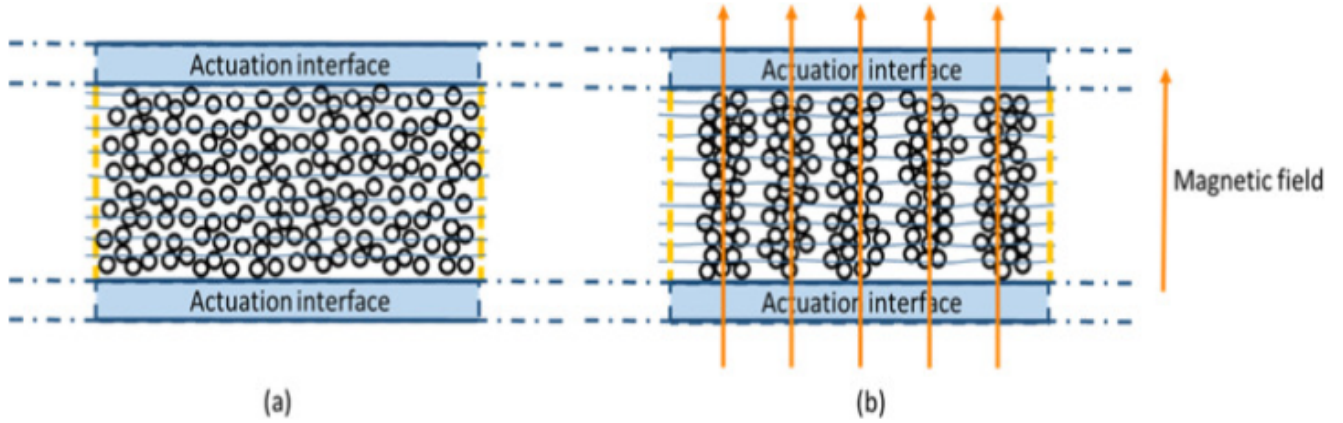


Figure 1.1: MR fluid representation (a) without and (b) with the applied magnetic field.

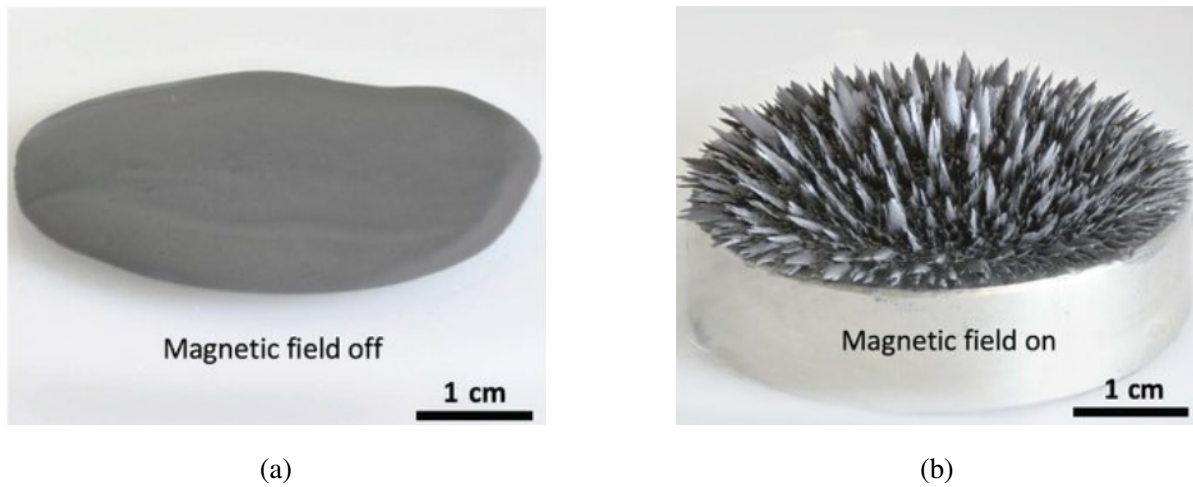


Figure 1.2: MR fluids states; (a) off-state and (b) on-state.

deactivation of the fluids are completed within milliseconds after the field is turned on or off [16]. In on-state, the fluid remains at rest when it is subjected to shear stresses up to a certain limit (shear yield stress), after which it behaves as viscous fluid under the action of shearing [2]. This phenomenon can be utilized in various engineering applications such as car suspension hydraulic valves, and brake mechanism to name a few [21]. However, it took several decades for the MR fluids to be widely used after Jacob Rabinow developed the MR fluid and patented the initial MR fluids devices back in 1940s [52].

MR fluids can be represented by the viscoplastic Bingham model, where the most important parameter is the strong dependence of the shear yield stress on the magnetic field (see the Fig.

1.3, adopted from manufacturer datasheet [19]). As a result of this dependence, the MR fluids

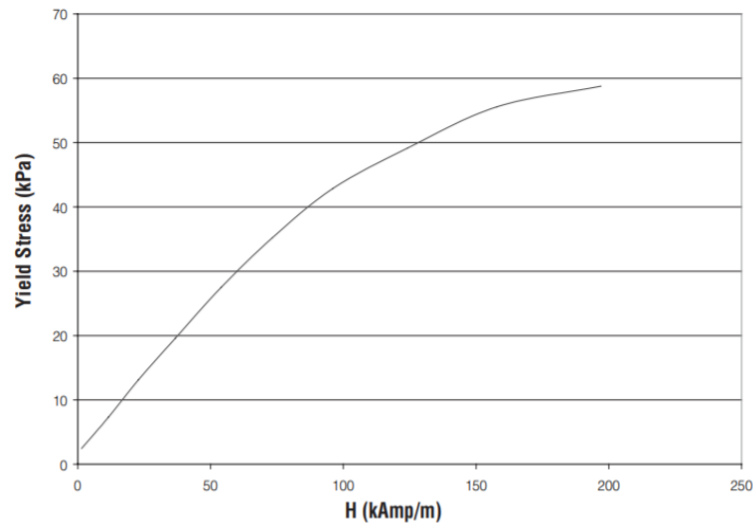


Figure 1.3: Relationship between yield stress and magnetic field.

exhibit several orders of magnitude increase in their apparent viscosity. MR fluids can be utilized by embedding them in a structure whose properties are to be controlled [2], Table 1.1 provides the general properties of MR fluids. Lord Corp has been developing rotary brake

Table 1.1: General properties of MR fluids

Property	Typical value
Work Temperature	(-50) to (150)°C
Initial Viscosity	0.2 - 0.3 Pa-s (at 25°C)
Magnetic Field Strength	150-250 kA/m
Density	3 - 4 g/cm ³
Reaction Time	Few milliseconds
Yield Point	50-100 kPa

based on this concept, which is a controllable rotary resistance system that is smooth acting, compact and consumes low power. They are simple and easy to control, which makes them a very cost-efficient model for a wide variety of applications that range from controllable exercise

equipment to precision active tension control [21].

If we think the opposite way, it is obvious that MR fluids can also offer advantages as actuators. Around twenty years ago, Takesue developed an actuator using MR fluids that featured high torque to mass and torque to inertia ratios with simple interface, compared to available electric actuators at the time [68]. Technological advances are a driving force for further development MR device in recent years. For example, MR actuators were employed in developing rehabilitation robots and exoskeletons [12, 13, 31, 54, 22, 14, 30].

An MR actuator can function as a brake or clutch according to the working states of the devices. When adjustable passive torque is preferred, the MR actuator will work as a brake, thus only very low power is consumed, when active torque is required, the MR actuator will work as a clutch [12]. To further exploit potential of Magneto-rheological actuation, Shafer and Kermani proposed a new actuation concept called, Distributed Active Semi-Active (DASA) actuation, and validated it through experiments [56]. In the following years, the group developed several prototypes including a 1-DOF [55], 2-DOF [85], 3-DOF [57], and most recently a 5-DOF antagonistically actuated manipulators [44, 43]. The success in developing these manipulators has proven that it is possible to achieve a high degree of safe interactions (with humans) with inherent compliance while also maintain the same level of performance as that achieved by stiff joint-torque controlled manipulators [55].

1.2 Motivation

Over the last two decades, the foundations for physical human–robot interaction (pHRI) have evolved from successful developments in mechatronics, control, and planning, leading toward safer lightweight robot designs and interaction control schemes that advance beyond the current capacities of existing high-payload and high precision position-controlled industrial robots [23]. Recently, a new generation of robots has appeared on the robotic landscape, including those qualified for collaborative tasks. This trend brings us closer to the long-term goal of safe, seamless physical human–robot interaction (pHRI) in the real domestic and professional world [23].

The fundamental breakthrough in this regard was the human-centered design of a robot

mechanism and control algorithm based on intrinsically elastic robot actuators such as series elastic actuators (SEA) or its generalization variable impedance actuators (VIA) [77, 47]. By considering the physical contact between the human and the robot at the design stage, the damage caused by unintentional contact was greatly reduced. Furthermore, taking into account the human's intention and preferences enabled the realization of human-friendly motions and interaction behavior. Some of the most advanced systems that were developed are now entering the industrial markets, e.g., LWR 4+ and iiwa 14 from KUKA [32] (see Fig. 1.4). These



Figure 1.4: KUKA LBR iiwa 14 Robot.

technologies serve both industrial and service-oriented domains. They range from industrial coworkers and mobile servants to robots in the professional service sector, assistive devices for the disabled, and support service robots for general household activities. All of these applications share the common requirement of close, safe, and dependable physical interaction between human and robot in a shared workspace. Therefore, such robots need to be carefully designed for human friendliness [23]. This is a challenging topic as it requires active and/or passive compliance and a lightweight design as the central components of the design [23]. The active compliance is usually achieved through implementing different force and/or impedance control schemes. Rather, the passive compliance is usually inherent to the structure of the robot.

In comparison to a compliant robot, a stiff robot is programmed to have predetermined

trajectories and follow the exact same trajectories no matter how it was disturbed by external force. Although this is a good thing for precise manufacturing procedures, it could be harmful, even fatal to people who work around the robots. Several attempts were tried to provide human-robot safety, safeguarding [87, 82, 27, 41], human-robot collision avoidance [18, 29, 20, 10], light weight robot [60, 75, 74, 28], sensor based compliant joint [67], and compliant joint with inherently compliant elements [46, 11, 78].

DASA was also developed under the same premises to support safe physical interactions with human. This actuation approach includes an active component (electric motor) and a semi-active component (MR clutches) which is coupled to the electric motor, hence the name “active, semi-active”. The electric motor is designed to provide input power to all MR clutches through distributed actuation, hence the name “distributed”. The motor locates at the base of the robot, and provides a constant output power to the input part of each MR clutch, and each MR clutch is able to control transmission of the torque to its output shaft independently [34]. By rotating the input shaft through electric motor, the fluid shearing happens with respect to the output shaft. This unique feature provides inherent compliance for mechanical transmission while enabling the actuator to produce high torque to mass ratio in comparison to same size actuators. The output torque can be precisely controlled with introducing simple PID controller, which makes MR clutches a good candidate for compliant actuation [55].

One of the challenges of using MR clutches is the nonlinear hysteresis relationship between the input current and output torque of the actuator. The hysteresis is mainly because of the use of ferromagnetic components inside the MR clutch including the coil, shaft and disks [34]. This nonlinear behaviour will result in undesirable tracking error. In order to compensate for the nonlinear behaviour without involving external sensors like torque sensors, several methods were proposed to linearize the MR clutch’s input-output relationship. A FPGA-based control scheme was reported in [34]. A new adaptive control scheme was proposed to compensate for the magnetic hysteresis in magnetorheological (MR) fluid-based actuators [84].

1.3 Objectives

The recent progress made by researchers around the world in the field of safe human-robot interaction has led to further expectation of developing technologies that support and facilitate implementation of such interactions. However, this is a challenging undertaking since it is difficult to achieve inherent compliance without sacrificing the accuracy, bandwidth, and complexity [76].

As mentioned earlier, DASA was developed for this purpose and the objective of the current research is to design and develop customized electronics and firmware that support adequate operation of DASA and facilitate application development using DASA actuation. One such application is the development of a 5-DOF antagonistic collaborative robot.

The overall design of this robot imposes several requirements on the design of electrical and electronic systems. These requirements are listed below. Each joint of the robot requires independent current levels for driving the MR clutches used in various joints. The maximum torque at each joint is different and a universal design of the current driver must support the demand of all joints. The current driver PCB should be compact enough that can be embedded into the smallest joint, without losing its heat dissipation properties. All joints and the components therein must be electrically isolated. The selection of microprocessor and relative decoupling capacitors, current-limiting resistances, etc., needs to be carefully done to meet the minimum system operating frequency and operating standard. All auxiliary components need to be included in each electronic driver in order to fulfill all other necessary functionalities such as communication, voltage conversion, and ADC/DAC interfaces.

The software development also needs to meet certain requirements. Real-time operating system is essential for processing the data from high numbers of sensors, communicating devices and executing control commands. It is essential for the sensors and communicating devices to not miss their deadlines. The design of real-time operating system plays an important role on the performance of the robot. The real-time operating system is also responsible for executing control commands. This requires accurate tuning and testing of control parameters. The architecture of the robot control system follows a master-slave structure. The real-time operating system supports necessary communications among different joints using this network

configuration. In this network configuration a master controller provides the desired commands (position, torque, velocity, etc.) for all robot joints as the slave controllers in the network.

Fig. 1.5 shows the overall structure of the robot and the electronics at joint level. Each joint of the robot includes a pair of antagonistic MR clutches and a set of proprioceptive sensors that measure values internal to the robot, including Hall sensors, absolute encoders, and current sensors. The figure also illustrates the communication among these components at joint level.

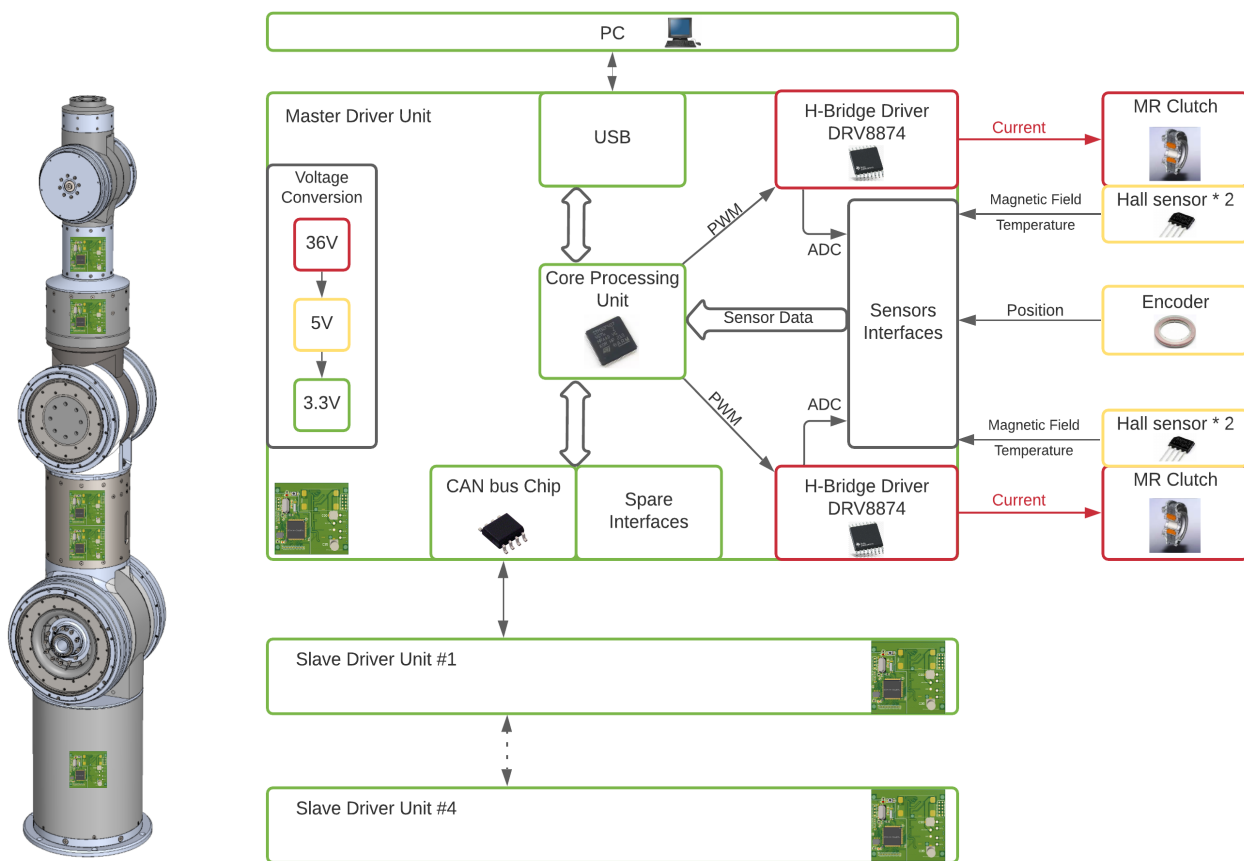


Figure 1.5: Schematic of the electronics of the 5-DOF collaborative robot.

The structure of the robot (shown in Fig. 1.6) has been previously reported in multiple publications and are not repeated in this dissertation for the sake of brevity [44, 43, 45].

To conclude, this dissertation mainly focuses on the development of new, integrated electronic drivers and the firmware that achieve joint-level control of individual clutch pairs as well as task-level control of 5-DOF collaborative robot with MR actuators.

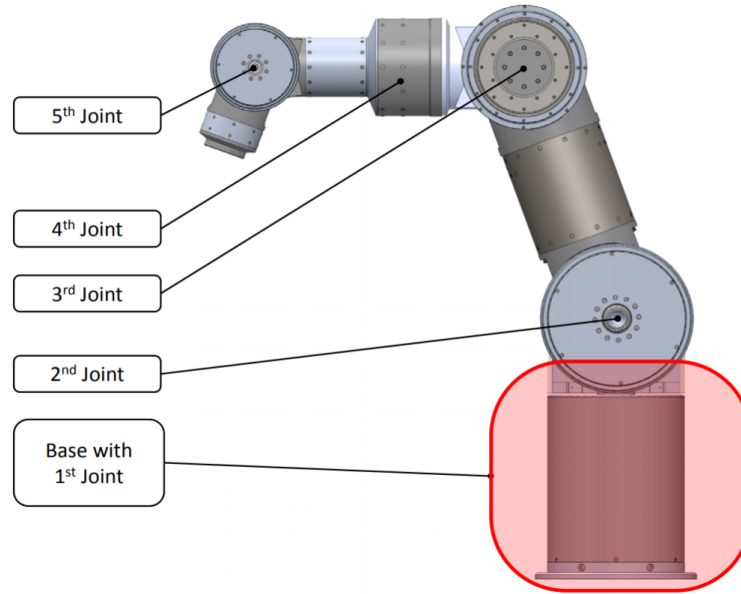


Figure 1.6: Visualisation of the 5-DOF robot.

1.3.1 Main Contributions

The main contributions of this work are listed below:

Design and Development of Customized Electronics

Customized electronics aiming at constructing a decentralized control network for a 5-DOF MR clutch based collaborative robot is designed, developed, and evaluated. The decentralized configuration of the control architecture will distribute the complexity of the controlling antagonistic MR actuators among the joints. The miniaturized design of the electronics allows them to be embedded inside the housing of each joint. Meanwhile, each electronic driver processes multiple sensors' data, drive the MR clutches, execute control commands and communicate with other joints. These functionalities cannot be achieved using commercially available control boards and driver boards without sacrificing performance and/or size. We used STM32F4 family processor as the core processing unit of our electronics. Each driver unit includes an STM32F4 processor to receive feedback signals from all sensors (Hall Sensor, Encoder, Current sensor) to control the antagonistic clutch pair in a closed-loop feedback. A series of experiments will be performed to evaluate the performance of the developed electronics and the

results will show the effectiveness of the electronic drivers and the robot itself for performing advanced control schemes.

Development of a Real-time Firmware

A real-time control scheme is constructed using $\mu\text{C}/\text{OS-II}$, a commercially available real-time operating system with preemptive kernel. It is essential for multiple joints of the robot to have predictable and reliable computing power environment due to the large amount of data exchange and the need to implement dependable control algorithm for interactive tasks with the environment including humans. The main advantage of the real-time program is that it handle miscellaneous devices and tasks with necessary priority while enhancing the overall performance of the control algorithm for human-robot interactions. The real-time program is assessed in various scenarios to evaluate the performance of the system.

Performance Evaluation the of Individual Robot Joints and Initial Control of the Robot

The performance of the electronic drivers and MR actuators in joint are thoroughly assessed. Through rigorous experimental studies, the controllability of the combination of the electronic driver and MR actuators in following aggressive control commands are tested. Both position and torque control in each joint of the robot are assessed. The results from these experiments will show the reliability of our design and its suitability for implementing more advanced control schemes. In addition to the joint-level control, the performance of the overall decentralized control system for task-level control of the robot is also assessed.

1.4 Thesis Outlines

The rest of this thesis is organized as follows.

In Chapter 2, the structure and the working principle of the MR clutch is first presented. Then the design and development of the customized electronic driver are discussed. These include the analysis of the design requirements, the selection of electronic components, and the optimization based on voltage/current transients and noise reduction. The necessary auxiliary functionalities are illustrated. A hardware schematic is included to further clarify the structure

of the customized electronic driver. The thermal performance of the board is evaluated based on calculation. Finally, the sensors that are integrated into the robot are listed. A series of experiments will be conducted to evaluate the stability of the voltage conversion system on-board, and the thermal durability of the driver unit.

In Chapter 3, the software development of a real-time firmware for processing sensors' data and controlling the robot joint is presented. The system architecture, real-time operating system working principle and relative system configuration are detailed. All of the real-time user tasks that are involved in real-time system are analyzed and detailed. The between-joint communication protocol is presented. Then a series of experiments are conducted to evaluate the performance of the designed software and hardware systems.

In Chapter 4, the thesis is concluded and new directions for future works are included.

Chapter 2

Design and Development of Embedded Electronic Drivers for a 5-DOF MR Clutch Based Collaborative Robot

In this chapter, the design and development of customized electronic drivers for constructing a decentralized control network for an MR clutch based 5-DOF collaborative robot is presented. The actuation mechanism of this 5-DOF robot is based on Antagonistic Distributed Active Semi-Active Actuation (A-DASA). We will first briefly describe the structure and working principle of the MR clutch. The design requirements of the electronic driver are then analyzed based on required functionalities and operational performance. Furthermore, transient suppression and noise reduction are analyzed and optimized based on theoretical calculations to help further improve the design. The thermal performance of the driver is also evaluated to assess the heat dissipation potential. In the end, the sensors are briefly introduced, and a series of experiments are conducted to evaluate the voltage conversion system.

2.1 Introduction

With the increasing amount of industrial robots being installed and operating in factories of all kinds, reports about injuries and even death related to industrial robots keep rising. The need for human-machine safety related products is urgent. However, the current methods for preventing collisions including proximity sensing, camera and the other commercially available solutions covered by a survey [81], seem only can treat the symptoms rather than the causes. Accidents and failures could be caused by engineering errors, human mistakes and poor environmental conditions (see Fig. 2.1 [80]). Engineering errors include errors in the robot's

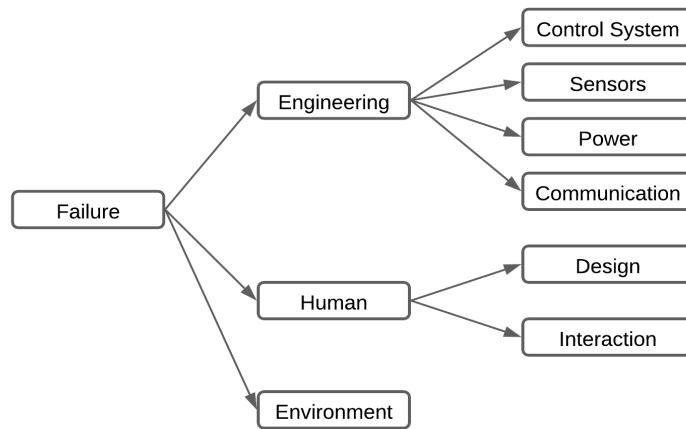


Figure 2.1: Taxonomy of failures

mechanics (loose connections across parts, faulty electronics), errors made by the controller (programming bugs, faulty algorithms), etc [80]. Therefore, the demand for developing an intrinsically safe actuators without sacrificing performance runs high.

Compliant actuators are a kind of intrinsically safe actuators that allow deviations from their equilibrium points, which is dependent on the external force applied to them. Compliance is achieved by disengaging the reflected inertia of the actuator from the output joint, by introducing a damper [15] or elastic component [47, 48] into the transmission mechanism. Examples of compliant actuator designs include MACCEPA (Mechanically Adjustable Compliance and Controllable Equilibrium Position Actuator) [79], and Electro-Thermal-Compliant (ETC) actuator [88]. It has been shown that the compliant actuation indeed reduces impact forces [40].

As previously explained, MR actuators are compliant in nature since there is no rigid coupling between the load and the actuation source. It has been shown that MR actuators replace the reflected inertia of the motor and gearbox with the inertia of the MR clutch itself [58]. This makes MR actuators an excellent actuation choice for safe human-robot collaborations. The higher torque-to-mass and torque-to-inertia ratios of MR actuators compared with similar size commercial motor-and-gearbox actuators result in overall lighter mechanisms with improved human-safe proprieties.

Over the past decade, our research group has rigorously studied the working principle of MR actuation and has developed several generations of the MR actuators. The studies have been reported in several publications [58, 59, 86, 35].

In the following, we briefly review the working principle of the MR actuation in a multiple-axis actuation mechanism such as a robot manipulator. Considering a robot manipulator with

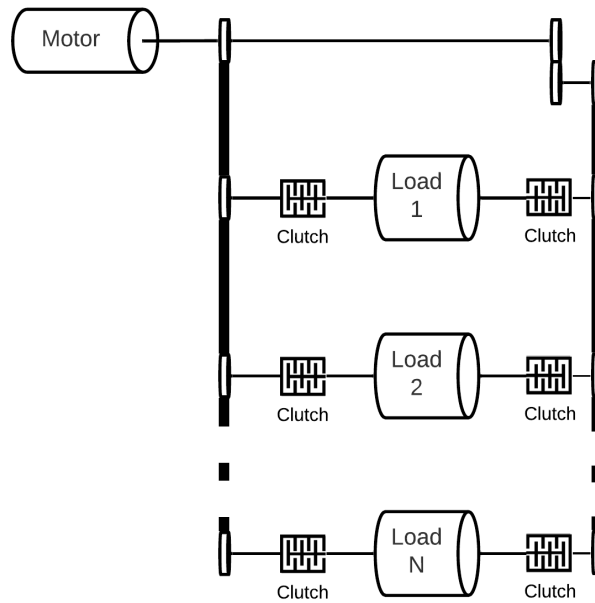


Figure 2.2: DASA actuation concept.

multiple joints, an electric motor located at the base of the robot can provide the required power to all robot joints through the distributed MR clutches as shown in Fig. 2.2 (modified from [45]). As seen, a pair of MR clutch arranged in an antagonistic configuration is attached to each load. The motor and the MR clutches are coupled using shafts, timing belts, or a

combination of them. The combination of active (motor) and semi-active components (MR clutches) allows for the controlled transmission of the torque to each individual load using MR clutches while decouple the reflected inertia of the actuator and the joint.

In the remaining of this chapter, we will fully analyze the control requirements of an MR clutch as well as the design requirements of the electronic driver. The organization of this chapter is as follows: Section 2.2 introduces the Magneto-rheological fluids and the working principle of the MR clutch. The working principle of the MR clutch is outlined. Section 2.3 lists the design requirements of the system, and selects the electric and electronic components according to these requirements. The circuit protection of transient suppression and noise reduction are detailed and experimentally verified. The system auxiliary functionalities is also included in this section. The hardware schematic is presented next followed by the analyses and the tests of the thermal performance of the control board. The last section briefs the propriospective sensors used in the robot and MR clutches.

2.2 Magneto-Rheological Clutch

MR fluids is a controllable fluids with variable rheological behaviour in response to an external magnetic field. Such fluid is composed of micron-sized magnetic particles that locate inside a liquid carrier [8]. The viscosity of the fluid can be changed by applying an external magnetic field, therefore changing the shear stress of the fluid. According to the manufacturer datasheet for MR fluids (see Fig. 1.3) and experimental results [35], the shear stress of MR fluids has a one-to-one relationship with the applied magnetic field [59]. This benefit prompts the invention of a clutch-like mechanism called MR clutch, which can be used for controlled torque transmission. When the external torque (on the output side of the MR clutch) exceed a predefined value, the clutch start to slip and a higher output torque cannot be transmitted [9]. This type of actuators truly achieve intrinsically compliant actuation. The structure of the MR clutch is shown in Fig. 2.3. It consists of four parts, input disks/shaft, output disks/shaft, electromagnetic coil, MR fluids and auxiliary components like aluminum spacers. The spacers are used to avoid disks from touching each others and to align the input/output disks. The input power is provided by an electric motor (or any other rotary source) to the input shaft/disks.

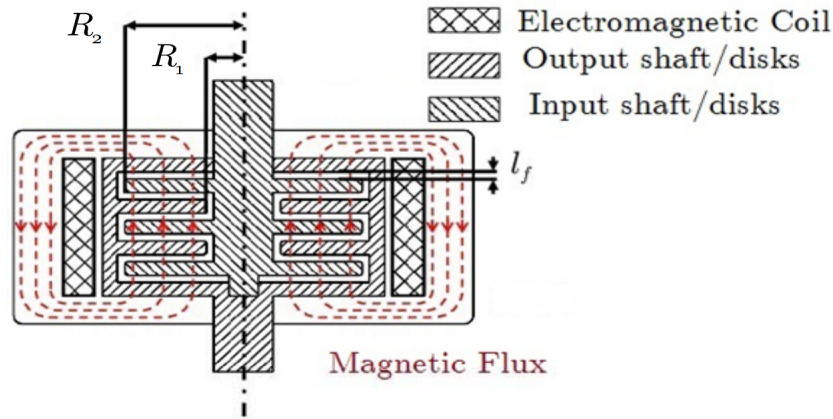


Figure 2.3: Cross-section of a multidisk MR clutch.

By controlling the current passing through the electromagnetic coil, the shear stress of the MR fluids within the magnetic path is controlled (increased), hence the input power is transmitted to the output disks/shaft. The motor speed, normally, is kept constant during operation.

2.3 Electronic Design

The electronic design typically contains the selection of electric and electronic components in order to meet specific design requirements. To select appropriate components, the design requirements for each component needs to be specified. Table 2.1 lists these requirements for the intended electronic drivers. The power supply system plays an important role in the overall system. It must provide power for all components, sensors, as well as the clutch itself at the appropriate level. To provide a stable voltage is to to each of these components, the power supply needs to have a stable and low noise voltage with small output ripples, good load-transient response, strong transient suppression capability and the ability to prevent prevent Electromagnetic Interference (EMI).

Apart from the power supply system design, the appropriate selection of the core processing unit is perhaps the second most important design aspect. This is a multifaceted and iterative process, in which the processing speed, debugging capability, availability of the required pins and other functionalities need to be accessed and validated. It is not the most straightforward

Table 2.1: Design requirements of electronic driver

Design requirements	Comments
Voltage conversion	36V - 5V - 3.3V
Multiple communication modes	CAN, RS485, USART
Serial Wire Debug	Debug interface
Hall sensors communication	SENT protocol
Encoder communication	Asynchronous Serial Data protocol
Maximum coils current	3.3A MAX
Separated signal and power ground	Avoid disturbances
Heat dissipation capacity	Enlarged copper area
Additional pins	USART, Timer, ADC, DAC
Size	To fit within robot joint housing
Other requirement	Reset button

path to determine whether these requirements can be met without either fully developed or emulated firmware.

The third important component is the selection of the current drivers. The current drivers require to deliver continuous current to each clutch without overheating. Given the intended application of the current driver for robot joint actuation, it is necessary for the current driver to be able to deliver cautious current for maintaining a particular joint configuration. In addition, the current drivers are required to be embedded within the housing of each joint which cannot be ventilated for thermal dissipation due to space limitations. Additional features such as current sensing, current regulation, fault protection and chip packaging are also taken into account.

The fourth and last consideration is the communication system, regarding implementability as well as diversity. Given our design requirement, at least 2 types of communication methods should be included.

In addition to component selection, the layout of the circuit board also determines the system performance. Two important factors that affect the system performance are noise reduction

and heat dissipation of circuit layout. EMI is one of the main reasons for the induced noise. Regarding the electronic driver discussed in this work, EMI is mainly generated by the onboard drivers and power conversion system, rather than the MR clutches. The MR clutches are not a source of EMI since these devices are designed such that their internal magnetic field remains inside the MR clutch and will not leak outside the MR clutch. This allows the driver units to be placed inside the robot housing. Regarding the EMI generated on the board, the solutions are to physically keep the power components (Current driver) as far as possible from or on the opposite layer of signal components (MCU, Crystal, Communication ICs, etc.). This can reduce the common-mode radiation. The second solution is to separate the power ground from the signal ground, in order to decouple the signal components from power components. The third one is to avoid long paths while placing tracks. The last solution is to select appropriate Surface-Mount Device (SMD) components. Note that even though the driver units will be placed in the robot housing, they will not be interfered by the magnetic field inside the MR clutches.

Another important issue is heat dissipation. The normal electronic drivers can use fans or large copper area to dissipate the heat. However, in our case, there is a strict restriction on the size of the circuit board. The location of the circuit board inside the housing of joint two is shown in Fig. 2.4. The diameter of the joint two housing is 152mm. To avoid the board from touching the transmission components inside the housing, the board must be restricted within 70mm by 70mm. Due to the space limitation, the board also cannot be ventilated for thermal dissipation. These restrictions make it harder to satisfy the design requirements without sacrificing performance.

Here we discuss some general approaches regarding this issue. Detailed calculation of the thermal performance of the driver will be discussed later in this chapter. In general, to achieve better heat performance, it is common to enlarge the copper area as much as possible to increase heat dissipation surface. Due to size limitation, the design uses two-layer structure to enlarge the copper area. Also, it is common to place more via points underneath the power components and nearby these components to transfer more heat to the opposite layer. In designing the circuit layout one also needs to ensure that the copper is not interrupted by tracks that run perpendicular to the thermal path away from the power component [70]. Lastly, connecting a

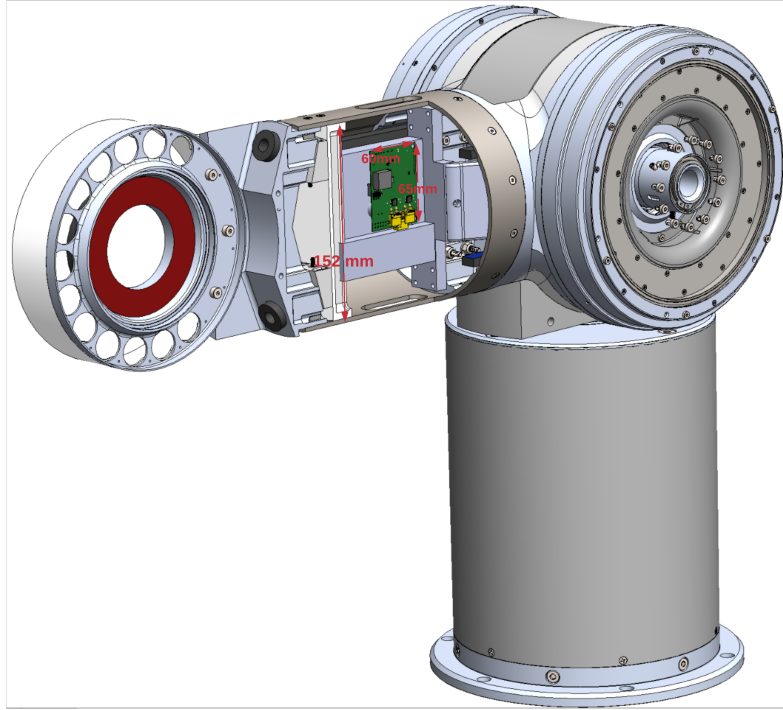


Figure 2.4: The location of the electronic driver inside the housing of Joint 2.

heat sink to the case of power component will significantly enhance the thermal performance of the board.

2.3.1 Component Selection

The requirements of the design are clear and definite. This section will give a brief description of the selected key components on the board (see Table 2.2).

The STM32F4 microprocessor features 168MHz processing frequency, 512 Kbytes of flash memory and sufficient number of AD/DA converters, timers, DMAs (Direct Memory Access), dozens of I/O ports as well as multiple communication interfaces, such as I²C, USART/UART, CAN [66].

The H-Bridge drivers selected for energizing the electromagnetic coils is DRV8874. The DRV8874 driver features a wide supply voltage range (4.5-37V) that meets the requirement of our design (i.e. maximum 36V). In addition, it satisfies the requirement of a continuous

Table 2.2: Key components table

Components	Comments
STM32F407VET6	Microprocessor
DRV8874	H-Bridge Motor Driver
SN65HVD232DR	CAN Bus Transceiver
MAX485	RS-485 Transceiver
MP4560	DC-DC Step-down Converter
AMS1117	Voltage Regulator

output current of 3.3A. The driver integrated current sensing and current regulation functions allows to eliminate the need for additional current sensors, which will significantly help with saving space on the circuit. Additionally, the driver includes fault protection features, including overcurrent protection and thermal shutdown. These features will simplify the design of firmware.

As mentioned previously, the design of the voltage conversion system is of great concern. The chip MP4560 is chosen to perform the first stage of voltage conversion from 36V to 5V. The 5V is used to supply all external sensors as well as some of the on-board components. The design of the MP4560 chip can achieve output voltage ripple of lower than 5mV when it is half loaded, and is able to prevent EMI [39]. Another voltage regulator IC, AMS1117 is used for the second stage of voltage conversion from 5V to 3.3V [3]. This regulator is used to power up the microprocessor and other on-board components. The regulator features ripple rejection. By introducing a two-stage voltage conversion system, not only the on-board components can be powered properly, but also the influence of EMI is reduced significantly.

2.3.2 Circuit Design

Every IC requires additional electric components to construct a complete circuit. Those components can filter undesired noises as well as protect the circuit. In addition, the design of a circuit that involves large inductance such as those used in the out system is by no means a

trivial task.

Transient-voltage-suppression Diode

In addition to the decoupling capacitors, the use of Transient-Voltage-Suppression (TVS) diode is necessary to protect the circuit from inductive switching transients, when the load is a coil or motor, as it is the case in this work. To be specific, while performing the control actions,

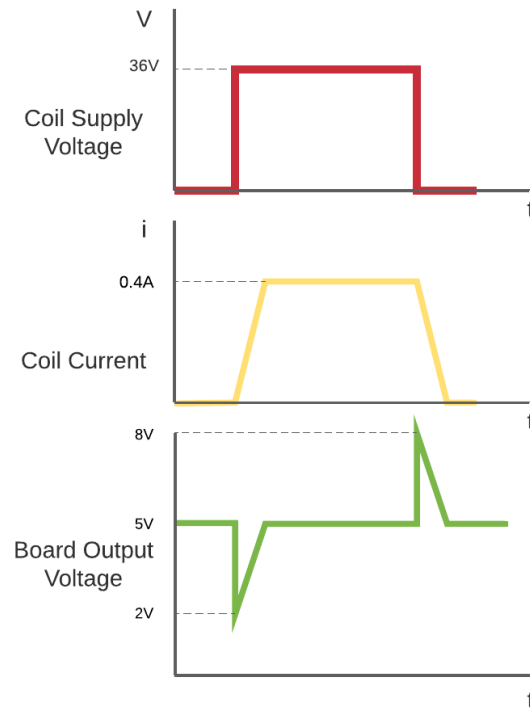


Figure 2.5: Schematic of voltage change.

the frequent changes of the current in the electromagnetic coil will generate heavy back electromotive force (EMF) because of the large inductance of the coil. The back EMF will cause undesirable behaviour such as voltage drop (jump) because of the current increase (decrease) (see Fig. 2.5). If voltage jump is large enough, it may cause permanent damage to other components. In contrast, a large voltage drop could negatively affect the performance of the system. To find out the maximum value of the voltage change in our system, we used a coil

with a shape and size similar to the coil used in the 2nd joint of the robot (i.e. the largest coil) in order to approximate the inductance and the induced back EMF of the coil. Fig. 2.6 shows the cross section of the coil used in our study.

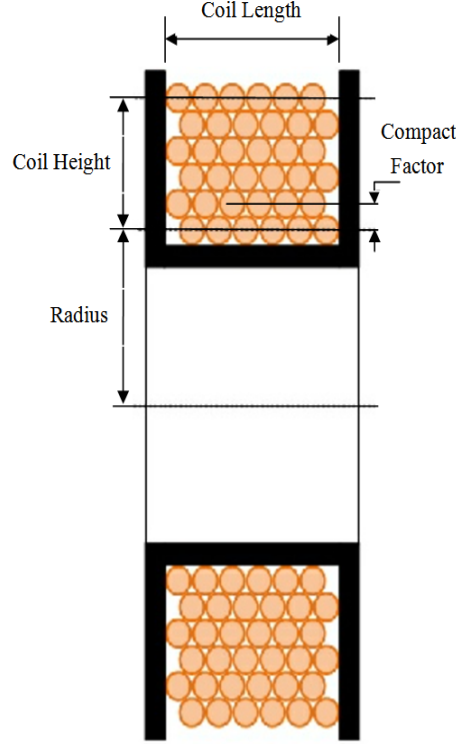


Figure 2.6: Cross-section of the coil.

The inductance of the coil can be estimated by,

$$L \approx N^2 \mu_o \mu_r \cdot \left(\frac{D}{2}\right) \cdot \left(\ln\left(\frac{8D}{d}\right) - 2\right). \quad (2.1)$$

where N is the number of turns, μ_o is the permeability of the free space ($4\pi \times 10^{-7}$), μ_r is the relative permeability of the core (Carbon steel 1018 with permeability of 529 [1]), D is the coil inner radius in meter, and d is the diameter of the AWG 20 copper wire with insulation in meter which includes the compacting factor of 0.866.

The maximum voltage change of the coil can be obtained easily as,

$$\Delta V_L(t) = -L \cdot \frac{di}{dt}, \quad (2.2)$$

where L is the inductance given in (2.1) and di/dt is the rate of current change. The parameters of the coil of the 2nd joint are listed in Table 2.3.

Table 2.3: Information of the second joint coil

Position	Dimension
Coil Inner Radius	58.5mm
Coil Height	11.5mm
Coil Length	26.3mm
Compact Factor	0.866
Number of Turns	445
Copper Wire Diameter	0.879mm
Core Material	Carbon Steel 1018

While estimating the inductance of the MR clutch, we cannot simplify the MR clutch as a coil, since the ferromagnetic material in the immediate area of the coil will have significant effect on increasing the apparent inductance of the coil. The ferromagnetic material includes the shaft and the disks of the MR clutch. For simplicity, we use an LCR meter (LCR400) to measure the inductance of the MR clutch. The result is around 40 millihenry. As expected this is a large inductance that clearly highlights the challenges addressed in this research with respect to the design of appropriate electronics. Substituting this value in (2.2), and assuming a sinusoidal input current of 1Hz with an amplitude of 3A, the voltage change in the coil can range from -1V to 1V, for a sampling interval of 100 milliseconds.

The current change in the coil does not always act as a sinusoidal waveform. As one example, when the power supply is turned off, the on-board power supplies experience a large voltage spike because of the rapidly collapsing current within the coil's. This will induce a transient voltage that can be larger by an order of magnitude than the one calculated above (see Fig. 2.5). It is also clear that the magnitude of voltage swing is proportional to the frequency of the input signal as well [72].

To address the issue of the induced voltage, we used TVS diodes in our design. The TVS clamping are used to divert the transient voltage to the ground, therefore limit the voltage

across the circuit to a safe level. We used a bidirectional Zener diode to implement TVS, which has a symmetrical and equal breakdown voltage in both positive and negative directions. The characteristic of the Zener diodes used in our design is shown in Fig. 2.7 (adopted from [49]). The selection of suitable TVS diodes entails paying attention to several specifications including the breakdown voltage and clamp voltage. The breakdown voltage, represented in Fig. 2.7 (adopted from [49]) as V_{BR} , indicates the voltage level in which the diode begins to

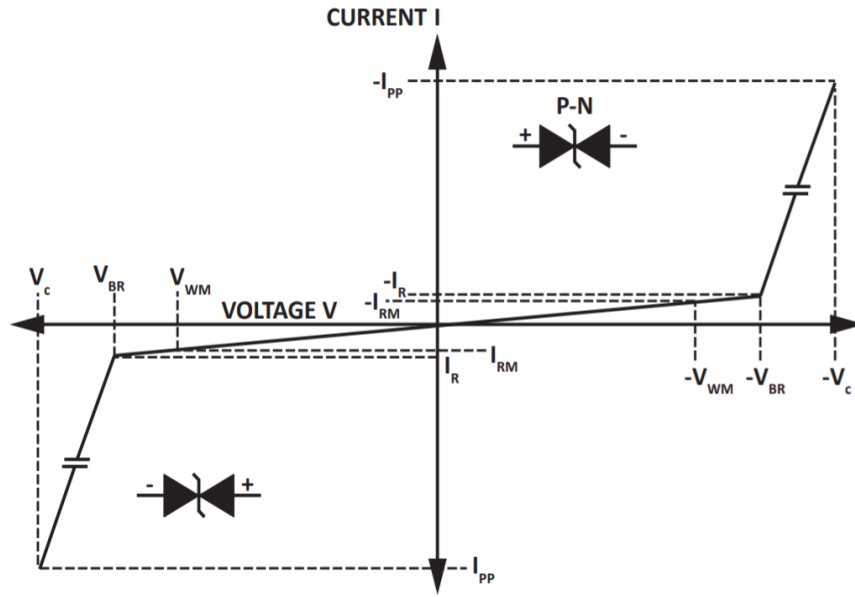


Figure 2.7: Transient voltage suppression diode I/V curve.

conduct the current. The diode does not fully conduct current until the voltage reaches V_C , i.e., the clamping voltage. In other word, V_C determines the voltage that the system will be exposed to during a transient period, so the lower the V_C , the less chance that the system will be endangered because of the voltage swing [72]. If the V_C is less than the absolute maximum voltage of the system, the TVS system will fail to protect the circuit from the transient voltages [72].

Protection Capacitors

As discussed in the above section, the voltage transients can be clamped by bidirectional TVS diodes. Even so, TVS diodes are still in danger, when facing extreme situations like large

transient voltage. In this case, additional input capacitors including local bypass/decoupling capacitors and bulk capacitors should be considered in the design of the voltage conversion system. These capacitors are the second barrier to protect the circuit, since input capacitors can be seen as an energy buffer to deal with unexpected situations. Thus, the capacitance cannot be randomly selected. Regarding the decoupling capacitors, their capacitance can be calculated as follows,

$$C_{DeMIN} = \frac{I_{OUT} \cdot dc \cdot (1 - dc) \cdot 1000}{f_{SW} \cdot V_P}, \quad (2.3)$$

where I_{OUT} is the steady state output current, f_{SW} is the switching frequency (e.g., 500 kHz), V_P is the maximum permissible peak to peak ripple voltage (e.g., 75mV in most cases), C_{DeMIN} is the minimum capacitance of decoupling capacitors in μF , and dc is the duty cycle defined as $dc = V_{OUT}/(V_{IN} \cdot \eta)$, in that η is the convert efficiency. Using (2.3), the total required input decoupling capacitance can be chosen within the range of 0.38 μF to 8.4 μF . These bypass/decoupling capacitors are often chosen to have a small capacitance of 1 μF and are placed in parallel to each component to decouple the chip from high and low frequency noises. The withstanding voltage of the bypass capacitors should be several times larger than the supply voltage of its corresponding ICs.

As for the bulk capacitor, it protects the circuit from voltage spike caused by a sudden release of the inductive energy while also compensates for the voltage drop at the input when responding to an output load transient. The required capacitance of the bulk capacitors can be calculated as,

$$C_{BulkInput} = \frac{I_{Load} \cdot V_{OUT} \cdot (V_{IN} - V_{OUT})}{\Delta V_{IN} \cdot f_{SW} \cdot V_{IN}^2}, \quad (2.4)$$

where I_{Load} is the load current (e.g., 2 A), V_{OUT} is the output voltage (e.g., 5 V), V_{IN} is the input voltage (e.g., 36 V), ΔV_{IN} is the permissible input voltage ripple (e.g., 0.1V), and f_{SW} is the MOSFET switching frequency (e.g., 500kHz). Accordingly, the required capacitance of the input bulk capacitor was obtained as 4.7 $\mu F/100V$. The actual selection of the input bulk capacitors should be a few times larger than its estimated value, to provide an appropriate safe margin for protecting the circuit. At the same time, the physical size of the capacitor becomes a limiting factor in case where space scarcity is a restricting factor as is the case here.

The capacitance of the output bulk capacitor can also be calculated as follows,

$$\Delta V_{OUT} = \frac{V_{OUT} \cdot (V_{IN} - V_{OUT})}{f_{SW} \cdot L \cdot V_{IN}} \cdot \left(\frac{1}{8 \cdot f_{SW} \cdot C_{BulkOutput}} + R_{ESR} \right), \quad (2.5)$$

where ΔV_{OUT} is the permissible output voltage ripple (e.g., 0.1 V), equivalent serial resistance R_{ESR} (e.g., 0.1 Ω), L is the inductance defined as,

$$L = \frac{(V_{IN} - V_{OUT}) \cdot V_{OUT}}{0.3 \cdot f_{SW} \cdot I_{OUT} \cdot V_{IN}}, \quad (2.6)$$

and all other parameters are as defined previously.

The inductance L in our system was estimated to be around 14 μ H. Substituting this value into (2.5), the required capacitance of the output bulk capacitor is obtained as 16 μ F. In addition to the bulk capacitor of the output side, several other small capacitors of less than 1 μ F can also be connected in the circuit to reduce output noise.

The required capacitance and inductance of the voltage conversion system calculated above satisfy the safety and noise reduction requirements of the design and render the circuit to operate without any safety concern.

Other components such as microprocessor, communication chips also require decoupling capacitors to be decoupled from other components. The suitable selection and placement of the decoupling capacitors are the key to decouple/bypass these components from the noise induced by other components. These capacitors also bypass other components from the induced noise of the frequent switching ICs. Because the logic circuits such as microprocessors tend to do frequent switching, the system is influenced by large EMI. The decoupling capacitors with small capacitance can effectively decouple these ICs if a suitable capacitance value is selected (see Fig.2.8). The impedance of the ceramic capacitors decreases monotonically as frequency increases until it reaches the self-resonant frequency [5]. The self-resonant frequency of the capacitors should be close to the value of the switching frequency of the ICs to achieve the most suitable decoupling effect. In this design, the selection of the capacitors was done with respect to STM32F4 microprocessor and the corresponding guidelines for this processor in [64].

The capacitor selection for the current driver is different from those described above. The protecting capacitors of current driver are also separated to bypass, and decoupling capacitors as well as local bulk capacitors. The bypass and/or decoupling capacitors reduce the ripple

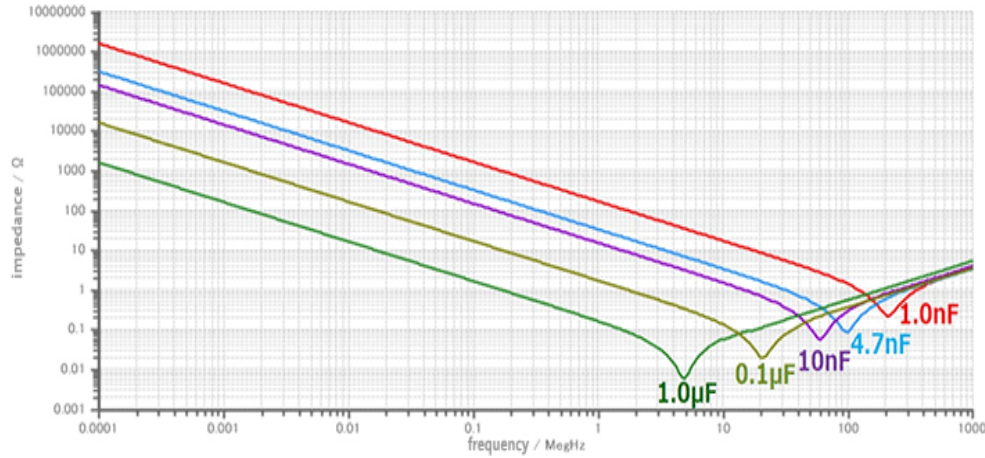


Figure 2.8: Frequency characteristic of ceramic capacitors.

voltage amplitude and decouple the noise that are generated by other components. A wrong value of the bypass and/or decoupling capacitors results in large ripple voltage. [69]. Therefore, the ripple voltage must be reduced to a safe level to not damage the local bulk capacitors. For this reason ceramic capacitors are used because of their low equivalent series resistance (ESR) as shown in Fig. 2.9. The ripple voltage is proportional to the ESR value.

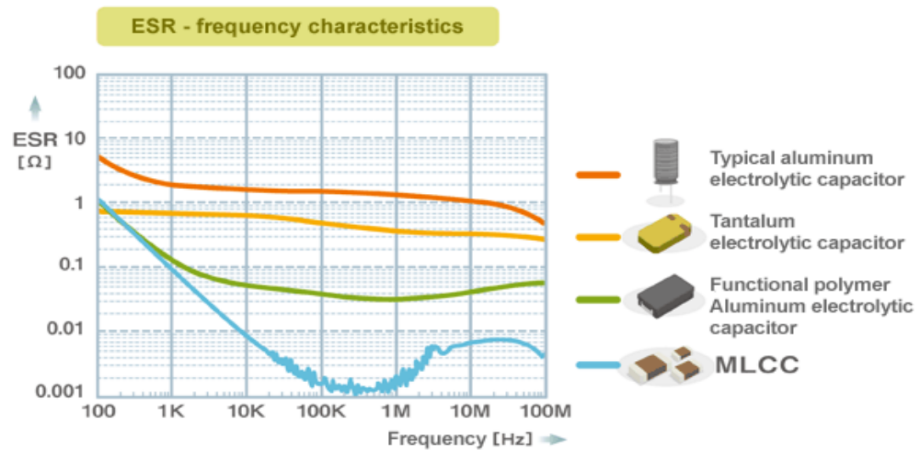


Figure 2.9: ESR frequency characteristic of capacitors.

The amplitude of the ripple voltage is reduced by the bypass and/or decoupling capacitors. A low ripple voltage ensures the local bulk capacitor to limit the rate of voltage change across the load. There are cases in which the power supply is expected to response to fast output load

transients. Such output load transients result in large voltage deviation [69]. In such cases, the bulk capacitors act as an energy storage component. A sufficiently large bulk capacitance can keep the system stable and prevent the system from experiencing voltage deviations and/or spikes. The difference between the bulk capacitor of the current driver and those used for other components is that the withstanding voltage of the current driver should be high enough to bear the voltage transients (e.g., 100 V).

System Reset

In addition to the component selection, there are other considerations that need to be made in the design of the electronic drivers. The reset circuit and boot mode selection are two such essential considerations. The reset circuit is necessary when the microprocessor runs into internal deadlock. Such deadlock are not uncommon consider the real-time nature of the required firmware for the electronic driver. It is impractical to reset the system simply by unplugging the power supply for a number of reasons including voltage swings described above. When resting the system is necessary, the reset circuit will reset all main registers and setting the appropriate voltage level on all output ports. This can be achieved by pulling down the voltage level on NRST pin of the processor. When the voltage level is pulled down, the system start the rebooting process. There are three reboot spaces, namely main flash memory, system memory and embedded SRAM to instruct the reboot process. The desired space needs to be selected by the hardware [64]. Rebooting from the main flash memory is the default setting. To this effect, BOOT0, BOOT1 pins are pulled low .

Debug Interface

Another equally important aspect of the circuit design is to build up the connection between the PC used to develop the firmware and to save the acquired data, and the electronic driver. Developing such interface requires two components, namely the hardware debug tool and Serial Wire (SW) connector as shown in Fig. 2.10. (adopted from [64]) According to the hardware design datasheet [64], the selected microprocessor has two kinds of debug ports, Joint Test Action Group (JTAG) and SW. Considering the constrain of board size, the 2-pin interface SW is chosen as the debug port. The 2-pin connection consists of SW data, and SW clock. Both of



Figure 2.10: Host-to-board connection.

these two pins need to be pulled up to keep the system stable when the SW debug equipment is not connected.

2.3.3 Hardware Verification

In previous sections we analyzed various aspects of the circuit protection and discussed the guidelines for selecting appropriate components. In this section, the results of several experiments used to evaluate the on-board voltage conversion system and the adherence of the design circuit to the original design requirements are provided.

We first evaluated the ripple voltage of the 36 V Power Supply Unit (PSU) and the voltage conversion system. A Tektronix TBS 2000 oscilloscope was used to measure the ripple voltage. During the first test, the probe was set to AC coupling to decouple the DC component. Therefore, the oscilloscope can measure the minimum and maximum voltage of the alternating component (ripple voltage) of the output. The results are shown in Fig. 2.11. The 36 V Power Supply Unit (PSU) has a ripple voltage of 21.75 mV as shown in Fig. 2.11 (a). The 5 V regulator has a ripple voltage of 13.5 mV (Fig. 2.11 (b)). The 3.3 V regulator has a ripple voltage of 9.25 mV (Fig. 2.11 (c)). The results clearly show that the voltage conversion system significantly reduces the ripple voltage of the PSU. The voltage conversion system can supply a clean and stable voltage signal to the on-board components.

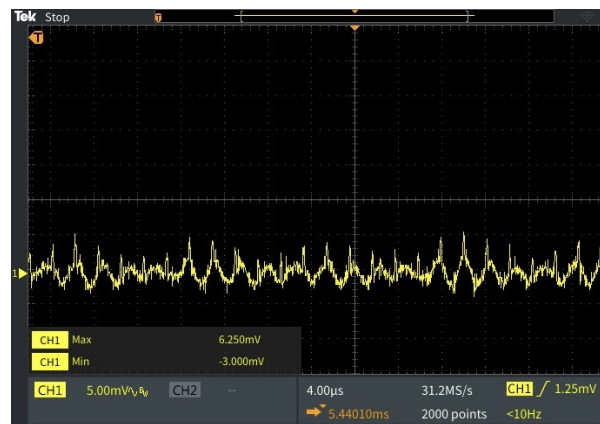
In addition to the ripple voltage, the startup of the voltage conversion system was also evaluated. The results are shown in Fig. 2.12. Fig. 2.12 (a) shows the startup voltage of the 36 V PSU. As seen, it is clear that the PSU features soft-start. This feature forces the output voltage to rise at a slower rate. As such, the output voltage rise time is about 21.9 ms. This feature of the 36 V PSU prohibit us from assessing the anti-shock ability of the other two voltage regulators. In order to asses the response of the on-board voltage conversion system to



(a)



(b)

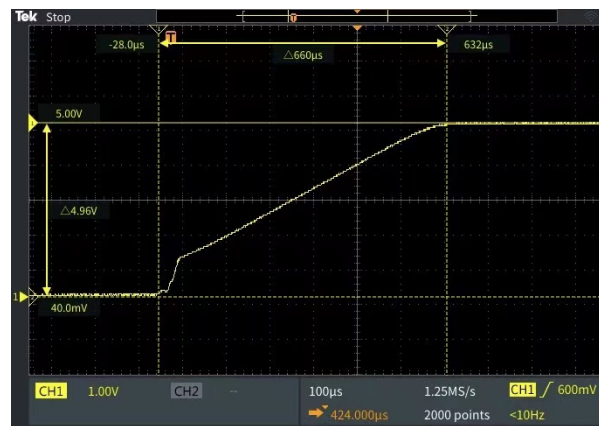


(c)

Figure 2.11: Ripple voltage of voltage conversion system; (a) 36V PSU ripple voltage, (b) 5V regulator's ripple voltage, (c) 3.3V regulator's ripple voltage.



(a)

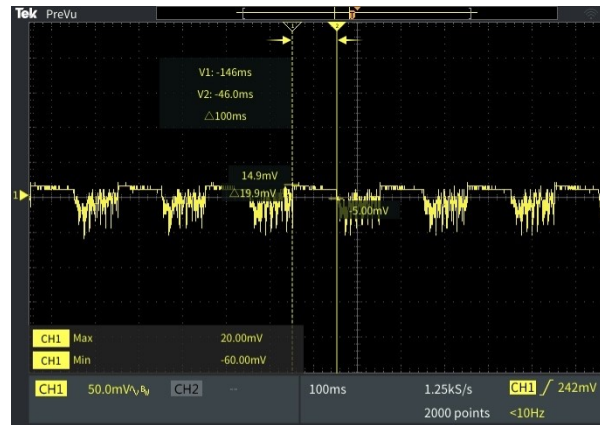


(b)

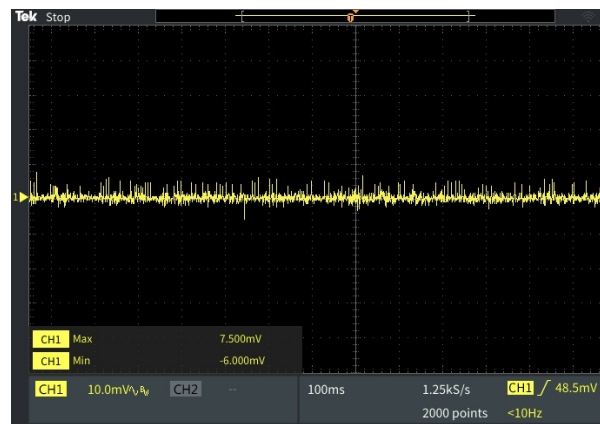


(c)

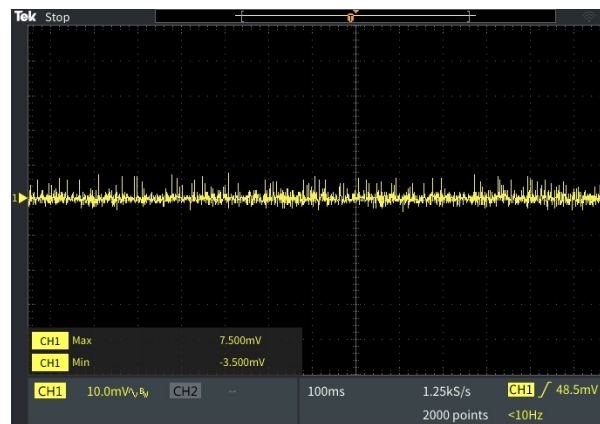
Figure 2.12: Startup procedure of voltage conversion system; (a) 36V PSU, (b) 5V regulator, (c) 3.3V regulator.



(a)



(b)



(c)

Figure 2.13: The response of voltage conversion system when supply the coil with a 5 Hz, 1 A square wave; (a) 36V PSU, (b) 5V output, (c) 3.3V output.

sudden voltages, the 36 V of the PSU was directly fed to the voltage conversion system after the rising period. The start up results for both the 5 V and 3.3 V regulators are shown in Fig. 2.12 (b) and (c). As seen, the rising periods of each regulator are $660\ \mu\text{s}$ and $515\ \mu\text{s}$, respectively. The results show that the response of each regulator is sufficiently quick and stable while also providing necessary anti-shock protection to the circuit.

Finally, the Anti-load disturbance ability and the transient suppression ability of the designed driver were evaluated. To this effect, the the MR clutch was used as the output load. To see the effect of load and evaluate the performance of the driver, the MR clutch was fed with a square wave current with 1 A magnitude and 5 Hz frequency. The results of the experiment are shown in Fig. 2.13. Fig. 2.13 (a), shows the effect of the output load on the 36 V PSU. It is clear that the voltage of the PSU is significantly disturbed as a result of the output load. However, looking at Fig. 2.13 (b) and (c), one can see that the voltage conversion system can filter these disturbances and supply a clean voltage to the rest of the on-board components.

The voltage conversion system designed above satisfy the safety requirements of the design and render the circuit to operate without any safety concern.

2.3.4 Schematic and Thermal Performance

The overall schematic diagram of the designed electronic driver is shown in Fig. 2.14, the figure depicts different voltage levels with different colour, i.e. red for 36V, yellow for 5V, green for 3.3V. Each electronic driver includes the processing unit as well as all necessary input and output modules to drive a pair of MR clutches through H-bridge driver, receive feedback signals from Hall sensors and encoders, and communicate with other drivers within the network. Fig. 2.15 shows three dimension views of the electronic driver.

As previously stated, two important requirements for the electronic drivers are the heat dissipation and the compact size of the board itself to fit within the housing of the robot joint.

To address the heat dissipation and to handle extreme overheating of the components, the designed driver includes custom-designed heat sink plates. The design of the heat sink plates was based on the theoretical thermal properties of the current drivers which are described next. We verified the heat dissipation performance of the heat sinks experimentally.

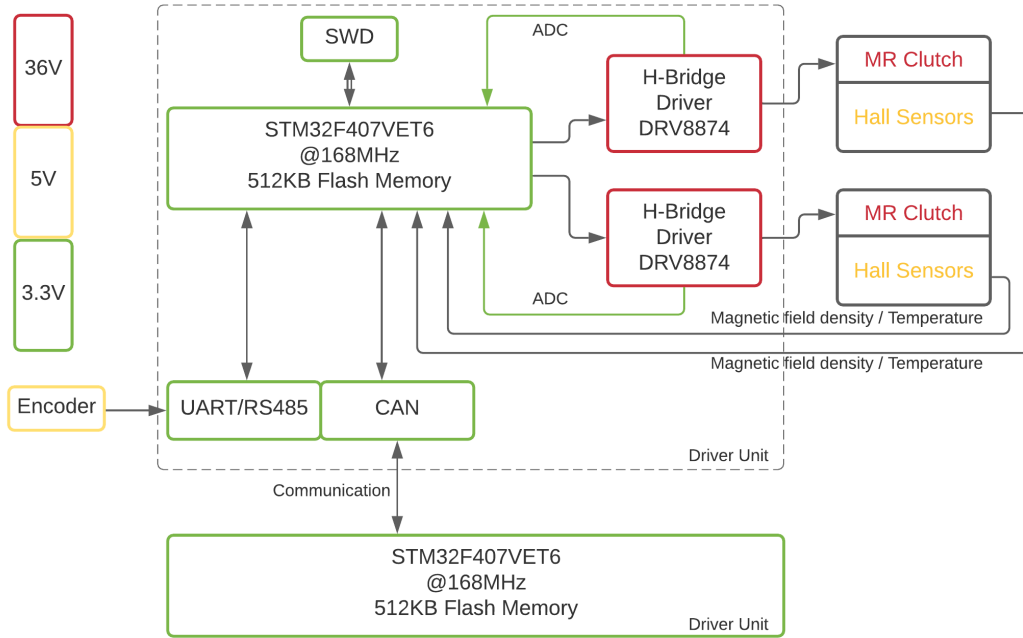


Figure 2.14: Hardware diagram.

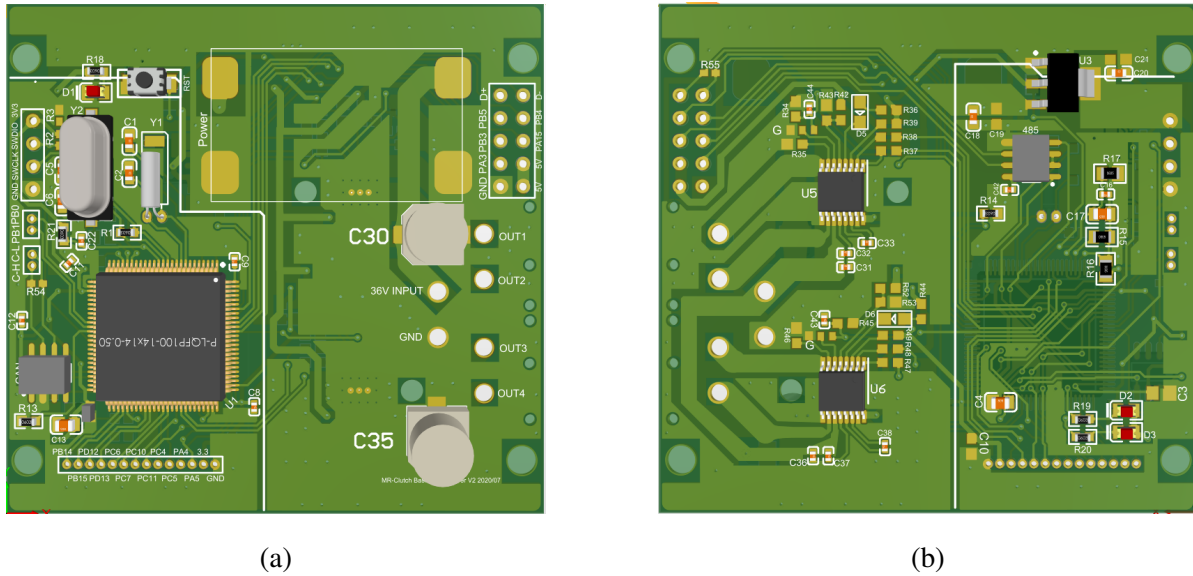


Figure 2.15: Three dimension view of the electronic driver; (a) Top layer, (b) Bottom layer.

The total power dissipation of an IC consists of three components including the quiescent supply current dissipation, the power MOSFET switching losses and the power MOSFET conduction losses [71]. The total power dissipation P_{Total} is represented as,

$$P_{Total} = P_{Supply} + P_{Switching} + P_{Conduction}. \quad (2.7)$$

The quiescent supply current dissipation P_{Supply} is represented as,

$$P_{Supply} = V_{VM} \cdot I_{VM}. \quad (2.8)$$

where V_{VM} is the nominal supply voltage and I_{VM} is the active mode current .

The power MOSFET switching losses $P_{Switching}$ is calculated as,

$$P_{Switching} = P_{SWRISE} + P_{SWFALL}, \quad (2.9)$$

which is the summation of the losses caused by the rising and falling edges of the signals where P_{SWRISE} and P_{SWFALL} are expressed as,

$$P_{SWRISE} = 0.5 \cdot I_{RMS} \cdot T_{RISE} \cdot V_{VM} \cdot f_{PWM}, \quad (2.10)$$

$$P_{SWFALL} = 0.5 \cdot I_{RMS} \cdot T_{FALL} \cdot V_{VM} \cdot f_{PWM}, \quad (2.11)$$

in which I_{RMS} represents the average output current, T_{RISE} and T_{FALL} are the output rise time and fall time, respectively, V_{VM} is the nominal supply voltage, and finally f_{PWM} is the PWM frequency.

The power MOSFET conduction losses $P_{Conduction}$ can be calculated as,

$$P_{Conduction} = I_{RMS}^2 \cdot (R_{DS(ON)HS} + R_{DS(ON)LS}), \quad (2.12)$$

where I_{RMS} is the average output current, $R_{DS(ON)HS}$ represents the high-side MOSFET on resistance, and $R_{DS(ON)LS}$ is the low-side MOSFET on resistance. The value of each of these parameters for our designed electronic driver are given in Table 2.4. Substituting these values in Table 2.4 in (2.9) through (2.12), the total power dissipation P_{Total} of the designed board is 1.217 Watt.

The value of the junction temperature T_J can be calculated as,

$$T_J = (P_{Total} \cdot R_{\theta JA}) + T_A, \quad (2.13)$$

Table 2.4: Parameters of thermal performance

Parameters	Value
V_{VM}	36V
I_{VM}	0.004A
I_{RMS}	Assume 2A
T_{RISE} and T_{FALL}	0.00015ms
f_{PWM}	6.8KHz
$R_{DS(ON)LS}$ and $R_{DS(ON)HS}$	100m Ω
T_A	80°C
Thermally attached copper area	24cm ²

where $R_{\theta JA}$ is the package thermal resistance and T_A is the device ambient temperature. The value of $R_{\theta JA}$ depends on the thermally attached copper area of current driver. In order to calculate the thermally attached copper area. We used the tool (i.e. calc. copper area) of Altium Designer. Using this tool we calculated the thermally attached copper area of the driver to be 24 cm². Referring to the relationship between the junction-to-ambient thermal resistance and the copper area as shown in Fig. 2.16 (adopted from [71]). and considering

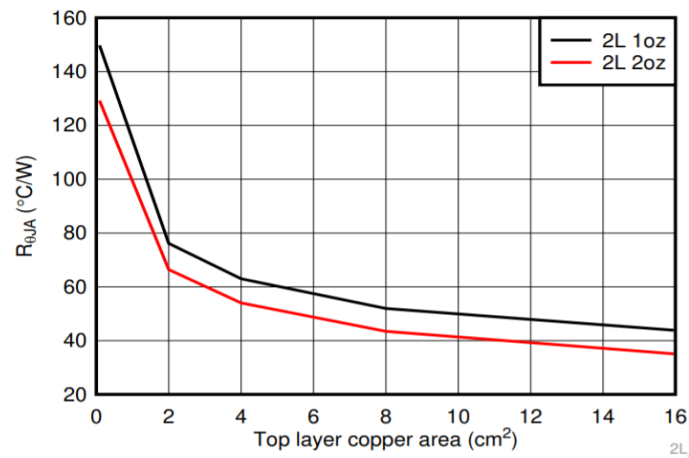


Figure 2.16: 2-layer PCB junction-to-ambient thermal resistance vs copper area.

the fact that the designed board uses the 1-oz copper layout, one can approximate the value of $R_{\theta JA}$ of the driver to be $30^{\circ}\text{C}/\text{W}$. , We assume the device ambient temperature T_A to be 80°C . Accordingly, the value of the junction temperature T_J can be obtained as 116°C . The value of junction temperature T_J is within the thermal shutdown temperature of 175°C and it is within the PCB operating region of 120°C .

In practical tests, it is rare to supply a continuous current of over 2 A to the MR clutch, except for the robot start-up. During the start-up, the current can reach for as high as 3.3A for a short period of time (e.g., 100 ms). In this case, the transient thermal performance of the driver can be calculated by replacing the $R_{\theta JA}$ with the thermal impedance parameter $Z_{\theta JA}$. The thermal impedance parameter is around $6^{\circ}\text{C}/\text{W}$ when experiencing a current transient of 100 ms. Accordingly, the value of junction temperature T_J is obtained as 98°C .

Several tests were conducted to evaluate the thermal performance of the electronic driver. The results are shown in Table 2.5. All tests were conducted when the heat sink was thermally attached. The room temperature was 23°C . The results demonstrate excellent thermal perfor-

Table 2.5: Thermal performance tests

Test condition	Duration
1A	more than 10 mins
2A	more than 8 mins
3A	210 seconds
3.3A	150 seconds

mance of the electronic driver in nominal conditions (e.g., 1 A and 2 A currents). The results are still persuasive in the case of extreme situations (e.g., 3 A and above) in which the driver can handle such high current for up to 150 s which is much longer than expected 100 ms.

2.3.5 Sensors

To control the robot position and each joint of the robot is equipped with a Zettlex absolute encoder (INC-9-150-171001-ASI1-RFC6-5-AN). To control the output torque and to monitor

the MR clutches' status (temperature), each MR clutch is equipped with Two Infineon digital hall sensors (TLE4998S4) that measure the magnetic field density inside the MR clutch and the temperature. Each joint of the robot include a pair of MR clutches configured antagonistically. There are also two current sensors used to measure the current in the coils. The feedback signals of these sensors are received by the processor to construct the control loop. The locations of these sensors are shown in Fig 2.17 (modified from [45]).

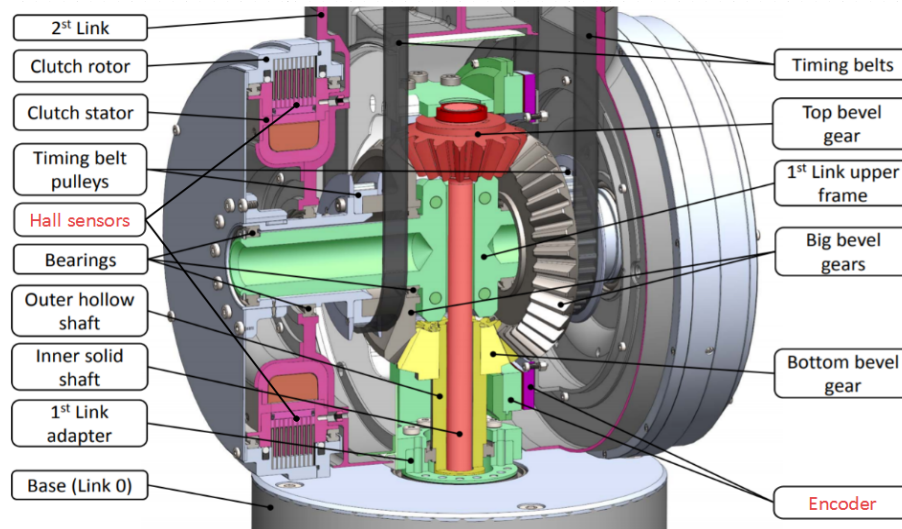
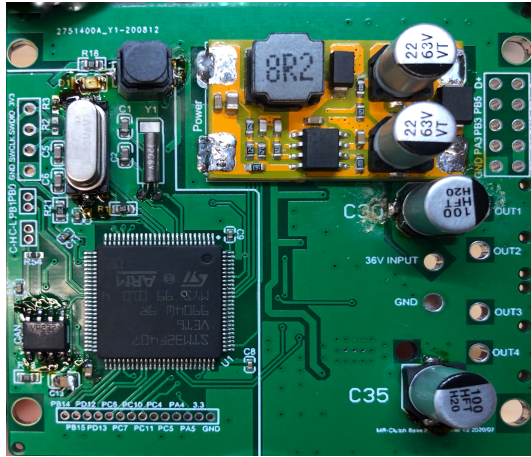


Figure 2.17: Zonal section view of Joint 2.

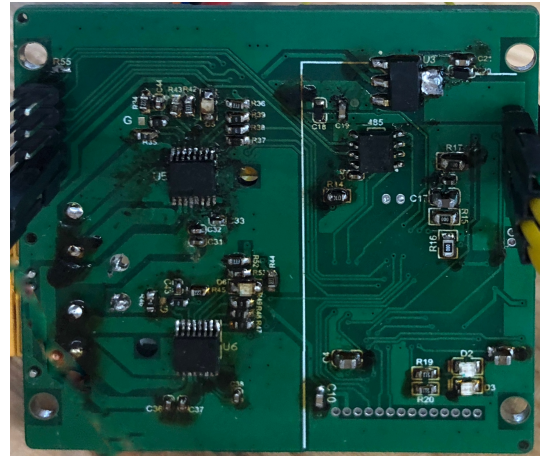
2.4 Conclusions

In this chapter, a multi-functional and miniaturized driver unit was designed and tested. The on-board components were selected according to the design requirements. The components for circuit protection and noise reduction were calculated and selected to help optimize the performance. A set of tests were conducted to evaluate the performance of the voltage conversion system. The results demonstrated that the designed system was capable of producing low ripple voltage, quick startup, stable output. The anti-shock and transient suppression abilities were also excellent. The thermal performance of the driver was carefully analyzed and it was shown that the board was capable of meeting thermal requirement of the design. The designed driver was shown to be capable of effectively handling all required inputs and outputs. The

overall size of the driver was within the desired dimensions set for the design. The pictures of the finished product of the electronic driver are shown in Fig. 2.18. Its dimension is 60 mm by 57 mm.



(a)



(b)

Figure 2.18: Pictures of the electronic driver; (a) Top layer, (b) Bottom layer.

Chapter 3

Firmware Development of a 5-DOF MR Clutch Based Collaborative Robot

In this chapter, the development of a real-time firmware for the electronic driver is presented. The portable and preemptive real-time operating system (RTOS) μ C/OS - II, is selected to construct a reliable computing environment to process sensors' data and control the robot joints. We analyze the clock systems of the host system (STM32F4) and the transplanted system (RTOS). The real-time tasks, start-up process and task scheduling of the RTOS are then studied and analyzed. The system architecture and the system flowchart are demonstrated to further explain the system. Furthermore, the tasks for sensor data processing and joint controlling are detailed and experimentally evaluated. In the end, a between-joint communication protocol is proposed to construct the robot control network. Experiments are conducted to evaluate the performance of the robot and software/hardware systems.

3.1 Introduction

The rapid technological advances in software development has led to the development of various operating systems (OS), such as Batch Operating System, Real Time Operating System, and general purpose Operating System. The OS provides an environment for running end-user applications on top of the system in an efficient and convenient manner [7]. Over the last few years, the scale and scope of robotics continue to grow. Different types of robots can have wildly varying hardware, which makes the reuse of the codes (firmware) challenging. To deal with this issue, a general framework for robotics software called Robot Operating System (ROS) was developed to allow peer-to-peer and tools-based requirements. ROS is comprised of a great number of independent programs that are continuously communicating with each other by passing messages [51]. Independent programs can be used in the fields of navigation, computer vision, grasping, trajectory planning, etc. However, ROS is not an operating system in the traditional sense of process management and scheduling. It can only provides a structured communication layer on top of the host OS, such as hardware abstraction and low-level device control [50]. Timing is the most important factor in robot control system. In some cases when the controlled object must react to control command in an extremely small amount of time, neither ROS nor general purpose operating system can meet such constraints. The robot control system also emphasizes reactivity and low latency. This makes the ROS not suitable for providing timing guarantees for robot control [83].

Contrarily, an RTOS can process a variety of individual tasks with adjustable priority levels while guaranteeing time-critical behaviour of tasks [7]. A real-time characteristic can be achieved by measuring and updating timing requirements within hundredths of seconds or shorter time increments. In each of the time increment, task scheduling can be performed base upon interrupts, tasks' priorities and events. In addition, multiple resource management functions such as semaphores and message mailboxes allow us to make better use of the resources. With these advantages, a RTOS offers a much more effective environment for robot control as well as a range of other applications such as engine control, anti-lock braking system, weapon systems, etc. There are several commercially available RTOS, such as μ C/OS, FreeRTOS, etc. This work mainly focuses on the development of a firmware based on μ C/OS-II because of its

reliability, predictability, portability and its well documented manual.

As mentioned above, an RTOS guarantees the processing of time-critical tasks. To be specific, the preemptive kernel is the core of $\mu\text{C}/\text{OS-II}$. It is responsible for task scheduling, task management and inter-task communications. Only a preemptive kernel can guarantee the system responsiveness. When a higher priority task is ready to run, the current task is preempted. The control of the CPU is then given to the higher priority task immediately. The principle of preemptive kernel is shown in Fig. 3.1 (adopted from [33]). Assume there are

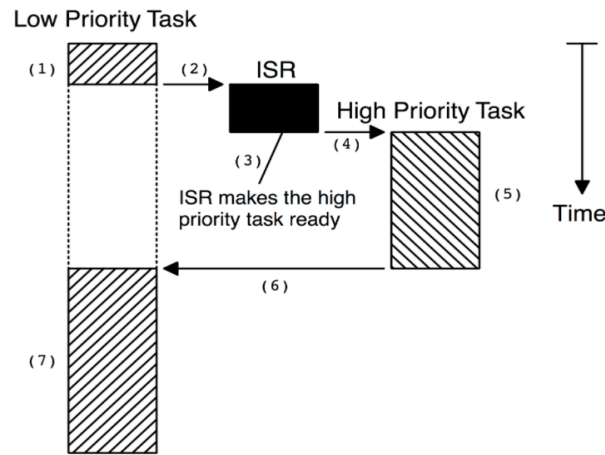


Figure 3.1: Principle of Preemptive kernel.

two tasks in a real-time program. When a low priority task is executing but interrupted by an interrupt service routine (ISR), the ISR deals with the event and searches for a high priority task that is ready to run. If a high priority task is found, then the kernel will perform context switching to execute the high priority task. When the high priority task is completed, the task is switched to a sleep status. Meanwhile, the kernel is aware that the low priority task needs to continue to execute, so a context switching is performed to resume the execution of the interrupted task. In a word, the highest priority task can always start to execute when it is made ready. In the case of non-preemptive kernel, even if a high priority task is already made ready to run, the non-preemptive kernel still requires the current task to complete or give up the control of the CPU voluntarily to perform context switching. For this reason, a non-preemptive kernel cannot guarantee the responsiveness of the system. To summarize, the RTOS with a preemptive kernel always guarantees responsiveness of the system.

In this chapter, a real-time firmware is constructed to guarantee real-time control of the robot with MR actuators. The overall system architecture and the details of each individual task are analyzed. The main objectives of constructing a real-time firmware is to (a) Process the data from multiple sensors, (b) perform real-time communication between devices in each joint, (c) perform real-time communication between joints (d) perform closed-loop feedback control at joint level. The real-time program is assessed in various scenarios to evaluate the performance of the system.

In the remainder of this chapter, the clock system of the STM32F4 microprocessor will be analyzed first in Section 3.2. Section 3.3 discusses the key components of RTOS, including the clock system, task structure, task scheduling, etc. Section 3.4 presents the overall system architecture of the robot control firmware and each part of the system is detailed and analysed. Section 3.5 evaluates the performance of the robot and software/hardware system through rigorous experiments. Finally, Section 3.6 concludes the chapter.

3.2 STM32 and RTOS Platform

$\mu\text{C}/\text{OS-II}$ is designed to be portable to multiple microprocessors. In our case, STM32F407 processor is chosen as the platform for running RTOS. The high clock speed of the processor (i.e., 168MHz) and its relatively large flash memory (512KB) make this processor a suitable platform to run $\mu\text{C}/\text{OS-II}$. All real-time kernels rely on an accurate clock system to handle internal processes such as task scheduling, interrupts generating and the other advanced operations. In this work, the main clock source of the STM32F4 processor provides the clock source for the RTOS kernel.

The configuration of STM32 clock system includes the selection of the system clock source, the configuration of phase lock-loops (PLL), the advanced high-performance bus (AHB) and the advanced peripheral bus (APB). There are three frequency sources that can be used as the STM32 system clock source. The high speed external crystal (HSE) has the advantage of producing a very accurate rate on the main clock. The high speed internal (HSI) crystal has a faster startup time than the HSE crystal oscillator. However, even with calibration, the frequency of the HSI is still less accurate than that of the HSE crystal oscillator. The PLL

clock (PLLCLK) can also be used as the system clock source. We used the last option, i.e., the PLLCLK, as the system clock source. The PLL uses an 8MHz HSE as its input source. The configuration of PLL includes the selection of appropriate multiplier and divider factors. The output of PLL (PLLCLK), is fed to system clock source (SYSCLK). The SYSCLK signal is fed to AHB with no pre-scaling. AHB provides the clock signal for the CPU clock (HCLK) and the Cortex system timer (Systick). Systick is the clock source of the RTOS (labelled with red in Fig. 3.2). AHB also provides a pre-dividable clock frequency for the APB. Both sub-classes

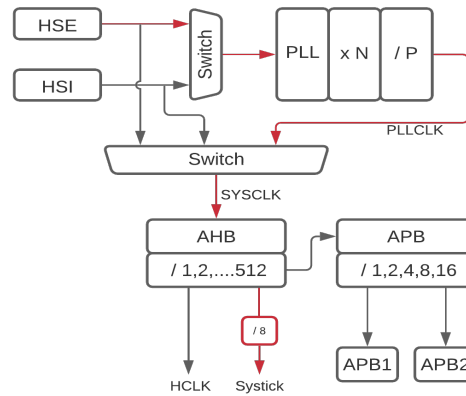


Figure 3.2: Clock system of STM32.

of APB provide the clock signal for the peripherals' clock and timers' clock.

3.3 μ C/OS-II Real-time Operating System

μ C/OS-II is a real-time operating system capable of performing multitasking in real-time using of a preemptive kernel. μ C/OS-II allows to maximize the utilization of the CPU while simplifying the system design by separating the application into several tasks. The tasks can be scheduled and managed by the kernel. μ C/OS-II relies on hundreds or even thousands of interrupts in one second to handle its operation through a specific Interrupt Service Routine (ISR) known as the kernel. A main responsibility of the kernel is task scheduling. The kernel determine if there is a higher priority task ready to run and perform task switching and associated inter-task communications. This process is known as task scheduling. Task scheduling allows

every eligible task to have a chance to execute within millisecond level.

3.3.1 Clock System

An RTOS relies on an accurate clock often known as tick to perform task scheduling and other necessary operations such as time delays and inter-task communications. In STM32, a count-down timer is used to generate the ticks for an RTOS used with the STM32 processor. The STM32 count-down timer is called SYSTICK. It is a 24-bit down counter that is clocked by one-eighth of the AHB clock signal (e.g., 21MHz). The reload value of SYSTICK determines the clock frequency of ticks. We have setup the reload value of SYSTICK to have a 1-millisecond clock cycle for ticks.

3.3.2 Task Structure

An RTOS improves program execution by separating the application into multiple tasks. Each part is responsible for a portion of the problem that the program is handling [33]. Every task has its task control block (TCB), which includes its priority, stack pointer (SP), etc. Each task also has its stack to store the CPU registers, including program counter (PC) for recording the current address of the code and several general registers (see Fig 3.3, adopted from [33]). This structure allows the task to be resumed from exactly where it was preempted previously.

Each task is an individual infinite loop that can be in one of the following five states: Ready, Running, Waiting, Dormant, and Interrupt (see Fig 3.4, adopted from [33]). A ready task with the second highest priority can be executed as long as the current highest priority task is completed. A task in running state has the control of the CPU. A task in waiting state means that it is waiting for the occurrence of a particular event such as timeout or semaphore. A task in dormant state resides in memory but has not been made available to the kernel for scheduling [33].

3.3.3 Startup Process of the RTOS

The startup process of the host system can be separated into two parts, which are the initialization of peripherals and parameters. μ C/OS-II as well needs to be initialized. The first step is

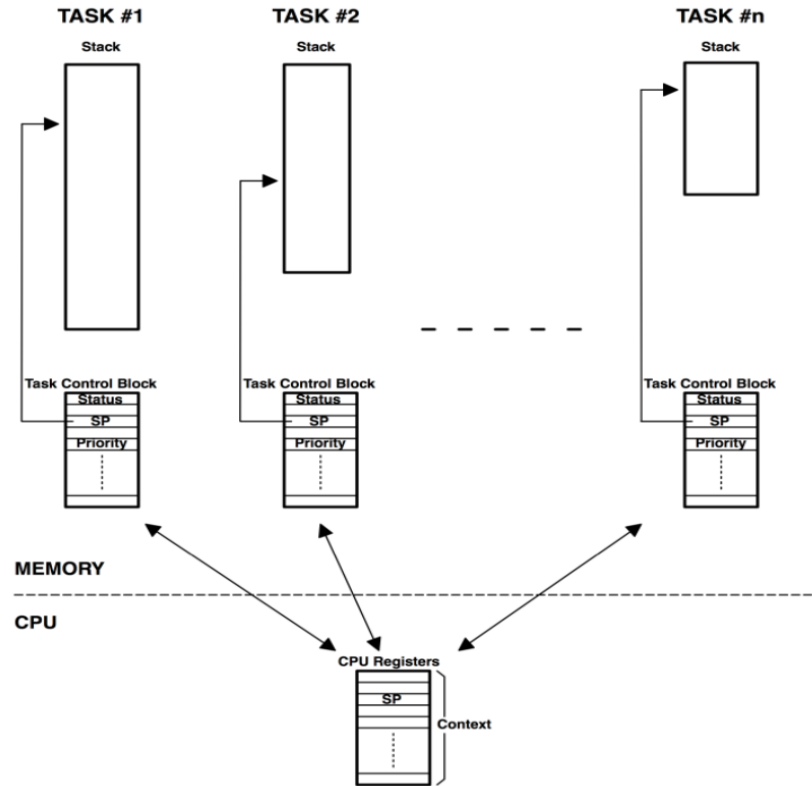


Figure 3.3: Task structure.

to initialize all the variables, data structures, and lists. The idle task and statistic task are also created. The idle task is the lowest priority task and it runs when there are no other tasks ready to run. The statistic task is the second lowest priority task which the statistic data such as CPU usage. The second step is the creation of a real-time task. This procedure can either happen prior to the start of multitasking or inside a running task. It includes the creation of the task stack and task control block, etc. The last step invokes the kernel to perform multitasking. It involves finding the highest priority task's priority number, pointing to the highest priority task that is ready to run and finally executing target code.

3.3.4 Task Scheduling

As mentioned before, the RTOS separates the whole application into individual tasks. Then the task scheduling can be triggered by many causes such as systick interrupts, external interrupts, semaphores, event flags and time delay, etc. (see Fig. 3.5) This section will discuss various

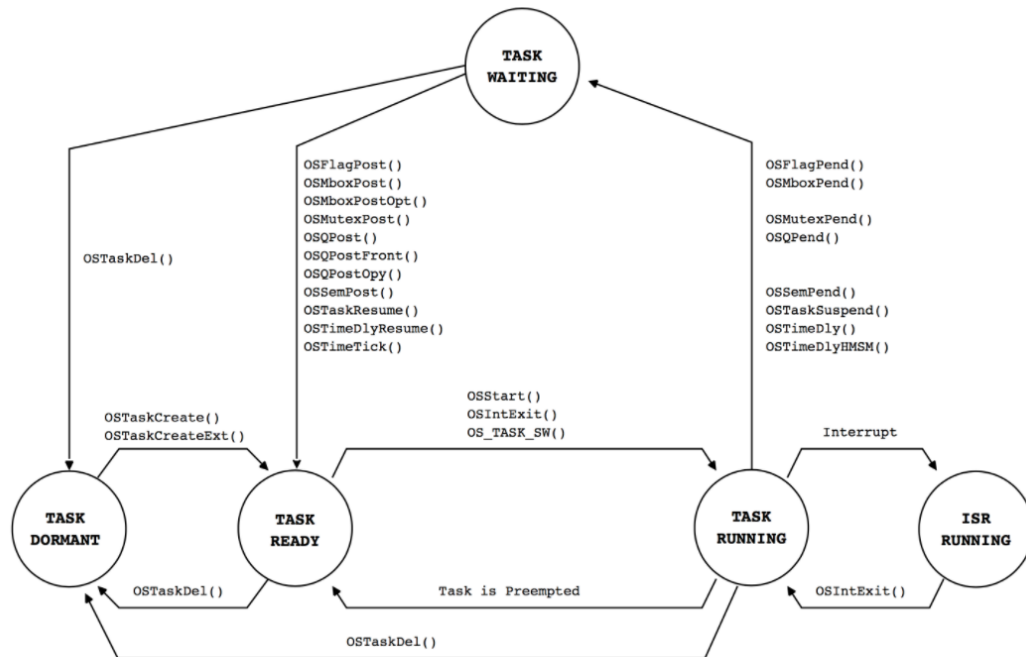


Figure 3.4: Task states.

scenarios.

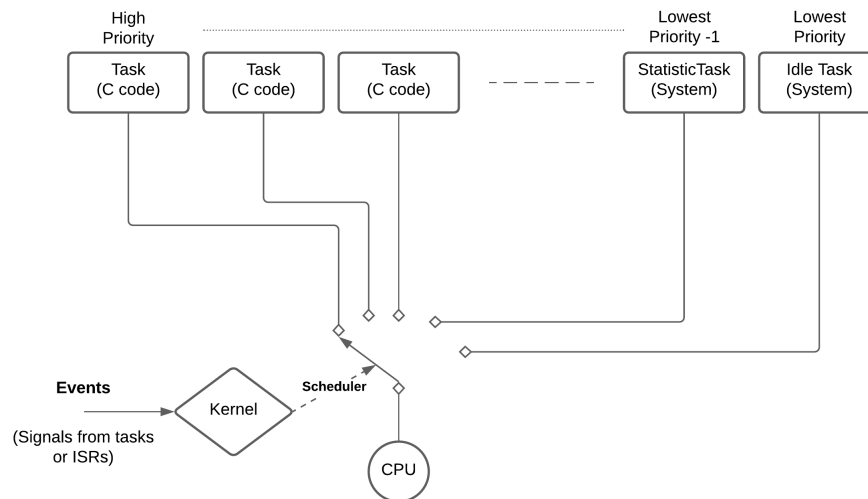


Figure 3.5: Concept of task scheduling.

A typical program context that includes an external interrupt is shown in Fig 3.6. In this figure, the internal interrupts generated by ticks act as the heartbeat of the system. The fig-

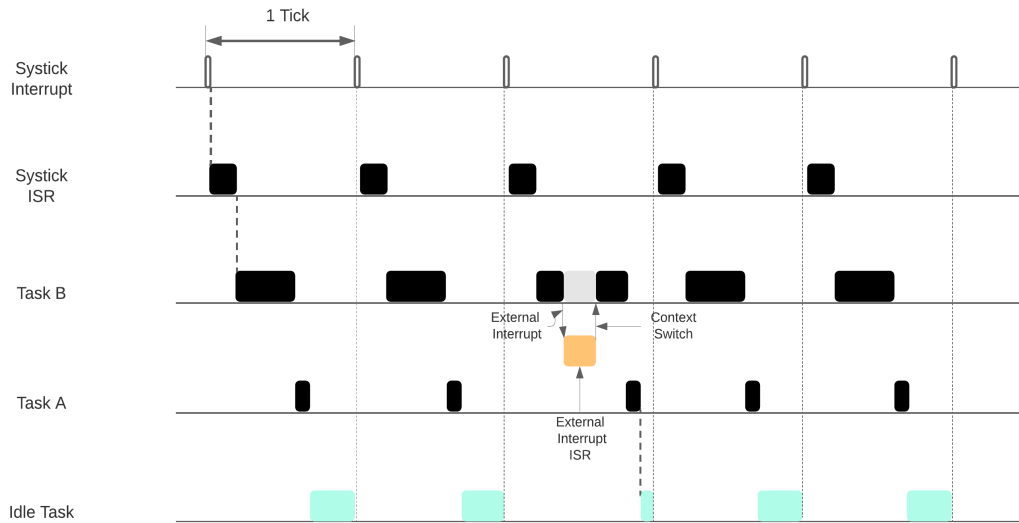


Figure 3.6: External interrupt.

Figure 3.6 shows periodical ticks (1ms or more) as systick. Each systick invokes the systick ISR to perform task scheduling. It is assumed that task B has higher priority. Hence, task B is immediately executed following the systick ISR call. After task B releases the control of the CPU by calling a time delay upon completion. This allows task A (the lower priority task) to start running. After the completion of task A, the idle task is invoked in the same fashion and runs for the remainder of the tick cycle. During this time, the idle task keeps the CPU running. If an external interrupt occurs during task execution, a context switching will be made to the external interrupt ISR. After the external interrupt is processed, the kernel switches back to the original task or the highest priority task that is ready at the time.

Another important functionality of the kernel is inter-task communication. A typical program context that includes inter-task communication is shown in Fig 3.7. A common mechanism for achieving inter-task communication is semaphore. Again, it is assumed that the priority of task B is higher than task A, and task B is waiting for a semaphore to be posted by an external interrupt ISR (e.g., in reaction to an external movement such as pressing a button). During the first two ticks, both tasks can be complete their functions within one tick. However, as seen, the external interrupt can happen at different point-in-time. In some cases

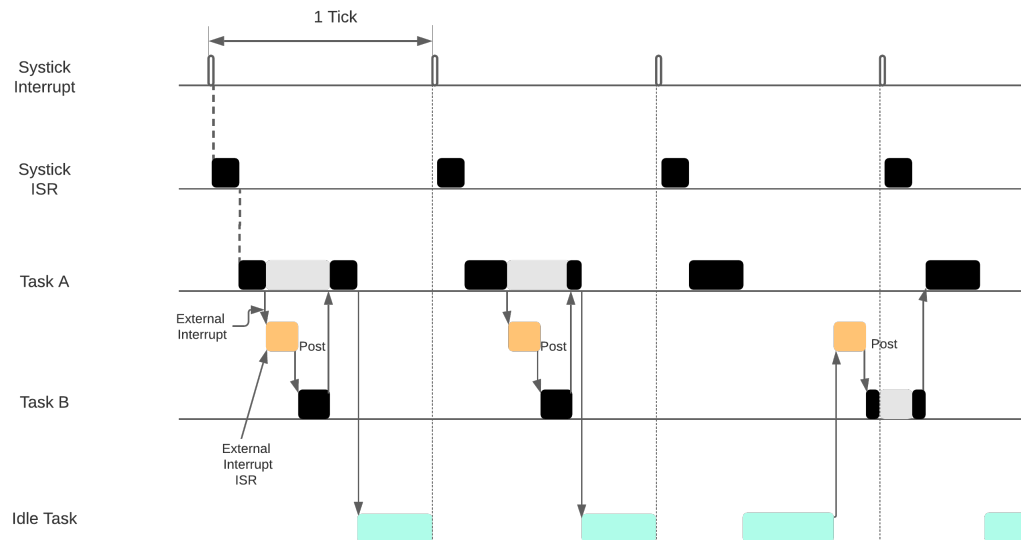


Figure 3.7: Task-wise communication.

such as the one shown during the third tick cycle, the external interrupt can happen right before the following systick interrupt. As a result, task B can be interrupted by systick ISR. Upon the completion of systick ISR, the scheduler notices that task B is still in ready status, so it is executed until task B complete its operation. It can also be posted by other tasks. The same mechanism applies to the other task-wise communication methods like event flag, message mailbox, etc.

A time delay can also trigger task scheduling as shown in Fig 3.8. In this figure, task B has a higher priority than A. The external interrupt interrupts the execution of task A, and post a semaphore that makes task B ready. Hence, task B immediately starts running. If for any reason task B is required wait for a given time interval, e.g. 2 ticks, At the same time, the scheduler switches back to task A and continue to schedule other lower priority task. The execution of lower priority tasks are not affected by the delay in task B execution. Upon completion of the time delay, task scheduling include task B once again.

In summary, there are several reasons that can trigger task scheduling and context switching as described above. Fig 3.9 shows the details of task scheduling and context switching. When a low priority task A with interrupt disabled is executing, an external device generates an external

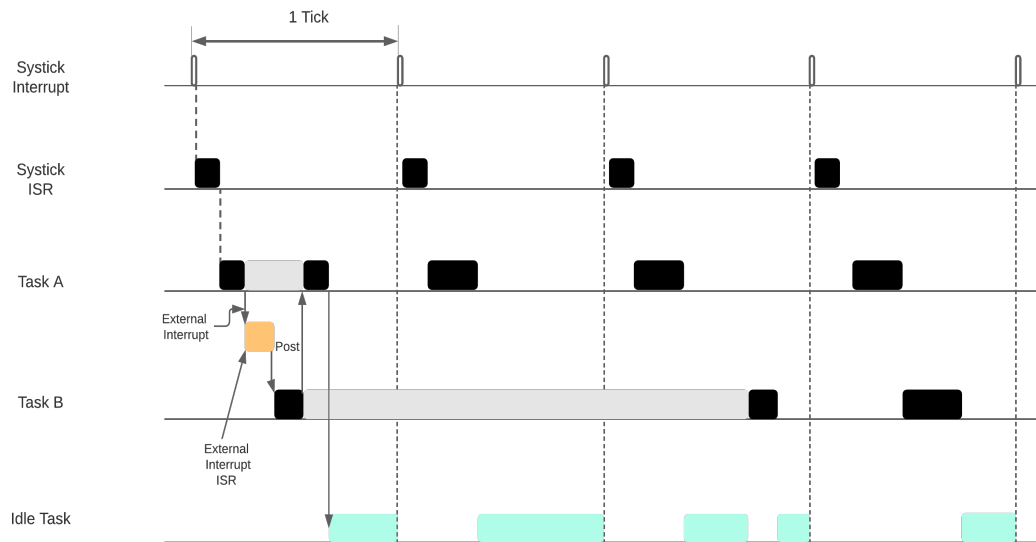


Figure 3.8: Time delay.

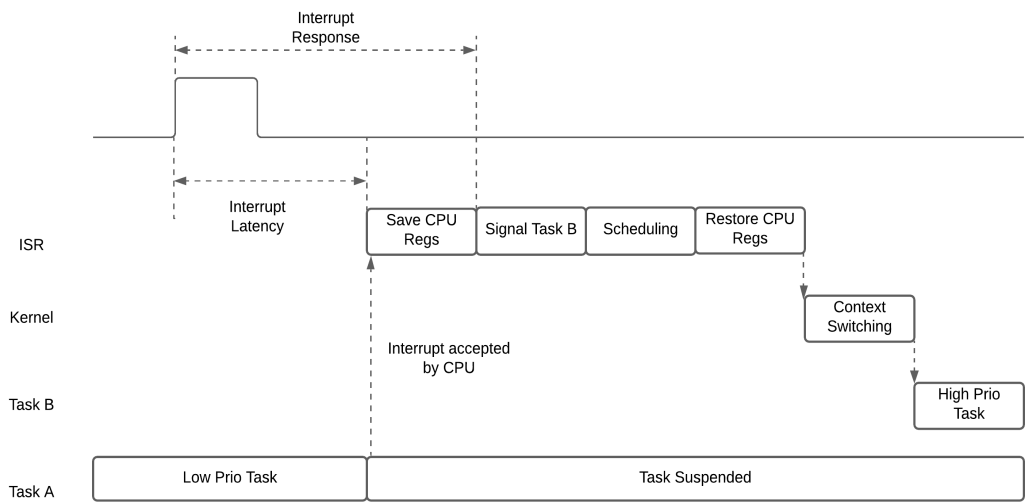


Figure 3.9: Procedures of task scheduling.

interrupt. However, the CPU cannot detect the interrupt until the low priority task re-enables the interrupt. After the interrupt is re-enabled, the CPU detects the interrupt. It saves all of its registers to the relative task stack, and a signal is sent to the high priority task B that is waiting for the device. This signal set the high priority task to ready state, and its relative registers are restored. Then the kernel performs context switching, and record the stack pointer of the

interrupted task A. Eventually, the high priority task B can execute and process the signal from the external interrupt. To determine which task should be returned to, after the CPU services the external ISR or systick ISR, the scheduler will search for the highest priority task among all the ready-to-run tasks. If it is still the original task, then context is restored and the task continues to execute. Otherwise, another task with the highest priority that is ready to run will be given the control of the CPU.

3.4 The Firmware Design of Driver Unit

The firmware design of the driver unit according to the requirements of the robot control system is illustrated in Fig 3.10. The firmware structure of the system can be divided into three layers.

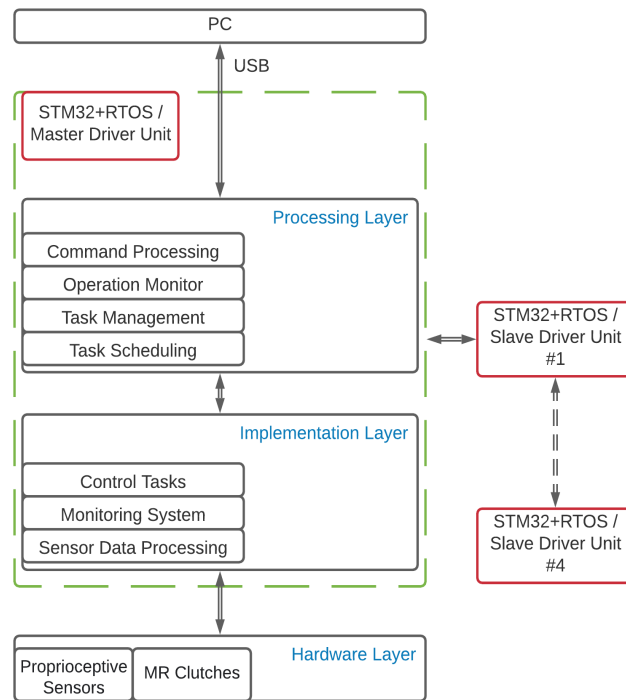


Figure 3.10: Real-time system architecture.

The processing layer is responsible for transmitting the monitoring data to PC interface in real-time. It also receives the high-level control commands from the PC in real-time, such as the trajectory planning commands and the torque control commands, etc. After received the

commands, the real-time task will process the commands and pass them to the implementation layer. The processing layer also initializes the RTOS and the multiple real-time tasks. Finally, it performs task scheduling.

As the central part of the system, the implementation layer conducts basic joint control actions including joint position control and joint torque/current control. This layer also processes all sensors' data such as hall sensors, current sensors, and encoder, etc. Additionally, the monitoring system is responsible for restricting the components within the safety range such as coil (MR clutch) temperature, coil current and position of the joint. These tasks are all operated in the form of real-time tasks, therefore promise the performance of the system. The robot control network is constructed by configuring the driver units of five joints in master-slave mode. The communication protocol can be either CAN bus (CAN controller) or RS485 (USART). The details will be discussed later.

3.4.1 System Operation Process

The firmware structure of the robot control system is described in the above section. To give a clear and definite view of the software system operation process, a system flowchart is illustrated in Fig 3.11.

The real-time robot control system firmware in this work runs on an STM32F4 platform. The AHB of the STM32 processor provides the clock source required by the RTOS kernel. When the board is powered up, the STM32 clock is first initialized. The second step is the initialization of the related peripherals such as GPIO, ADC, DMA, Timer, etc. Then, the clock system and data structure of the RTOS are initialized. Subsequently, we can create all the user tasks related to sensor data processing and communication, etc. In the end, the kernel/scheduler will perform task scheduling and run the tasks based on the priority. However, if the safety monitor detects errors, such as overheat, the system will be shutdown.

3.4.2 Tasks

The system architecture and system flowchart have already been outlined in the above sections. Among the three layers, the implementation layer will do most of the jobs. It includes the

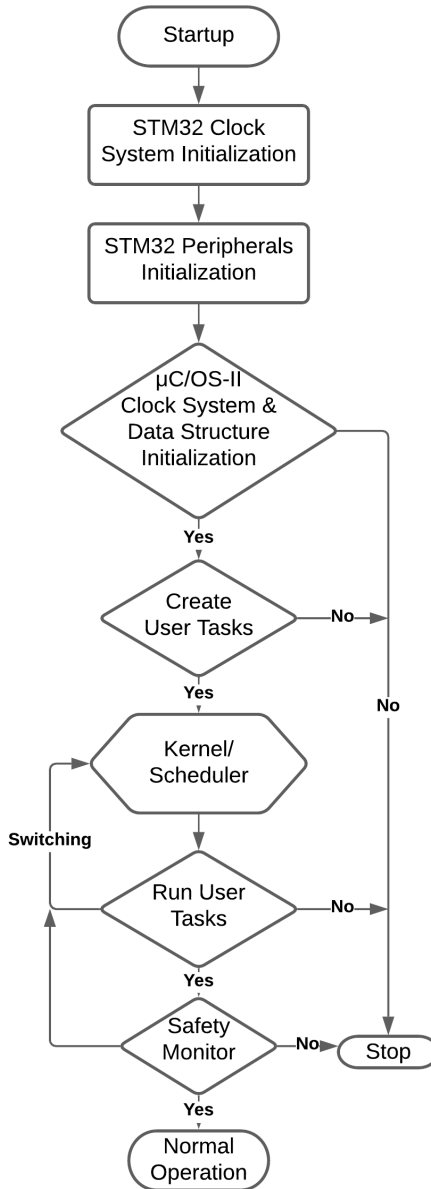


Figure 3.11: System flowchart.

control tasks and the sensor data processing tasks, etc. The details of these tasks are illustrated in Table 3.1.

The Hall sensor task is used to receive and process the magnetic field density and the temperature that are sent through Single Edge Nibble Transmission (SENT) protocol. Afterwards, these sets of data can be used in closed-loop torque control task. Encoder task is used to receive and process the joint position data that is sent from an absolute zettlex encoder through

Table 3.1: Task information

Tasks	Priority	Functionality
Hall Sensor	10	Process magnetic field data
Encoder	4	Process position data
PWM	8	Generate PWM signal
ADC	7	Process clutch current
Position Control	5	Execute joint position control
Current Control	6	Execute joint current control
Communication	9	Construct control network
Monitor	12	Monitor safety issues
Statistic	63	Calculate CPU usage
Idle	64	Keep CPU running

Asynchronous Serial Interface (ASII) protocol. It provides a feedback signal for joint position control. PWM task uses several specific timers that work in output compare mode to generate the required PWM signals. The PWM signals are then input to the H-bridge current drivers on-board. Therefore the drivers can output reversible, adjustable and fast-response current to the electromagnetic coils of the MR clutches. The ADC task is used to provide the current feedback signals that are sampled from the MR clutches. They are used in both monitoring task and current control task. The ADC data transmission is achieved through Direct Memory Access (DMA) which avoids the involvement of the CPU. This method partly released the system load.

The position control task, which uses the encoder data as the feedback signal to construct the position control loop. The proportional–integral–derivative (PID) control algorithm is used to provide fast-response output signal to change the duty cycle of PWM signal. The details of the algorithm will be discussed in the next section. The current control task uses the sampled current as feedback signal to construct the current control loop and achieve the open-loop torque control. The PID algorithm is also used in this task, because the advantages of fast

reaction to disturbances. Finally, the last user task, monitor task is used to monitor if the robot is within the safe range, such as joint limit, clutch temperature and coil current, etc. Additionally, the statistic task is created by the system as the second lowest priority task to calculate the CPU usage. The idle task is also created by the system as the lowest priority task and keep the CPU running when none of the other tasks is ready to run.

This section briefly describes the functionalities of the tasks. A sequence chart is shown in Fig. 3.12 (a) to demonstrate the time spent on each task and the sequence of task scheduling when the robot is performing position control actions. The program context of the system is shown in Fig. 3.12 (b) to present the relevant tasks that the system needs to execute in a time cycle.

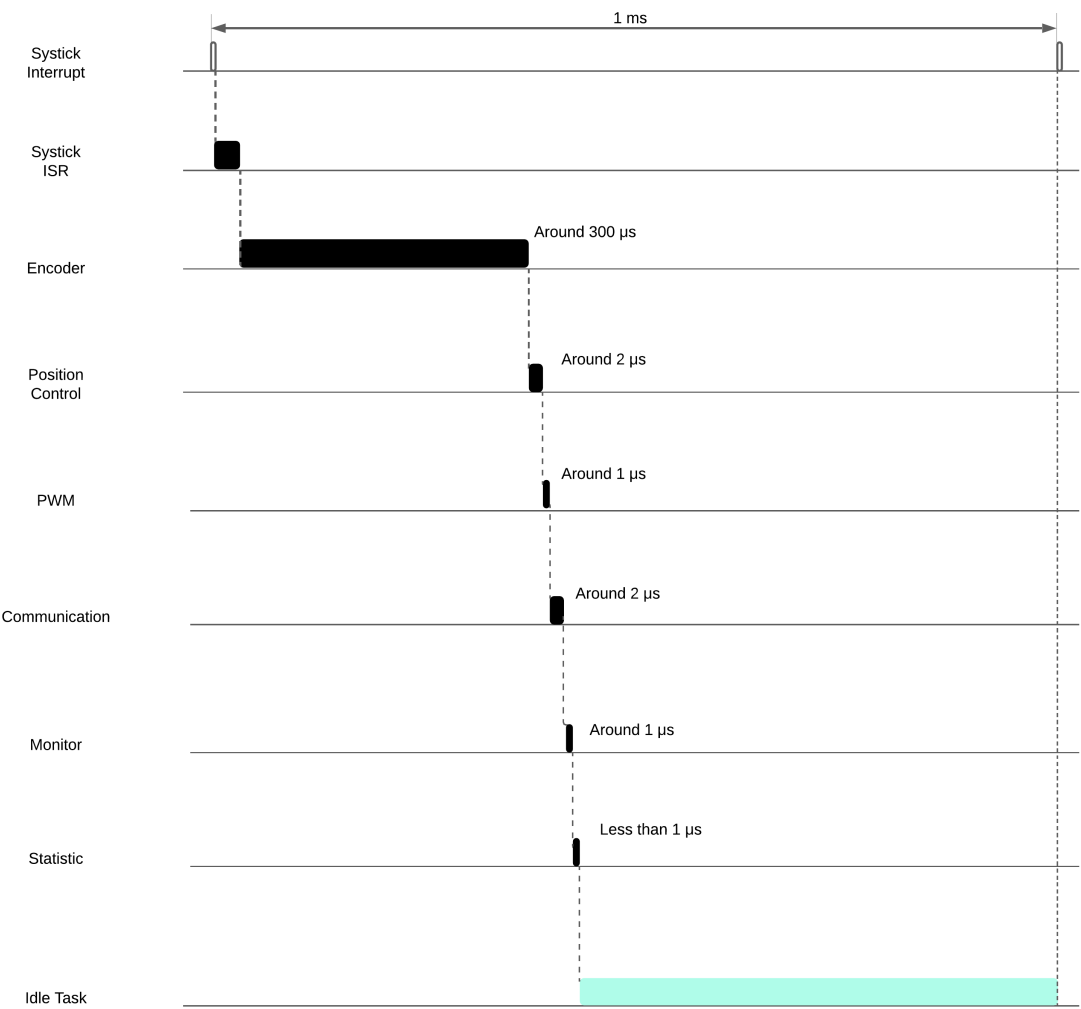
3.4.3 Data Processing and Control Algorithms

The low-level control algorithms cannot execute properly if without the precise reading of the sensors. This section will further discuss the working principle and program interface of the sensors and control algorithms.

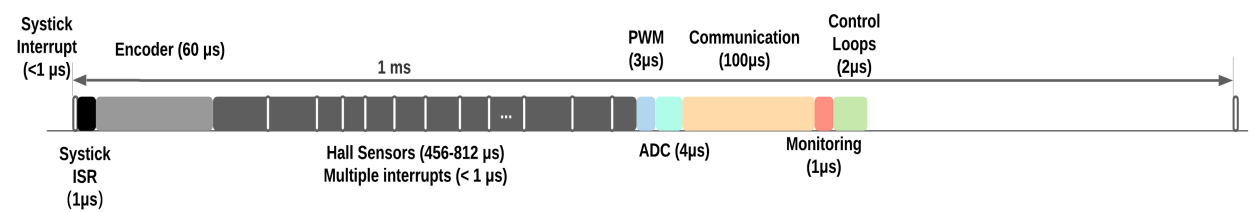
Hall Sensor

Hall sensor (TLE4998S4) is a significant part of torque control loop. It transmits the magnetic field density and the temperature data. The sensor supports a basic version of the SENT protocol defined by SAE J2716 [53], a implementation of the digital pulse scheme for reporting sensor information. It provides a point-to-point scheme for transmitting signal values from a sensor to a controller and allows the transfer of high-resolution data with a low system cost. The basic unit of time in SENT protocol is called a tick, where a tick can between 3 - 90 μs (3 μs in this case) [63]. The minimum data unit is called a nibble, which communicates 4 bits of data encoded in the combined pulse timing of an initial fixed-width low period (3 ticks) followed by a variable-width (12-27 ticks) high period [24], representing values from 0 to 15. Its encoding scheme is shown in Fig 3.13.

A synchronization/calibration period (168 μs) always flags the start of a message frame and provides a synchronization point for measuring the tick time of the SENT output [4].



(a)



(b)

Figure 3.12: Task, (a) program sequence chart, (b) program context.

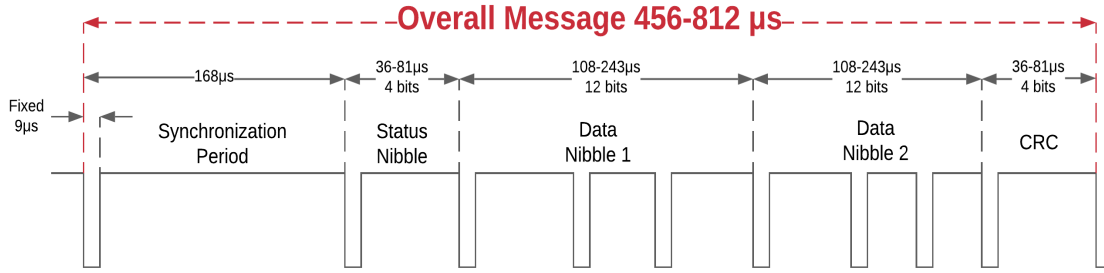


Figure 3.13: SENT Frame.

However, due to the variation between the transmitter (Hall sensor) clock and receiver (driver unit) clock. It is allowed to have a synchronization/calibration trim within a timing error of less than 20 percent of clock error. Careful measurement of this synchronization/calibration pulse allows the receiver to reliably normalize and interpret nibble periods as data values [63]. The next is the status nibble. This nibble represents for error code (overvoltage, normal state, etc.) and the measurement range (+/-50,100,200 mT). Then, a sequence of six data nibbles (12 to 27 clock ticks each) represent the values of the signals to be transmitted. The first four data nibbles transfer the magnetic field density value of the selected magnetic range. The last two data nibbles transfer the temperature (range from -55 to 200 Celsius). The cyclic redundancy code (CRC) nibble is used to diagnose the error and verify the result. The CRC is calculated using a polynomial $x^4 + x^3 + x^2 + 1$ with a seed value of 5 (0b0101) [25].

The method of decoding the SENT protocol is called input capture. First, a hardware interrupt is triggered when the timer detects a rising edge. At this moment, the counter starts to count with 1MHz (1μs per count) frequency until another interrupt is triggered by a falling edge. The number of counts in between (high-level) represents the value from 0 to 15. All the values are recorded to calculate the final results of magnetic field density and temperature through bit-wise operation and conversion of number systems.

Encoder

The protocol that is used in transmitting hall sensor data and the decode algorithm are analyzed in the above section. This part continues to detail the encoder data processing. The Zettlex encoders are non-contact devices for precise angle measurement. They are consisted of two

main parts: a stator and a rotor. Each of them shapes like a flat ring. Among which, the stator is powered and measured the angular position of the passive rotor [89]. Such encoders have advantages of no contacts and compact size, etc. The communication interface of the encoders is Asynchronous Serial Interface (ASII). Data is transmitted by the encoder continuously formatted into frames. Each frame consists of six 8-bit data. Each 8-bit data is transmitted from a standard Universal Synchronous Asynchronous Receiver Transmitter (USART) using 8-N-1 (no parity, 8 data bits, 1 stop bit) configuration. The receiving end customizes the USART ISR to store, process, verify and calculate the received angle position data. The data format is shown in Fig 3.14.

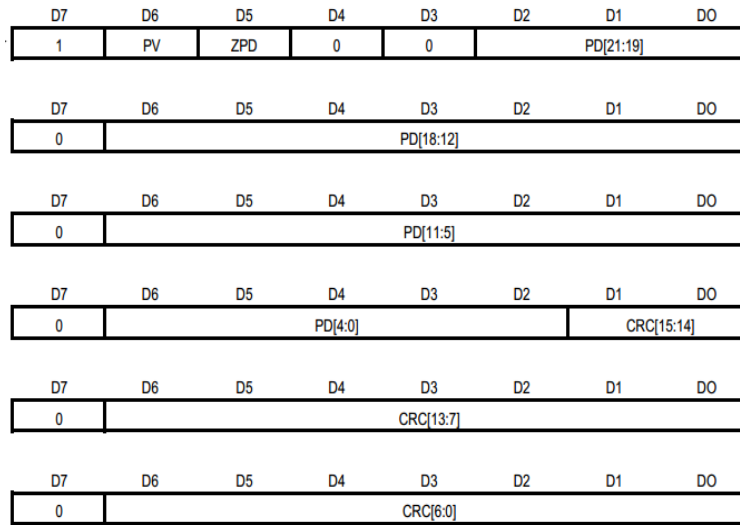


Figure 3.14: ASII Data Frame.

The first byte contains two flags, which are Position Valid (PV) and Zero Point Default (ZPD). They are used to verify the validity of position data. The Position Data (PD) is separated into the first four bytes of data transmission. The data for CRC verification is separated into the last three bytes of data. In the period of data verification, the whole frame is verified through CRC16/BUYPASS verification algorithm. The CRC parameters include the polynomial ($0x8005 / x^{16} + x^{15} + x^2 + 1$) and initial data (0x0000), etc. The algorithm of CRC is illustrated in Fig 3.15. The first step is to move the first byte of the encoder data 8-bit to the left to XOR the CRC initial value. Then updates the CRC value. The most significant bit (MSB) of the updated CRC value is pass to “IF(MSB == 1)” statement. If it is true, then CRC is left

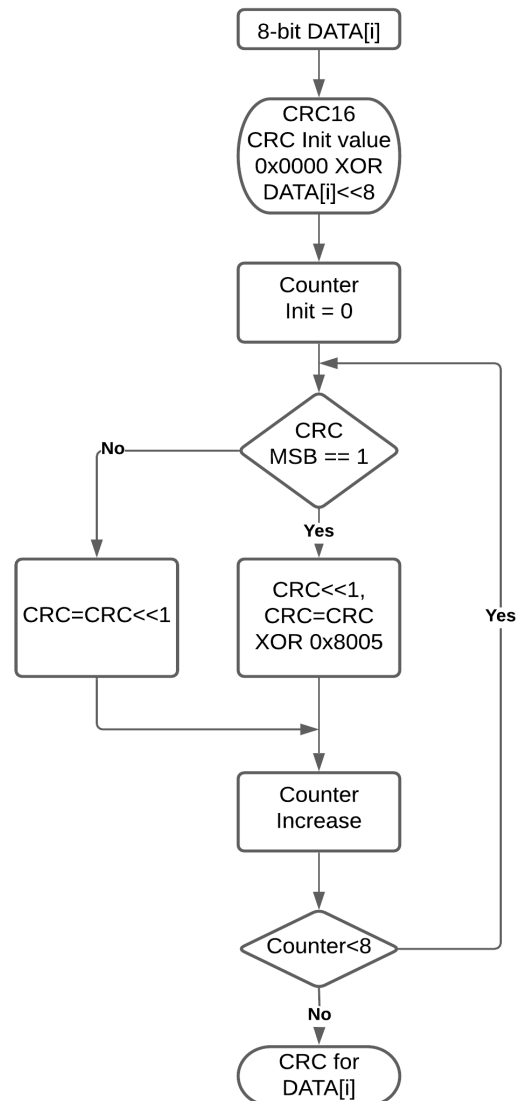


Figure 3.15: CRC-16 algorithm.

shifted with one bit and XOR the polynomial. If not, the CRC is only left shifted with one bit. Then, the counter increases, the same procedure is repeated until loop around all the bits. Finally, the CRC result is compared with the received CRC to verify the validity. If the result matched, the data will be processed and stored, otherwise, the data will be discarded.

ADC

Analog-to-Digital Converter (ADC) is a system that converts an analog signal into a digital signal. It plays an important role in constructing the current control loop. In this work, the ADC measures the voltage drop across the current sensing resistor (0-3.3V), which indicates the load current in the range of 0 and 3.3 A.

The software implementation is as follows. The ADC channel is triggered by a timer's PWM signal every 1 ms. The resolution, sampling time and two-sampling-delay are configured to manage the duration of ADC conversion. We set the conversion time to $4\mu\text{s}$. To further reduce the load of CPU, DMA is also used to transfer the ADC data to the memory (peripherals to memory) without the involvement of CPU.

Communication Protocol

The communication network is the most significant part in implementing the overall decentralized control of the robot. Therefore, a between-joint communication protocol needs to be carefully designed to meet the requirements. With this communication protocol, the scheme of task-level control can be achieved. At the first stage, both CAN bus and RS-485 communication are the candidates of this work, as they both are commonly used. Besides, they both feature differential output and the fault protection. The differential output means that it has no reference to the ground (see Fig. 3.16, adopted from [38]), rather, it is measured between

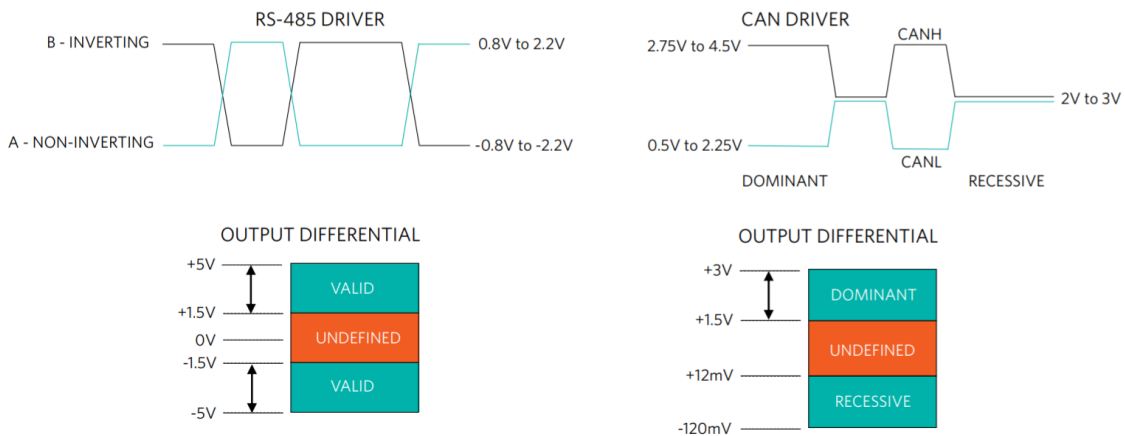


Figure 3.16: Differential output.

the two output wires. Its main benefit is the ability of noise rejection. Both of the wires are subjected to the noise influence, therefore the noise can be filtered out by the data acquisition system. In the other single-ended communication methods such as serial communication, only one wire is subjected to the noise. As a result, it leads to the error in data transmission. The termination resistors (120Ω) are also used in the design to match the characteristic impedance of the transmission line. Another advantage of both CAN and RS-485 transceivers is the internal overvoltage protection circuit to protect the chip from accidental shorts between data lines and power supply.

However, the major reason why this work uses CAN bus rather than RS-485 transceivers is the message handling mechanism. In a RS-485 system with many nodes, such as the robot between-joint communication. When multiple joints are communicating to the main controller simultaneously, several messages are sent out onto the bus simultaneously by different nodes. Therefore, it results in a collision of messages. Furthermore, if multiple RS-485 transceivers on the bus are in one state and only one transceiver is in the opposite state. In such condition, the lone RS-485 will cause significant current draw that will cause thermal shutdown of the IC or permanent damage to the system [38].

In this case, the CAN bus has a huge advantage over the RS-485. The CAN bus nodes are preset with the sending priority which is indicated by the number of consecutive dominant states. The dominant (logic 0) and recessive (logic 1) states mean that the voltage difference between two output wires are around 2.5V and 0V respectively (see Fig. 3.16). During contention, the message with the most dominant state will continue to transmit, the others will stop to transmit. This advanced technique is called bus arbitration. Additionally, the arbitration is achieved without additional controller. A CAN controller integrated within the microprocessor is responsible for the bus arbitration (see Fig. 3.17).

The CAN message data frame format is shown in Fig 3.18. The arbitration is determined by the ID field (green). For example, the node one and two in Fig. 3.19 want to send messages simultaneously. At this moment, the priority of sending need to be arbitrated. Therefore, the CAN controller starts to compare the number of consecutive dominant states of these two nodes. Finally, Node two can start to transmit because Node one loses arbitration at the bit labelled with red. Additionally, the selected STM32F4 MCU's CAN controller is capable of

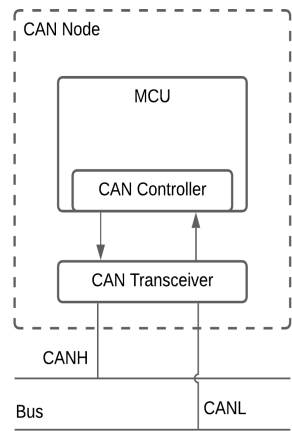


Figure 3.17: CAN bus.

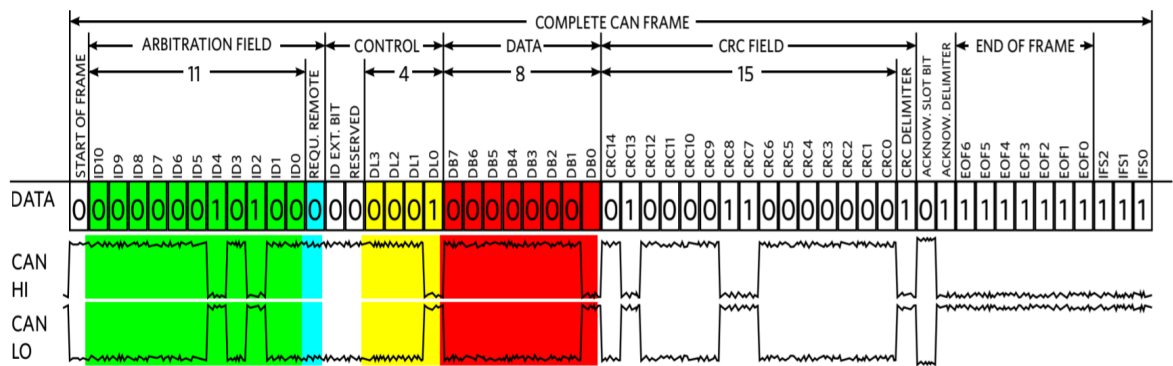


Figure 3.18: CAN data frame.

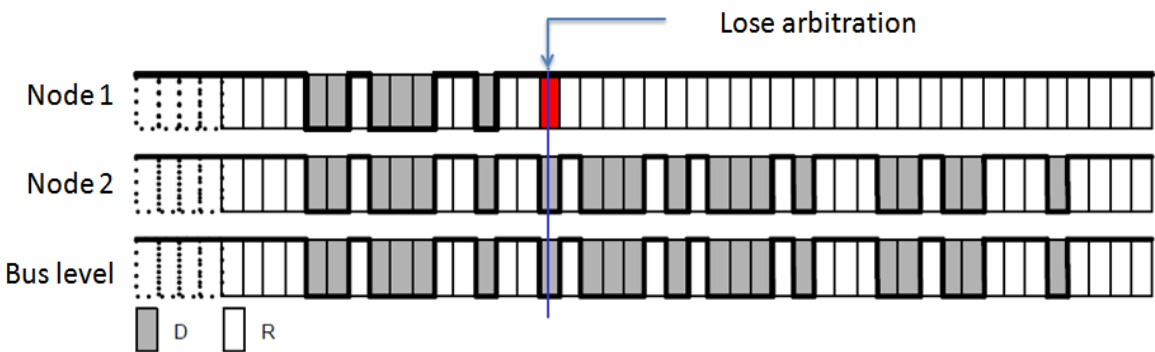


Figure 3.19: Bus arbitration.

filtering the received messages based on the ID. It further reduces the CPU load, because only the matched messages are received.

On the contrary, RS-485 cannot achieve bus arbitration in hardware level. Therefore, a stable software bus arbitration achieved by interrupt must be built. In consequence, the load of the microprocessor and real-time system are increased.

Apart from the features listed above, CAN bus transmission also contribute to the robustness of the system. Because the messages are continuously checked for errors. The corresponding indication bits are CRC and Acknowledgement (ACK). A CRC result is calculated by the transmitter and filled in the CRC field. The receiver uses the same polynomial to calculate the CRC of the sent bits. Then, the CRC result are compared with the received. If it matches, the frame is stored and the receiver transmits a dominant state in the ACK bit, overwriting the recessive state of the transmitter. Otherwise, acknowledgement errors are detected. This indicates a transmission error detected by the recipients.

Additionally, a frame of CAN message has the ability to transmit 0 to 8 bytes of data. We need to specify the message ID and the data, the rest are added by the CAN controller. In order to have an explicit view of the CAN bus advantages over RS-485, Table 3.2. is illustrated as below.

Table 3.2: Features of communication protocols

Characteristic	CAN	RS-485
Interface	CAN controller	USART
Bus arbitration principle	Message identifier	Null
Data capacity	MAX 8 bytes	Unlimited
Max transmission rate	1 Mbit/s (up to 50 m)	10 Mbit/s (up to 12 m)
Error detection	Hardware implementation	Software implementation
Supply voltage	3.3V	5V

The implementation of CAN bus communication on the driver unit includes the initialization of the relative clock, the selection of CAN bus working mode (normal mode and loopback

mode, etc.), baud rate setup and the configuration of ID filter. During data transmission, a transmitter broadcasts its messages to all the receivers. On the message reception side, every receiver node can decide whether the message is needed or not, depending on the message ID [65]. This is achieved by the configurable and scalable filter banks that are provided by the CAN controller. It can be configured into two modes, identifier list or identifier mask, respectively. The identifier list mode is used to select a single identifier. The message can be received only when its ID is the same as the filter. Another mode is use to screen a group of identifiers as shown in Fig 3.20. In this case, a self-configured transmitted ID is compared with a self-

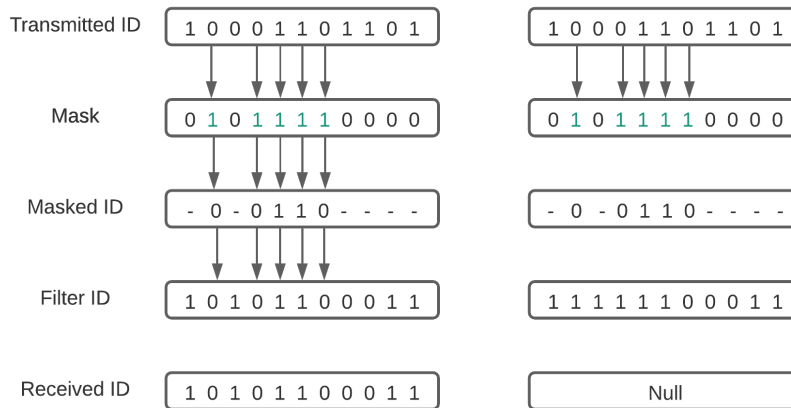


Figure 3.20: Identifier mask mode.

configured mask. 1 means must match, 0 means don't care. The result (masked ID) is used as a comparator to compare the other received messages ID (Filter ID). If all the five bits (labelled with green in Fig. 3.20) match with the received message ID, the message can be received, otherwise the message is discarded.

As mentioned earlier, a CAN data frame is capable of transmitting a maximum of 8 bytes of data. In order to achieve between-joint data communication. A data protocol needs to be preset for the convenience of data transmission in the communication network as shown below in Fig 3.21. The slave driver units link on the CAN bus are able to communicate with the main driver unit and perform actions such as real-time data transmitting, robot status monitoring and command executing, etc. The CAN bus arbitration technique as mentioned before avoids message

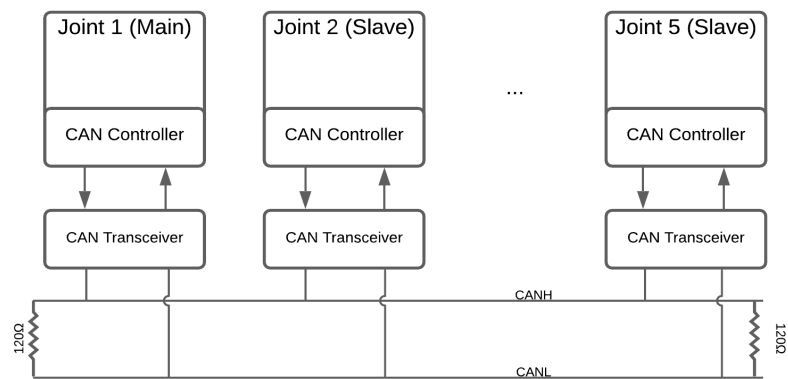


Figure 3.21: CAN bus communication network.

collision and achieves point-to-point message delivery. The self-defined CAN bus data frame is shown in Fig 3.22. The upper computer (PC) is responsible for transmitting the commands

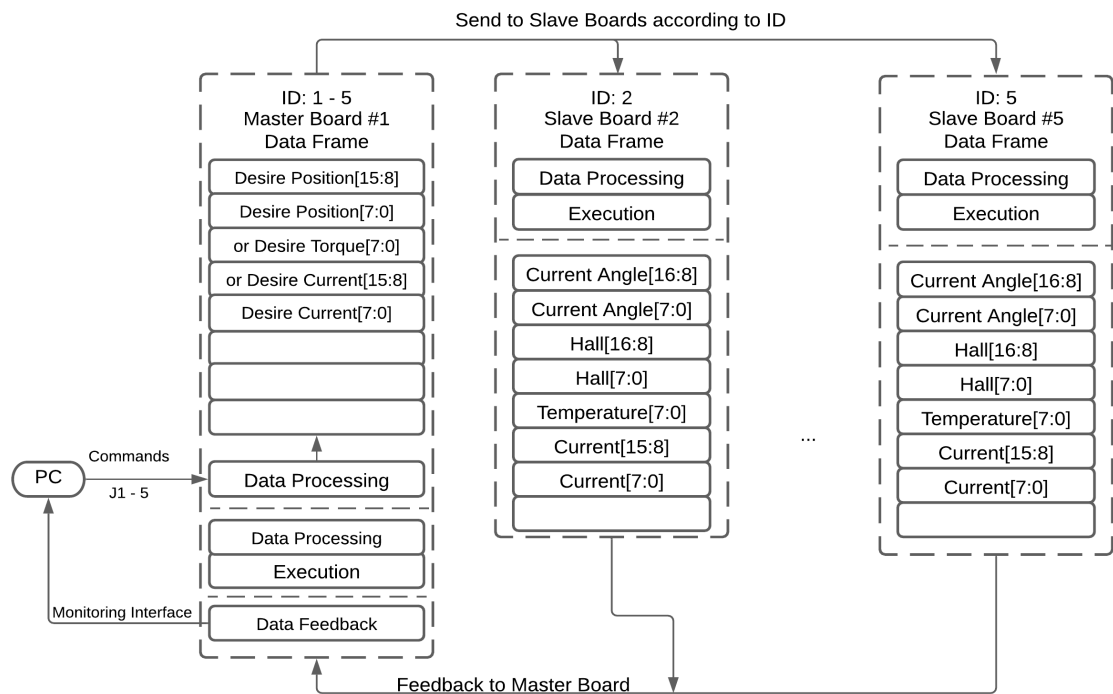


Figure 3.22: CAN bus data frame.

to the master driver unit. The master driver unit will immediately process the commands and

send out the messages which are marked with ID numbers. Meanwhile, the slave driver units will filter the messages and receive the message with the matched ID number. Upon completion of receiving, the data is fed to the slave controller to execute the corresponding actions. At the same time, the current statuses of the joint including current angle, magnetic field density, coil temperature and current are sent from the slave driver units to the master driver unit. This information is then sent to the upper computer to update the monitoring interface.

Position Control

The decentralized control can be achieved through implementing low-level control algorithms in the driver units that are distributed into every joint. The low-level control network they construct allows us to control the robot with high-level commands. Robot low-level control includes joint position control and joint torque control, etc. This section explains the joint position control algorithm. Since the model of the robot is not yet constructed, we treat the system as an unknown black box and use a PD/PID controller as a suitable control solution. In our robot position control loop, this PID controller uses the feedback signal from the encoder located at the joint to control the joint. The mathematical form of the PID controller can be represented as,

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}, \quad (3.1)$$

where K_p is the proportional gain, K_i is the integral gain, K_d is the derivative gain, $e(t)$ is the error between the desire value and the actual value, t is the time, τ is the variable of integration which takes from the initial time to the present time. However, this form cannot be implemented in the microprocessor unless discretize the PID controller. The integral term and the derivative term can be replaced with,

$$\int_0^t e(\tau) d\tau \approx T \sum_{n=1}^k e_n, \quad (3.2)$$

$$\frac{de(t)}{dt} \approx \frac{e_k - e_{k-1}}{T}, \quad (3.3)$$

where T is the sampling period, and the error at the k -th moment of sampling can be represented as $e(t) \approx e_k$. Subsequently, we have,

$$u_k = K_p \cdot e_k + K_i \cdot T \sum_{n=1}^k e_n + \frac{K_d \cdot (e_k - e_{k-1})}{T}. \quad (3.4)$$

There are several practical issues regarding the implementation of the PID controller on the robot. The first is the selection of the sampling period T . The T should be small enough compared with the system dynamics. A rather large sampling period will make the system unstable. On the other hand, a rather small sampling period will increase the system overhead. Therefore, the sampling period is selected to be 10 ms according to the actual system performance. The second problem is the integral windup. For a control system with a wide range of operating conditions, and large and fast changing desired values, such as the bidirectional actuation of robot joint in this work, there will be large and frequent changes in the set-points. We also require the robot to respond to and to finish the position tracking commands at millisecond level. These can cause the integral term to accumulate spurious error value and lead to the control saturation. To overcome the problem, it is necessary for the control error to have an opposite sign for a long period before things return to normal [6]. In practical experiments, the robot joint will have unexpected overshoot while switching directions. Hence, the anti-windup technique is introduced to eliminate the summation of errors in the integral term. This algorithm is called incremental/velocity PID algorithm, its output is redefined as an incremental control value $\Delta u_k = u_k - u_{k-1}$. It can be represented as,

$$\Delta u_k = K_p \cdot (e_k - e_{k-1}) + K_i \cdot T \cdot e_k + \frac{K_d \cdot (e_k - 2e_{k-1} + e_{k-2})}{T}. \quad (3.5)$$

Implementing the incremental PID algorithm can limit the position tracking overshoot within a controllable and acceptable range. The incremental PID algorithm calculates the difference between the current and the last tracking errors. Therefore, the system noise can be cancelled out, and derive a better result.

After clarifying the PID algorithm, we need to implement the algorithm on the driver unit. The inappropriate implementation can result in the inaccurate PID sampling period. This can either make the system unstable or cannot meet the control requirements. According to the practical performance, the position PID algorithm is implemented in a 100 Hz real-time task. It

is configured as the second highest priority task to make sure the execution of position control is not delayed. Finally, the output of the PID controller is fed to a timer's capture compare register to generate a time-varying PWM signal. It provides the input signals for H-bridge current drivers. Therefore, the clutches are able to output the regulated torque to control the position of the joint and switch the motion direction according to the sign of the output of PID controller.

Current Control

In this section, the open-loop torque control of the joint is achieved by the closed-loop current control algorithm. To achieve the current control scheme, the ADC signal is used as the feedback signal to construct the close loop control scheme using the incremental/velocity PID controller. This method can easily avoid integral windup and limit the rate of change of the control signal. The closed-loop control configuration is shown in Fig 3.23. The frequency of the current control task is set to 1 kHz, which is the maximum stable running frequency of the RTOS. The data transmission is achieved by the DMA to reduce the execution delay. To summarize, the implementation of current control provides a more reliable control scheme to respond to the fast changing currents caused by control actions. This method also allows us to perform the predictable and precise open-loop torque control. The experimental results will be shown in the next section.

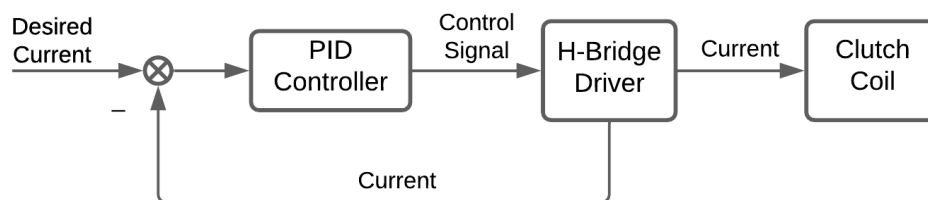


Figure 3.23: Current control scheme.

Trajectory Generation

Different from path planning which only provides a geometric description of the robot motion. Trajectory planning specifies the dynamic aspects of the motion, such as the joint velocities and joint accelerations while traversing the path [62]. In this work, a cubic polynomial is implemented in the MATLAB to generate the trajectory. Trajectory points are used as the desired position signals of the robot joint. Its start/end velocities, initial/final positions and time duration are specified according to the requirements. We will call these as motion parameters. The cubic polynomial $q(t)$ can be written as,

$$q(t) = a_0 + a_1 \cdot t + a_2 \cdot t^2 + a_3 \cdot t^3. \quad (3.6)$$

Differentiating the above equation, we can get,

$$\dot{q}(t) = a_1 + 2a_2 \cdot t + 3a_3 \cdot t^2, \quad (3.7)$$

where $\dot{q}(t)$ represents for the velocity of the joint at time t , and the a_x are unknown. Combine the (3.6) and (3.7), we can get,

$$q_0 = a_0 + a_1 \cdot t_0 + a_2 \cdot t_0^2 + a_3 \cdot t_0^3, \quad (3.8)$$

$$v_0 = a_1 + 2a_2 \cdot t_0 + 3a_3 \cdot t_0^2, \quad (3.9)$$

$$q_{final} = a_0 + a_1 \cdot t_{final} + a_2 \cdot t_{final}^2 + a_3 \cdot t_{final}^3, \quad (3.10)$$

$$v_{final} = a_1 + 2a_2 \cdot v_{final} + 3a_3 \cdot v_{final}^2, \quad (3.11)$$

where q_0 and v_0 represent for the position and velocity of the joint at initial time, q_{final} and v_{final} represent for the position and the velocity of the joint at final time. The four unknowns can be solved by substituting the motion parameters into equations (3.8 - 3.11). Fig. 3.24 (adopted from [61]) illustrates the trajectory, velocity and acceleration profiles of a cubic polynomial trajectory. The initial and final positions are 10 degree and -20 degree, the time duration is 1 second and the velocities at both initial and final points are zero deg/sec. The result shows that the cubic polynomial trajectory generator can generate a smooth trajectory with continuous velocity.

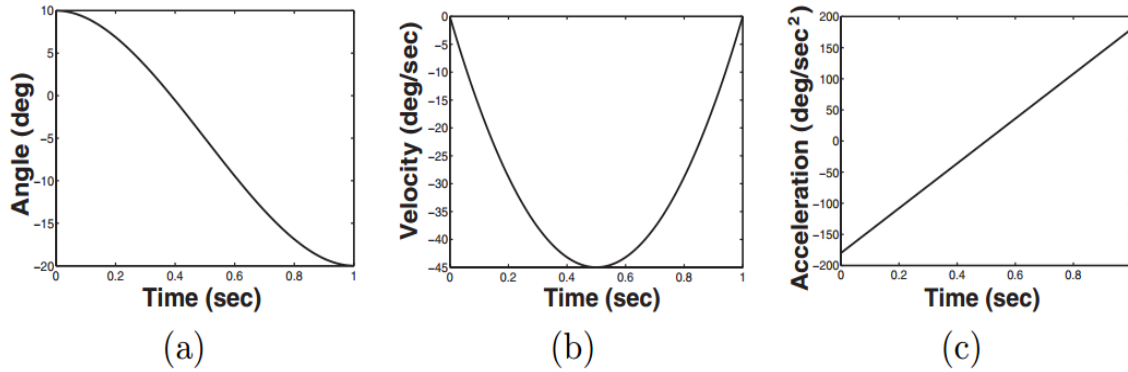


Figure 3.24: Cubic polynomial trajectory.

3.5 Experimental Results

This section conducts a set of experiments to evaluate the performance of the robot. So far, the first two joints of the robot were built, the tests were mainly conducted on the first two joints. The experiment setup is as follows, an electric motor (Q80-13XS F3A) provides the rotational input to a gear reducer. It actuates the gears and belts to transmit the power to input shaft/disks of the MR clutches (see Fig. 3.25, adopted from [45]). The RPM of the motor is maintained

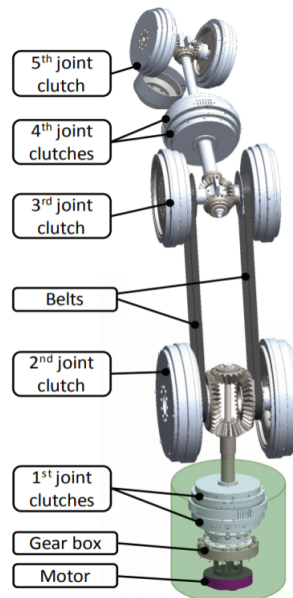


Figure 3.25: Transmission mechanism.

constant during tests.

3.5.1 Position Control Experiments

Sinusoidal Signal Tracking

In this set of experiments, sinusoidal signals were input to the joint two to test the performance of real-time control firmware and the MR actuator in the robot joint. The joint two is 0.4 m long covered by the aluminum housing. The motion range was 20 degrees. The position PID controller's gain values were selected by manual tuning. In the following experiments, the angle of joint two is 163 deg when the arm is perpendicular to the ground. Fig 3.26 to Fig 3.28 show the results of sinusoidal position tracking experiments that were conducted at 0.25 Hz,

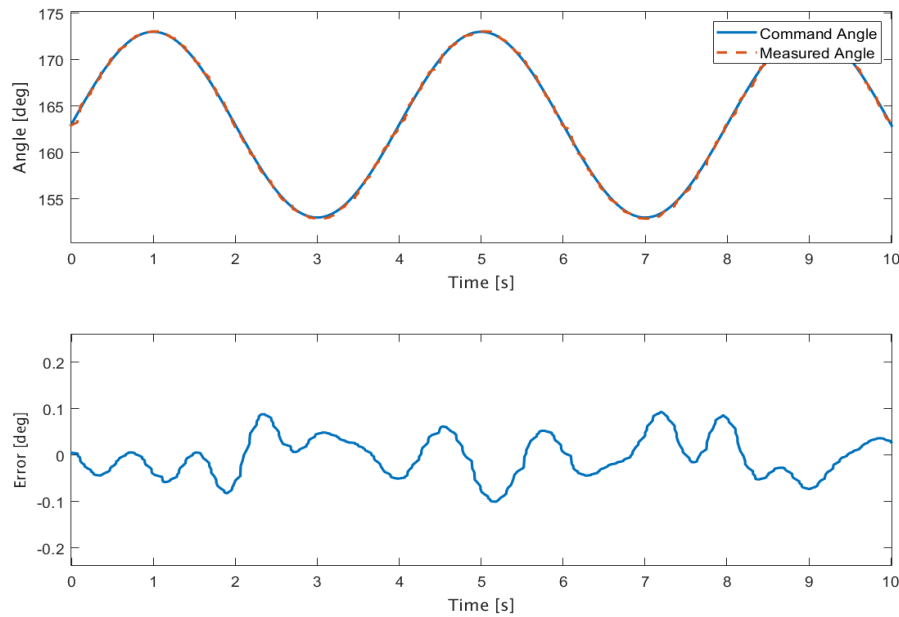


Figure 3.26: Joint 2 position tracking of a 0.25Hz sinusoidal reference signal.

0.5 Hz and 1 Hz. As can be seen in the results, the command and the measured angle has a good consistency. The following section conducts trajectory tracking experiments on both the first and the second joints.

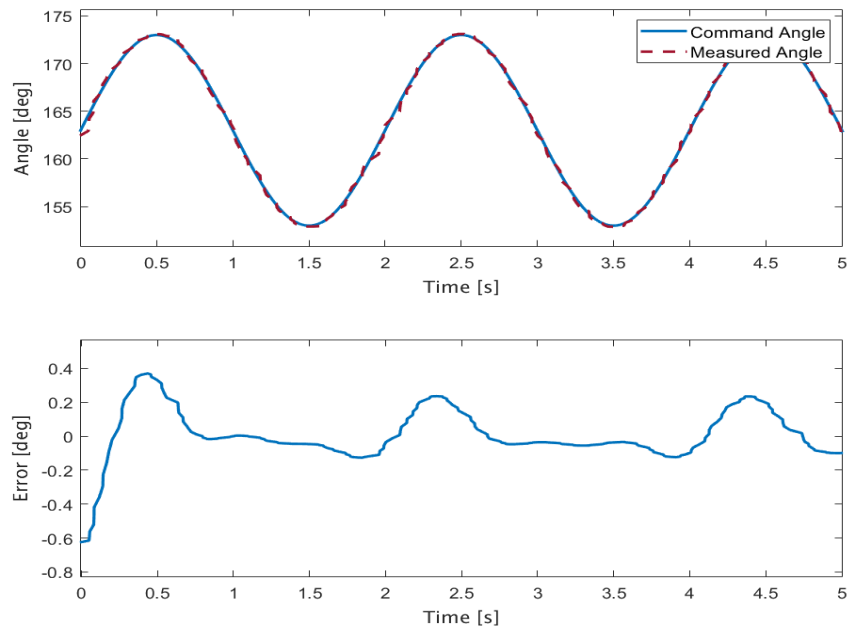


Figure 3.27: Joint 2 position tracking of a 0.5Hz sinusoidal reference signal.

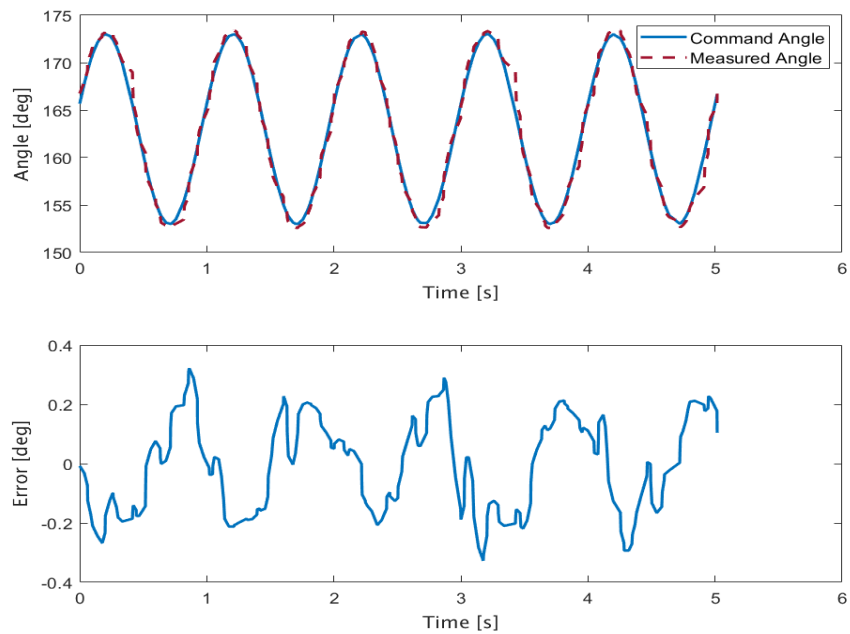


Figure 3.28: Joint 2 position tracking of a 1Hz sinusoidal reference signal.

Trajectory Tracking

The experiments were conducted on the first two joints to test the tracking precision. The angle of joint two is 163 degree when the joint is perpendicular to the ground and the motion range is 40 degrees. The tests were conducted in this motion range to further test the impact of the MR clutch switching to the joint motion. The motion range of the joint one was selected randomly because its axle of revolution is perpendicular to the ground. The speed of the motor was also kept constant during experiments. The results are illustrated in Fig 3.29 and 3.30. These

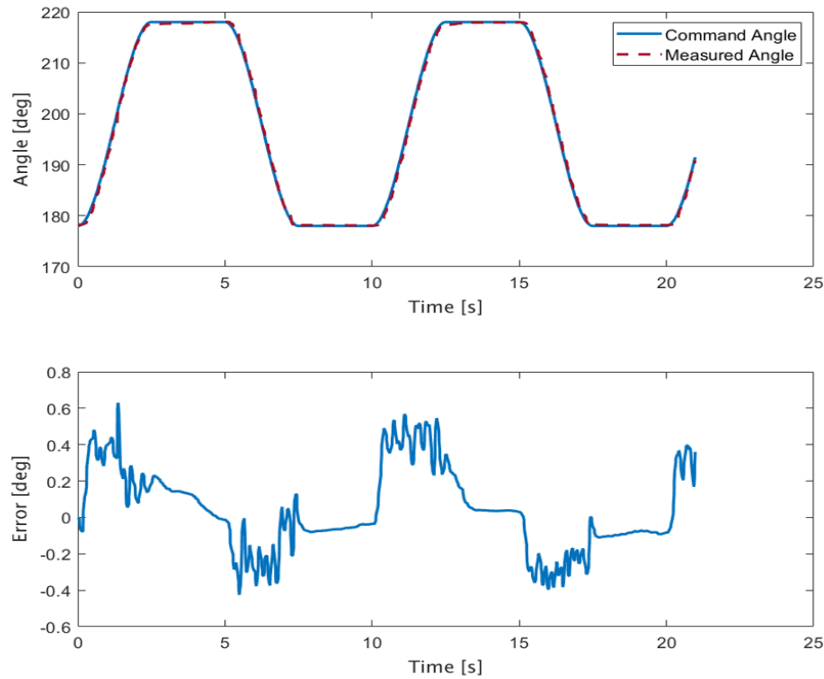


Figure 3.29: Joint 1 trajectory tracking.

results indicate the MR clutches in antagonistic arrangement can achieve favorable precision in position control tasks. The functionalities of the MR actuators and the real-time firmware can also meet the design requirements.

The joint debugging of the first two joints was also conducted to verify the functionality of the robot control network. The joint one driver unit was used as the master unit. It sent position commands to the joint 2 and controlled itself simultaneously. Therefore, both joints can perform trajectory tracking at the same time. The results in Fig. 3.31 and 3.32 demonstrate that the robot control network can perform predictable and stable control actions to all of the

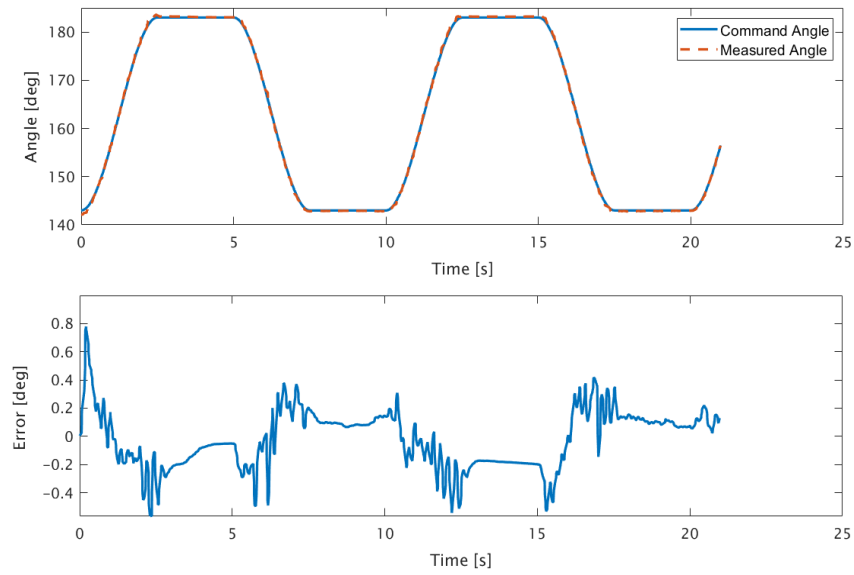


Figure 3.30: Joint 2 trajectory tracking.

linked joints. Each individual joint also achieves favourable precision in position control tasks.

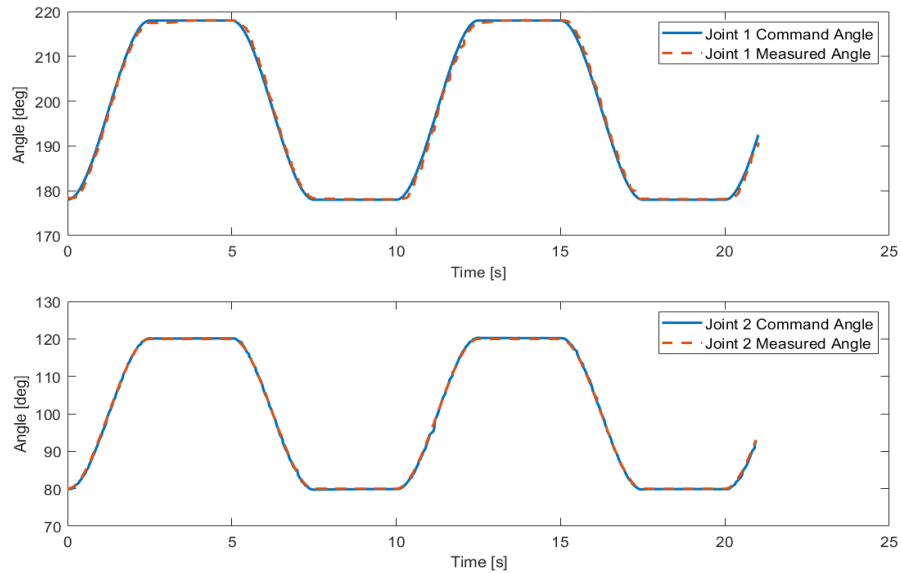


Figure 3.31: Joint debugging of Joint 1 and Joint 2 trajectory tracking.

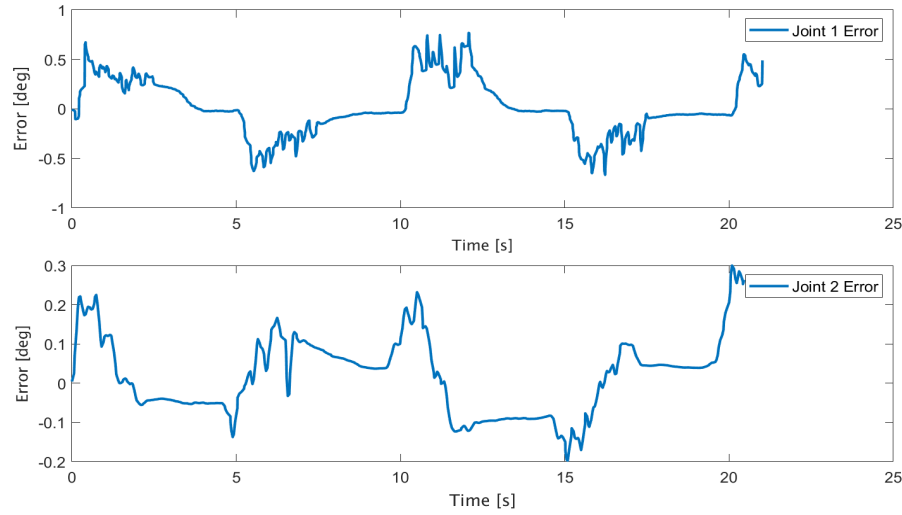


Figure 3.32: Error signal of joint debugging of Joint 1 and Joint 2 trajectory tracking.

3.5.2 Open-loop Torque Experiments

In this set of experiments, the 6-axis torque/force transducer (ATI Gamma SI-15-50) was attached to the end of joint 2, and the joint two was locked. The experiment setup is shown in Fig 3.33.

The output of the torque transducer was received and processed by the MATLAB through a RS232/USB converter. Its value reflected the output torque during current control. The gain values were manually tuned to achieve the best performance. Fig 3.34 shows the block diagram of the system used to obtain the experimental results present in this section.

Step response

In order to evaluate the transient performance of the joint 2 clutch and current PID controller. The output torque and current of the clutch in response to a step current signal were recorded. The output torque was measured through the stationary torque transducer when the input side

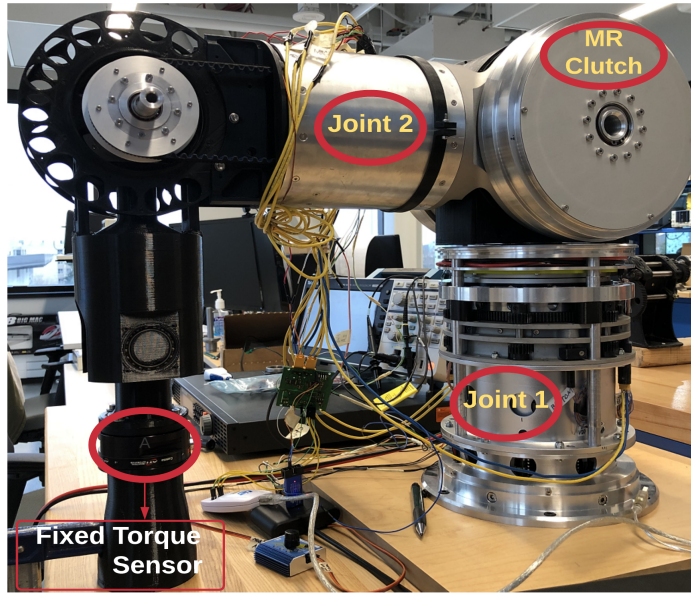


Figure 3.33: Experiment setup.

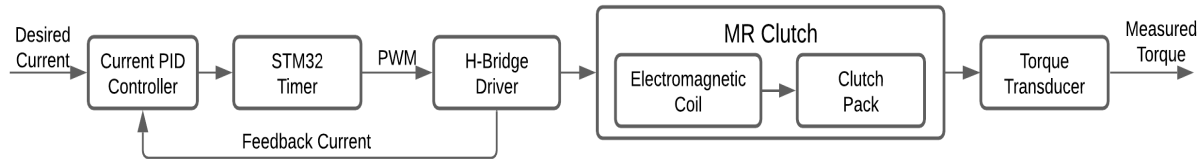


Figure 3.34: Diagram of the current control experiments.

of the clutch was actuated by a constant speed motor (see Fig. 3.33). In the experiment, the desire current of the current PID controller was set to 0.4 A. The result is shown in Fig 3.35.

The output torque has a relatively short rise time of around 100 milliseconds. It mainly results from the inductance of the electromagnetic coil, the response time of magneto-rheological fluids (around 20 ms). Also partly due to the delay imposed by electronics and software delay, as well as the time that is used by the magnetic field to propagated to the clutch pack [58].

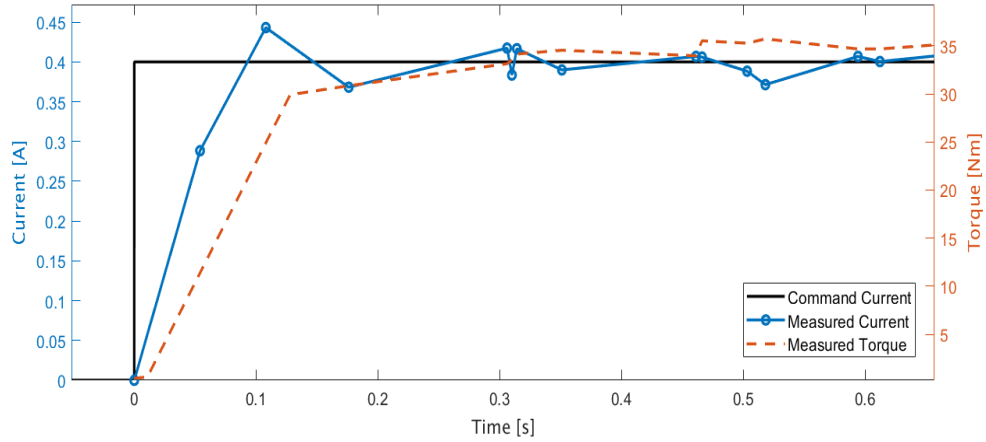


Figure 3.35: Transient performance of the joint 2 MR clutch when subjects to 0 A to 0.4 A input current.

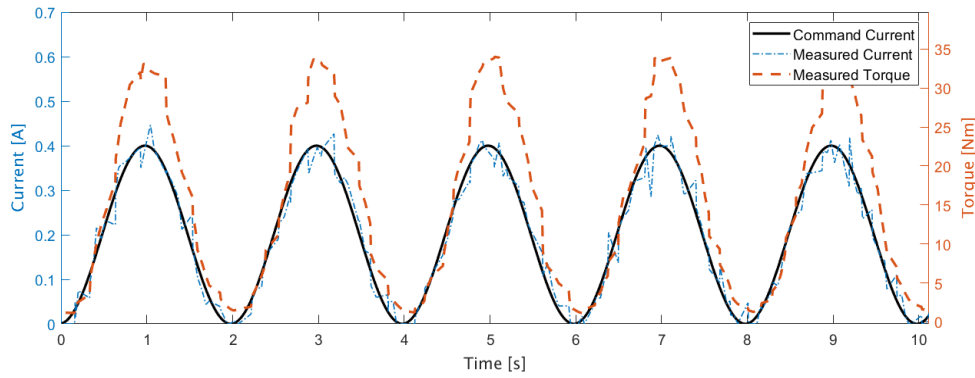


Figure 3.36: Current tracking of a 0.5 Hz sinusoidal reference signal.

Controllability of Output Torque

In order to further test the performance of the driver unit and evaluate the predictability of the output torque. Sinusoidal input current signals were input to the clutch. Fig 3.36 and Fig 3.37 present the results when the output currents were commanded to follow sinusoidal currents. At both 0.5 Hz and 1 Hz, the actual output currents overlay that of the commands. In addition, the output torque signals also retain the sinusoidal signal quite well. The distort of measured torque signal mainly results from the poor accuracy and low sampling rate of the torque transducer. To conclude, the predictable torque outputs are an asset to the development of closed-loop torque control scheme in the future.

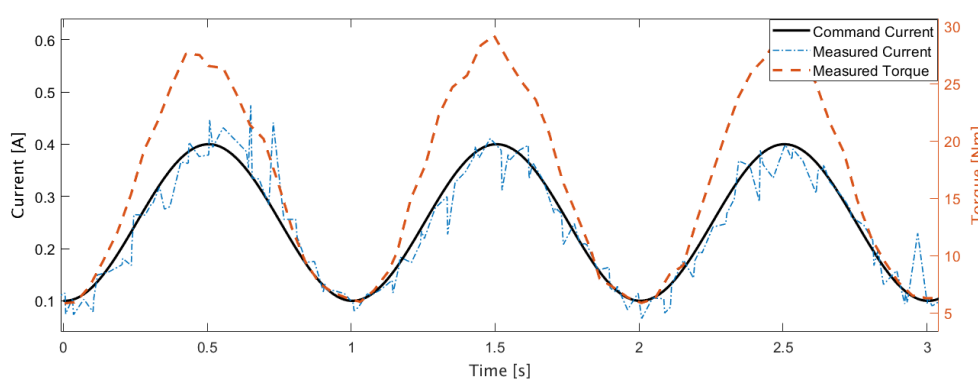


Figure 3.37: Current tracking of a 1 Hz sinusoidal reference signal.

3.6 Conclusions

In this chapter, the development of a real-time firmware based on the STM32F4 processor was presented. This firmware provides a predictable and reliable computing environment for the robot joints to achieve design requirements. The clock systems of the host system and the RTOS were first analyzed. The real-time tasks, start-up process and task scheduling algorithms were then detailed. The system architecture was illustrated. All of the user tasks and their methods of implementation were analyzed thoroughly. A series of experiments were conducted to evaluate the performance of the system. The position tracking and trajectory following results are persuasive, which verify the functionalities of the real-time software, electronic driver, MR actuators and the robot. The results of the joint debugging tests also exhibit excellent controllability of the robot control network. Finally, the current control results demonstrate that the system is capable of providing high-quality current control. It lays the foundation for the implementation of the closed-loop torque control.

Chapter 4

Conclusion and Future Works

4.1 Conclusion

In this thesis, a customized electronic driver aiming at constructing a decentralized control network for a 5-DOF MR clutch based collaborative robot was designed, developed, and evaluated. The electronic driver features a miniaturized design. This allows it to be embedded into the housing of the robot joint. Meanwhile, each electronic driver is responsible for processing sensors' data, driving the MR clutches, executing control commands and communicating with other joints. These functionalities cannot be achieved using commercially available driver units without sacrificing performance and/or size. Each driver unit includes an STM32F4 processor to receive feedback signals from all sensors (Hall Sensor, Encoder, Current sensor) to control the antagonistic clutch pair in a closed-loop feedback. A series of experiments were performed to evaluate the performance of the voltage conversion system on the driver unit. The results are persuasive. The results of the thermal stability tests as well show durability in both nominal and extreme conditions.

A real-time firmware was constructed using $\mu\text{C}/\text{OS-II}$, a commercially available RTOS with the preemptive kernel. It provides a predictable and reliable computing environment for the robot to exchange sensors' data and implement control algorithms. Then, the real-time program was assessed in various scenarios to evaluate the performance of the developed software/hardware systems and the robot. The results have shown that both the constructed robot control network and the firmware of individual driver unit are predictable and stable.

4.2 Future Works

Construction of MR Clutch Model and Closed-loop Torque Control

Compare with common electric actuators that feature a linear relationship between the input current and output torque, the MR actuator, otherwise, has a nonlinear behaviour because of the magnetic hysteresis (see Figs 4.1 and 4.2).

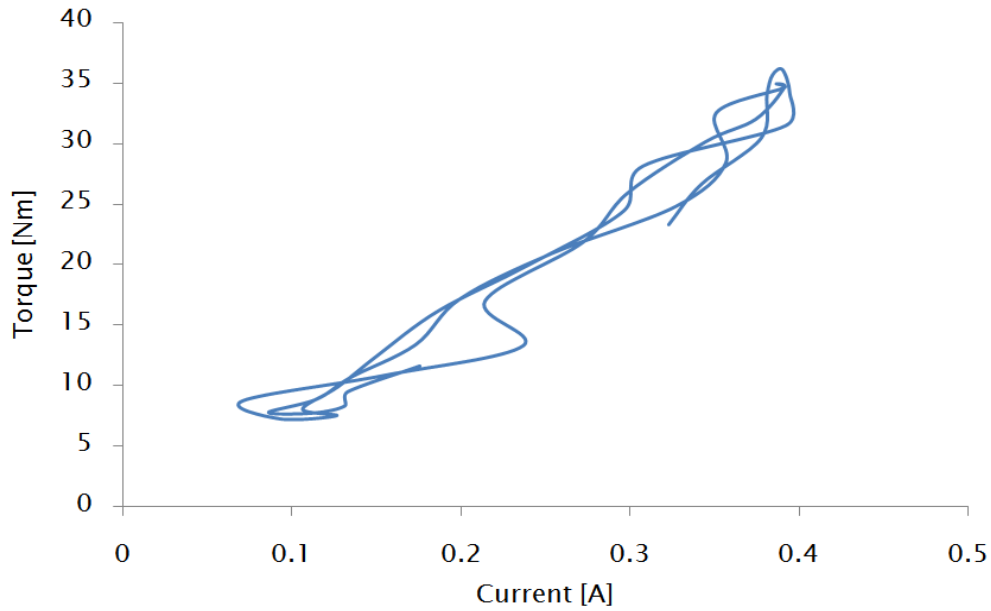


Figure 4.1: Hysteresis behaviour of the MR clutch when applying a 1 Hz sinusoidal reference signal.

The figures above clearly show that the magnetic hysteresis has a rate-dependent non-linear behaviour. The non-linearity is mainly associated with the ferromagnetic components, such as the electromagnetic coil, shaft, and disks. In practice, the rate-dependent hysteresis behaviour makes the MR actuator hard to control. Therefore, the non-linear behaviour should be compensated to achieve better performance. Given that the relationship between MR fluids' shear stress and the magnetic flux density is one-to-one (see Fig. 1.3). A prospective method is to estimate the output torque by input the magnetic field feedback signals to the MR clutch model. Therefore, an MR clutch model needs to be constructed to help estimate the output torque. If the model is sufficiently accurate, we can estimate the output torque without using torque

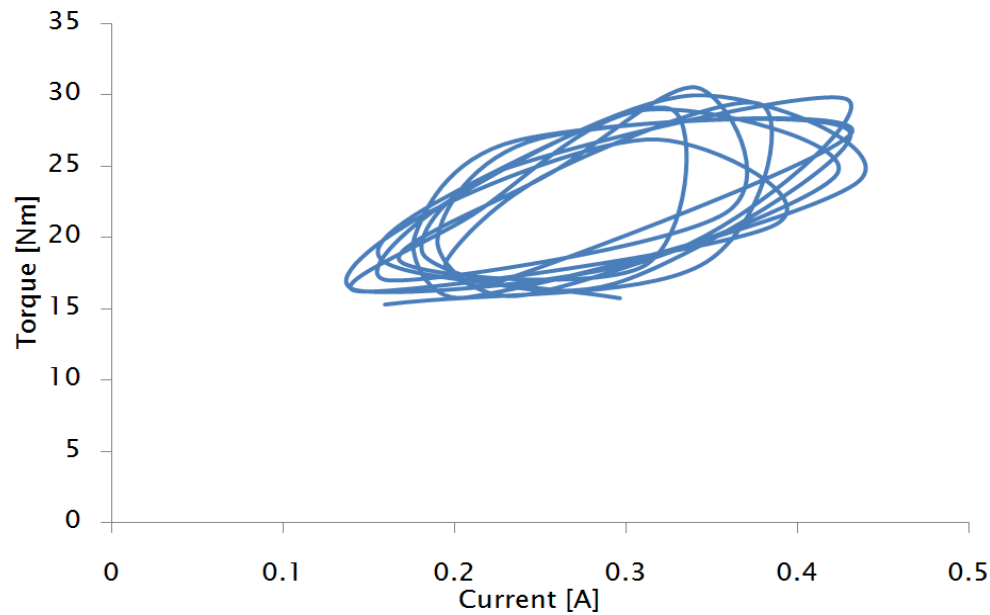


Figure 4.2: Hysteresis behaviour of the MR clutch when applying a 5 Hz sinusoidal reference signal.

sensors, which either save the budget or eliminate measurement noise. Additionally, from the experimental results of the chapter three, we can clearly see that the measured output torque follows the shape of the input current. This is an asset to perform closed-loop torque control in the future. The prospective closed-loop torque control is shown in Fig. 4.3.

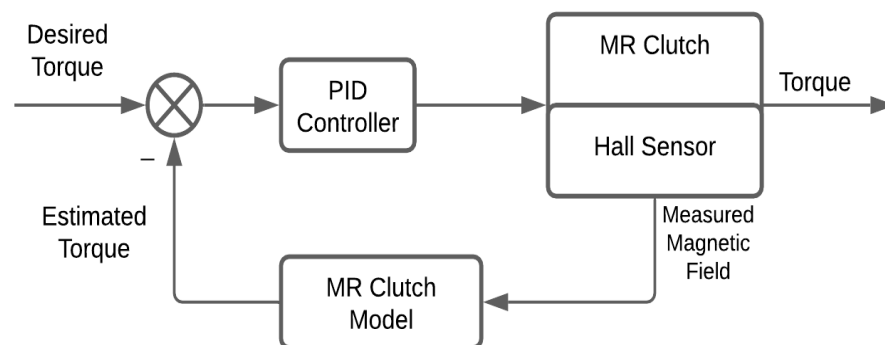


Figure 4.3: Proposed control scheme of the torque control.

Overall Control of the 5-DOF Robot

As previously mentioned, the first two joints were manufactured, assembled and tested at the current stage. However, the final goal is to achieve the overall control of the 5-DOF MR clutch based collaborative robot actuated by a unidirectional motor. This robot is developed as a prototype intended to demonstrate the capacity of the A-DASA approach in executing collaborative activities between the environment (including humans) and the robot. To achieve this goal, the basic control algorithms such as the joint torque control and the joint position control need to be implemented, tuned and practically tested on each joint of the robot. In addition to the individual tests of the joints, joint debugging is the most significant procedure. While joint debugging, the constructed robot control network will process a massive amount of data process in real-time and perform accurate control actions on each joint. Thus, stability and processing speed will be the two key-points in implementing the overall control of the robot.

Collaborative Tasks

Nowadays, new applications such as robots in hospitals and manufacturing companies require robots to work collaboratively with people in human-robot teams. As an emerging area, Human-Robot Collaboration (HRC) is comprised of classical robotics, human-robot interaction, machine learning and mechanical design, etc. In order to perform collaborative activities on our robot, more aspects should be learnt to achieve favourable results.

Learn to Collaborate from Human Demonstrations

In the field of intelligent manufacturing, collaborative robots are widely employed in assembly tasks. The collaborative robots need to first learn from human demonstrations. Then the robots can assist humans in shared working situations. Machine learning algorithms can be used to train the robots and update their knowledge. This is a prospective field that deserves to continue to study.

Bibliography

- [1] *Materials library in FEMM 4.0*, 2 2015. Steel magnetic properties.
- [2] J. Aboudi, M. Pindera, and S. Arnold. Higher-order theory for functionally graded materials. *Composites Part B-engineering*, 30:777–832, 1999.
- [3] Advanced Monolithic Systems. *1A LOW DROPOUT VOLTAGE REGULATOR*.
- [4] ALLEGRO Microsystems. *Using the SENT Communications Output Protocol with A1341 and A1343 Devices*, 10 2019. Rev 3.
- [5] Analog Devices. *Decoupling Techniques*, 2009.
- [6] Karl Johan Åström. Control system design lecture notes for me 155a. *Department of Mechanical and Environmental Engineering University of California Santa Barbara*, 333, 2002.
- [7] W. Atmadja, B. Christian, and L. Kristofel. Real time operating system on embedded linux with ultrasonic sensor for mobile robot. In *2014 International Conference on Industrial Automation, Information and Communications Technology*, pages 22–25, 2014.
- [8] More Avraam, Mihaita Horodincu, Pierre Letier, and André Preumont. Portable smart wrist rehabilitation device driven by rotational mr-fluid brake actuator for telemedicine applications. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1441–1446. IEEE, 2008.
- [9] H Bose, T Gerlach, and J Ehrlich. Magnetorheological torque transmission devices with permanent magnets. In *J. Phys: Conf. Series*, volume 412, page 012050, 2013.

- [10] Cheol Chang, Myung Jin Chung, and Bum Hee Lee. Collision avoidance of two general robot manipulators by minimum delay time. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(3):517–522, 1994.
- [11] Chee-Meng Chew, Geok-Soon Hong, and Wei Zhou. Series damper actuator: a novel force/torque control actuator. In *4th IEEE/RAS International Conference on Humanoid Robots, 2004.*, volume 2, pages 533–546 Vol. 2, 2004.
- [12] J. Chen and W. Liao. A leg exoskeleton utilizing a magnetorheological actuator. In *2006 IEEE International Conference on Robotics and Biomimetics*, pages 824–829, 2006.
- [13] J. Chen and W. Liao. Experimental evaluation of an assistive knee brace with magnetorheological actuator. In *2010 IEEE International Conference on Robotics and Biomimetics*, pages 1238–1243, 2010.
- [14] G. Cheng, L. Xu, S. Chen, L. Dong, H. Xu, and J. Liu. Design and control of a sitting/lying style lower limb rehabilitation robot with magnetorheological actuators. In *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pages 1–6, 2019.
- [15] Chee-Meng Chew, Geok-Soon Hong, and Wei Zhou. Series damper actuator: a novel force/torque control actuator. In *4th IEEE/RAS International Conference on Humanoid Robots, 2004.*, volume 2, pages 533–546. IEEE, 2004.
- [16] John P. Coulter, Keith D. Weiss, and J.David Carlson. Engineering applications of electrorheological materials. *Journal of Intelligent Material Systems and Structures*, 4(2):248–259, 4 1993.
- [17] Michele Dassisti and Giovanna Brunetti. *Introduction to Magnetorheological Fluids*. 01 2020.
- [18] Fabrizio Flacco, Torsten Kröger, Alessandro De Luca, and Oussama Khatib. A depth space approach to human-robot collision avoidance. In *2012 IEEE International Conference on Robotics and Automation*, pages 338–345. IEEE, 2012.

- [19] MRF-140CG Magneto-Rheological Fluid. Lord corporation. 2019.
- [20] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, 1997.
- [21] Pranav Gadekar, V.S. Kanthale, and N.D. Khaire. Magnetorheological fluid and its applications. *International Journal of Current Engineering and Technology*, 3 2017.
- [22] H. Guo and W. Liao. Optimization of a multifunctional actuator utilizing magnetorheological fluids. In *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 67–72, 2011.
- [23] Croft E. Haddadin S. *Physical Human–Robot Interaction*. In: *Siciliano B., Khatib O. (eds) Springer Handbook of Robotics*. Springer Handbooks. Springer, Cham., 2016.
- [24] IDT System Architect. *A Tutorial for the Digital SENT Interface*, 2014.
- [25] Infineon. *TLE4998S4 Programmable Linear Hall Sensor*, 7 2008. Rev 1.
- [26] Julie Jackson, Mark Messner, Nikola Dudukovic, William Smith, Logan Bekker, Bryan Moran, Alexandra Golobic, Andrew Pascall, Eric Duoss, Kenneth Loh, and Christopher Spadaccini. Field responsive mechanical metamaterials. *Science Advances*, 4:eaau6419, 12 2018.
- [27] Nils Karlsson. Theory and application of a capacitive sensor for safeguarding in industry. In *Conference Proceedings. 10th Anniversary. IMTC/94. Advanced Technologies in I & M. 1994 IEEE Instrumentation and Measurement Technolgy Conference (Cat. No. 94CH3424-9)*, pages 479–482. IEEE, 1994.
- [28] Amir Khajepour, Saeed Behzadipour, Robert Dekker, and Edmon Chan. Light weight parallel manipulators using active/passive cables, February 6 2007. US Patent 7,172,385.
- [29] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous robot vehicles*, pages 396–404. Springer, 1986.

- [30] C. Khazoom, P. Caillouette, A. Girard, and J. Plante. A supernumerary robotic leg powered by magnetorheological actuators to assist human locomotion. *IEEE Robotics and Automation Letters*, 5(4):5143–5150, 2020.
- [31] C. Khazoom, C. Véronneau, J. L. Bigué, J. Grenier, A. Girard, and J. Plante. Design and control of a multifunctional ankle exoskeleton powered by magnetorheological actuators to assist walking, jumping, and landing. *IEEE Robotics and Automation Letters*, 4(3):3083–3090, 2019.
- [32] KUKA-LWR-Brochure. Kuka corp. 2019.
- [33] Jean Labrosse. *MicroC/OS-II: The Real Time Kernel*. CRC Press, 2002.
- [34] W. Li, P. Yadmellat, and M. R. Kermani. Linearized torque actuation using fpga-controlled magnetorheological actuators. *IEEE/ASME Transactions on Mechatronics*, 20(2):696–704, 2015.
- [35] Wenjun Li, Peyman Yadmellat, and Mehrdad R Kermani. Design optimization and comparison of magneto-rheological actuators. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5050–5055. IEEE, 2014.
- [36] Modesto T Lopez-Lopez, Juan DG Duran, and Andrey Y Zubarev. Rheological analysis of magnetorheological fluids. 2019.
- [37] Modesto López-López, Juan Duran, and Andrey Zubarev. *Rheological Analysis of Magnetorheological Fluids*. 01 2019.
- [38] Maxim Integrated. *CAN vs. RS-485: Why CAN Is on the Move*.
- [39] MPS. *MP4560 2A, 2MHz, 55V Step-Down Converter*, 2 2014. Rev.1.01.
- [40] Francesca Negrello, Manolo Garabini, Manuel G Catalano, Jörn Malzahn, Darwin G Caldwell, Antonio Bicchi, and Nikolaos G Tsagarakis. A modular compliant actuator for emerging high performance and fall-resilient humanoids. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 414–420. IEEE, 2015.

- [41] Olesya Ogorodnikova. Methodology of safety for a human robot interaction designing stage. In *2008 Conference on Human System Interactions*, pages 452–457. IEEE, 2008.
- [42] Abdul Ghani Olabi and Artur Grunwald. Design and application of magneto-rheological fluid. *Olabi, Abdul-Ghani and Grunwald, Artur (2007) Design and application of magneto-rheological fluid. Materials Design, 28 (10). pp. 2658-2664. ISSN 0261-3069, 28, 12 2007.*
- [43] S. Pisetskiy and M. R. Kermani. Design and simulation of the first joint for a prospective 5 degrees of freedom manipulator with mr actuators. In *Proceedings of The Canadian Society for Mechanical Engineering International Congress 2018*, 2018.
- [44] S. Pisetskiy and M. R. Kermani. Development of mr clutch for a prospective 5 dof robot* this work was supported in part by canada foundation for innovation (cfi) and natural sciences and engineering research council (nserc) of canada under grant no.25031 and rgpin-346166. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5900–5905, 2018.
- [45] Sergey Pisetskiy and Mehrdad R Kermani. Compliant 5-dof robot: Mechanical design.
- [46] G. A. Pratt and M. M. Williamson. Series elastic actuators. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, volume 1, pages 399–406 vol.1, 1995.
- [47] Gill A Pratt and Matthew M Williamson. Series elastic actuators. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, volume 1, pages 399–406. IEEE, 1995.
- [48] Gill A Pratt and Matthew M Williamson. Elastic actuator for precise force control, July 22 1997. US Patent 5,650,704.
- [49] PROYEK. *Understanding TVS Clamping*.
- [50] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

- [51] Morgan Quigley, Brian Gerkey, and William D Smart. *Programming Robots with ROS: a practical introduction to the Robot Operating System.* ” O’Reilly Media, Inc.”, 2015.
- [52] J. Rabinow. The magnetic fluid clutch. *Electrical Engineering*, 67(12):1167–1167, 1948.
- [53] SAE Mobilus. *SENT - Single Edge Nibble Transmission for Automotive Applications*, 4 2007.
- [54] G. Savioz, V. Ruchet, and Y. Perriard. Study of a miniature magnetorheological fluid actuator for haptic devices. In *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1197–1202, 2010.
- [55] A. S. Shafer and M. R. Kermani. Design and validation of a magneto-rheological clutch for practical control applications in human-friendly manipulation. In *2011 IEEE International Conference on Robotics and Automation*, pages 4266–4271, 2011.
- [56] A. S. Shafer and M. R. Kermani. On the feasibility and suitability of mr fluid clutches in human-friendly manipulators. *IEEE/ASME Transactions on Mechatronics*, 16(6):1073–1082, 2011.
- [57] A. S. Shafer and M. R. Kermani. Development of high performance intrinsically safe 3-dof robot. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 619–624, 2014.
- [58] Alex S Shafer and Mehrdad R Kermani. On the feasibility and suitability of mr fluid clutches in human-friendly manipulators. *IEEE/ASME Transactions on Mechatronics*, 16(6):1073–1082, 2010.
- [59] Alex S Shafer and Mehrdad R Kermani. Design and validation of a magneto-rheological clutch for practical control applications in human-friendly manipulation. In *2011 IEEE International Conference on Robotics and Automation*, pages 4266–4271. IEEE, 2011.
- [60] Bruno Siciliano and Wayne J Book. A singular perturbation approach to control of lightweight flexible manipulators. *The International Journal of Robotics Research*, 7(4):79–90, 1988.

- [61] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*. John Wiley & Sons, 2020.
- [62] Mark W Spong, Seth Hutchinson, Mathukumalli Vidyasagar, et al. *Robot modeling and control*. 2006.
- [63] STMicroelectronics. *SENT demo driver for the SPC57xx/SPC58xx microcontroller families*, 6 2016. Rev 2.
- [64] STMicroelectronics. *Getting started with STM32F4xxxx MCU hardware development*, 2018.
- [65] STMicroelectronics. *STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 advanced Arm-based 32-bit MCUs*, 2 2019. Rev 18.
- [66] STMicroelectronics. *STM32F407VE*, 8 2020. Rev 9.
- [67] Alejandro Suárez, P Sanchez-Cuevas, Manuel Fernandez, Manuel Perez, Guillermo Heredia, and Anibal Ollero. Lightweight and compliant long reach aerial manipulator for inspection operations. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6746–6752. IEEE, 2018.
- [68] N. Takesue, J. Furusho, and Y. Kiyota. Analytic and experimental study on fast response mr-fluid actuator. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 1, pages 202–207 vol.1, 2003.
- [69] Texas Instruments. *Input and Output Capacitor Selection*, 2 2006.
- [70] Texas Instruments. *AN-2020 Thermal Design By Insight, Not Hindsight*, 4 2013. Rev. 2.
- [71] Texas Instruments. *DRV8874 H-Bridge Motor Driver With Integrated Current Sense and Regulation*, 12 2019. Rev 2.
- [72] Texas Instruments. *How to select a Surge Diode*, 1 2019.
- [73] Mohd Hishamuddin Unuh, Pauziah Muhamad, Nur Waziralilah, Mohamad Amran, and Penerbit Akademia Baru. Characterization of vehicle smart fluid using gas

- chromatography-mass spectrometry (gcms). *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences*, 55:240–248, 03 2019.
- [74] PB Usoro, R Nadira, and SS Mahil. Control of lightweight flexible manipulators: a feasibility study. In *American control conference*, number 21, pages 1209–1216, 1984.
- [75] PB Usoro, R Nadira, and SS Mahil. A finite element/lagrange approach to modeling lightweight flexible manipulators. 1986.
- [76] H. Vallery, J. Veneman, E. van Asseldonk, R. Ekkelenkamp, M. Buss, and H. van Der Kooij. Compliant actuation of rehabilitation robots. *IEEE Robotics Automation Magazine*, 15(3):60–69, 2008.
- [77] Bram Vanderborght, Alin Albu-Schäffer, Antonio Bicchi, Etienne Burdet, Darwin G Caldwell, Raffaella Carloni, MG Catalano, Oliver Eiberger, Werner Friedl, Ganesh Ganesh, et al. Variable impedance actuators: A review. *Robotics and autonomous systems*, 61(12):1601–1614, 2013.
- [78] Bram Vanderborght, Alin Albu-Schäffer, Antonio Bicchi, Etienne Burdet, D.G. Caldwell, Raffaella Carloni, M. Catalano, O. Eiberger, Werner Friedl, Gowrishankar Ganesh, Manolo Garabini, Markus Grebenstein, Giorgio Grioli, S. Haddadin, Hannes Höppner, Amir Jafari, Matteo Laffranchi, Dirk Lefeber, F. Petit, and Sebastian Wolf. Variable impedance actuators: a review. *Robotics and Autonomous Systems*, 61:1601–1614, 12 2013.
- [79] Bram Vanderborght, Nikos G Tsagarakis, Ronald Van Ham, Ivar Thorson, and Darwin G Caldwell. Maccepa 2.0: compliant actuator used for energy efficient hopping robot chobino1d. *Autonomous Robots*, 31(1):55, 2011.
- [80] Milos Vasic and Aude Billard. Safety issues in human-robot interactions. In *2013 IEEE International Conference on Robotics and Automation*, pages 197–204. IEEE, 2013.
- [81] Valeria Villani, Fabio Pini, Francesco Leali, and Cristian Secchi. Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics*, 55:248–266, 2018.

- [82] Christian Vogel. Safeguarding and supporting future human-robot cooperative manufacturing processes by a projection- and camera-based technology. 12 2017.
- [83] H. Wei, Z. Huang, Q. Yu, M. Liu, Y. Guan, and J. Tan. Rgmp-ros: A real-time ros architecture of hybrid rtos and gpos on multi-core processor. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2482–2487, 2014.
- [84] P. Yadmellat and M. R. Kermani. Adaptive control of a hysteretic magnetorheological robot actuator. *IEEE/ASME Transactions on Mechatronics*, 21(3):1336–1344, 2016.
- [85] P. Yadmellat, A. S. Shafer, and M. R. Kermani. Design and development of a single-motor, two-dof, safe manipulator. *IEEE/ASME Transactions on Mechatronics*, 19(4):1384–1391, 2014.
- [86] Peyman Yadmellat, Alexander S Shafer, and Mehrdad R Kermani. Design and development of a single-motor, two-dof, safe manipulator. *IEEE/ASME Transactions on Mechatronics*, 19(4):1384–1391, 2013.
- [87] Yoji Yamada, Yasuhiro Hirasawa, Shengyang Huang, Yoji Umetani, and Kazutsugu Suita. Human-robot contact in the safeguarding space. *IEEE/ASME transactions on mechatronics*, 2(4):230–236, 1997.
- [88] Gil Ho Yoon. Topological layout design of electro-fluid-thermal-compliant actuator. *Computer Methods in Applied Mechanics and Engineering*, 209:28–44, 2012.
- [89] Zettlex. *Inc der Product Guide Midi Range 75-300mm Inductive Angle Encoders*, 7 2018. Rev 4.11.5.

Curriculum Vitae

Name: Ziqi Yang

Post-Secondary The University of Western Ontario

Education and London, Ontario, Canada

Degrees: 2018 - 2020 M.E.Sc

Electrical and Computer Engineering Dept., Robotics and Control

Shenyang University

Shenyang, China

2014 - 2018 B.E.Sc.

School of Information Engineering

Related Work Teaching Assistant

Experience: The University of Western Ontario

2019 - 2020