Acceleration Feedback Controller Processor Design of a Kids Personal Transporter (kPT)

Ronal P. Chand [†,*], Ravinesh Chand [†], Mansour Assaf [‡],
Parmesh R. Naicker [†], and Shailendra V. Narayan [†]

***Abstract*—In robotics, one of the major tasks in prototyping a robot after validating it through simulations is the design of its controller and implementing on processor using a hardware descriptive language such as field programmable gate arrays (FPGA) board,very-large scale integration (VLSI) chip, and application-specific integrated circuit (ASIC). This paper presents the development of the embedded application-specific instruction set processor(ASIP) design for acceleration feedback control laws of a nonholonomic autonomous kids personal transporter (kPT) robot, which contains two front driven wheels and a rear free-wheeling castor for balance with an omnidirectional obstacle sensor of limited detection range. The acceleration control laws are extracted from the total potentials, which is developed from the Lyapunov based Control Scheme (LbCS). The acceleration controllers were validated through simulations in Wolfram Mathematica software. The design could be implemented in hardware like VLSI chip, ASIC chip or an FPGA board for fast prototyping. Hence, this would provide a personal transporter for kids regardless of their physical and cognitive skills or abilities for safe navigation as the rider does not need to control the transporter.**

***Index Terms*—Kids Personal transporter, Artificial potential fields, Obstacle avoidance, Acceleration feedback controller , Processor design**

I. Introduction

Robots have assisted humans in a variety of endeavors, including the exploration of space and the depths of oceans. Currently, their applications are being extended to the fields of medicine, agriculture, food preparation, manufacturing, transportation and education to mention a few. In particular, transportation robots are an essential part of people's lives, and a lot of research is carried out on them. The need for increasing degrees of precision and autonomy in robotics has necessitated the development of increasingly complex systems [1] with the search for relatively simpler and cost effective control algorithms [2]. The development of IoT technologies has presented the possibility of autonomous vehicle (AV) technology, which has the potential to reduce vehicle accidents, energy consumption, pollution, and congestion while simultaneously assisting people in their daily activities [3]. There are various types of robots, such as autonomous vehicles [1], social robots [4], humanoids [5], industrial robots [6], and medical robots [7]. This research focuses on the specific area of robotics which deals with the control and design of autonomous personal transportation robots for kids.

Personal transporters like segways and e-scooters are available for children commercially; however, their riders need to manually control them. There have been reported accidents in [8], which shows that the riders have failed to control their transporter. Studies in the literature have also shown that research carried out on autonomous vehicles for kids mainly contains assistive wheelchairs [9], [10], which is mostly targeted for children with disabilities or injuries. Hence, there is a need for a fully automated vehicle for all children regardless of their physical and cognitive skills or abilities, such that the riders do not have to control the vehicle, rather enjoy the ride from one point to another which may include various obstacles and restrictions.

The transportation robot is driven by a chip or a logical circuit known as the processor, which receives, responds to, and executes the fundamental instructions. There are different types of processors in the embedded system such as General purpose processors (GPP), Special purpose processor (SPP) and Application-specific instruction set processor (ASIP). In this paper, the development of the embedded ASIP processor design of a proposed nonholonomic autonomous kids personal transporter (kPT) robot for acceleration feedback control laws, which could be implemented in hardware like VLSI chip, ASIC or an FPGA board for fast prototyping. The kPT robot contains two front driven wheels and a rear free-wheeling castor for balance with a limited detection range omnidirectional obstacle sensor. The acceleration control laws are extracted from the total potentials, which is developed from the Lyapunov based Control Scheme (LbCS) [11]–[31], which is essentially a artificial potential field method. The readers are referred to [32] for a detailed insight of LbCS. The presented kPT robot could be used to entertain children and also provide assistance to children who have difficulty in walking. Figure 1 shows the proposed top and side elevations of a kPT adopted and modified from [33].

II. Nonholonomic Kids Personal Transporter Kinematic Model

Definition 2.1: The two front-wheel driven kPT robot can be described by a disk with a radius r_s, where (x, y) is the center of mass. The kPT robot can be considered as the set

$$C = \{(z_1, z_2) \in \mathbb{R}^2 : (z_1 - x)^2 + (z_2 - y)^2 \leq r_s^2\}. \quad (1)$$

[‡] School of Information Technology, Engineering, Mathematics & Physics, The University of the South Pacific, Suva, Fiji.

[†] School of Mathematical & Computing Sciences, Fiji National University, Suva, Fiji.

*Corresponding author email: ronal.chand@fnu.ac.fj

979-8-3503-9988-2/22/$31.00 ©2022 IEEE

2022 IEEE 7th Southern Power Electronics Conference (SPEC) | 979-8-3503-9988-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/SPEC55080.2022.10058436

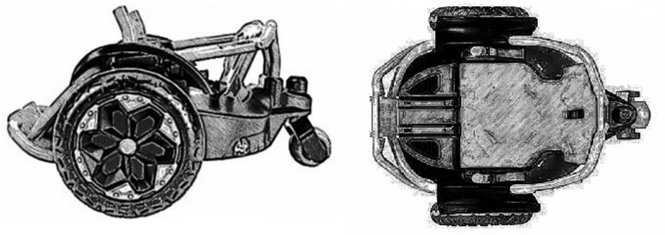


Fig. 1. Top and side elevations of a kPT with two front driving wheels with one rear castor wheel modified from [33].

The robot is equipped with two sets of diametrically opposed drive wheels, each with a radius of r, and a rear free-wheeling castor to maintain balance, and an omnidirectional obstacle sensor of detection range of r_d as shown in Figure 2. With

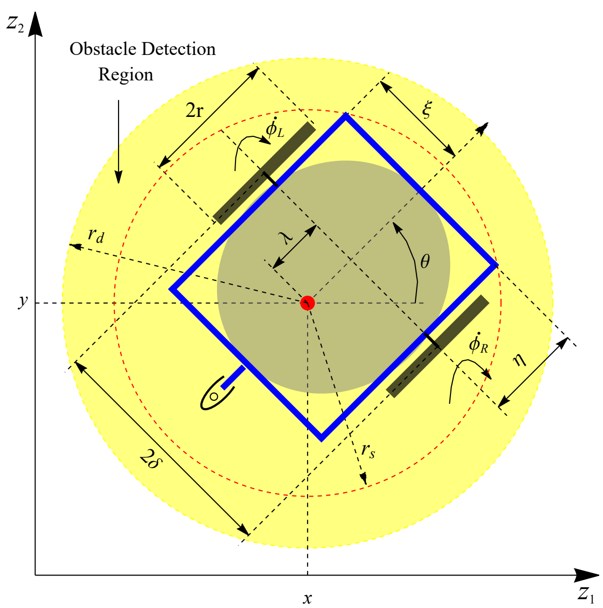


Fig. 2. kPT robot modelled as a wheeled mobile robot with two front driving wheels and one rear castor wheel.

respect to z_1-axis of the z_1z_2 cartesian plane the orientational angle of the kPT robot is θ. The center of mass is (x, y), and the distance from (x, y) to diametrically opposed wheels is λ. The robot is enclosed within a circular protective region with radius $r_s := \sqrt{\xi^2 + (\eta + \lambda)^2}$.

The configuration vector for the robot is,

$$\mathbf{x} = [x, y, \theta, \phi_R, \phi_L, \upsilon_R, \upsilon_L]. \tag{2}$$

Assuming that there is pure rolling of the two driven wheels with respect to $(x, y) \in \mathbb{R}^2$, the non-holonomic constraints of the robot are as follows:

$$\dot{y}\cos\theta - \dot{x}\sin\theta - \dot{\theta}\lambda = 0 \tag{3a}$$
$$\dot{x}\cos\theta + \dot{y}\sin\theta + \delta\dot{\theta} - r\dot{\phi}_R = 0 \tag{3b}$$
$$\dot{x}\cos\theta + \dot{y}\sin\theta - \delta\dot{\theta} - r\dot{\phi}_L = 0. \tag{3c}$$

The kinematic model could be derived as

$$\left.\begin{array}{l} \dot{x} = \frac{r}{2\delta}\Big(\upsilon_R(\delta\cos\theta - \lambda\sin\theta) + \upsilon_L(\delta\cos\theta + \lambda\sin\theta)\Big), \\ \dot{y} = \frac{r}{2\delta}\Big(\upsilon_R(\delta\sin\theta + \lambda\cos\theta) + \upsilon_L(\delta\sin\theta - \lambda\cos\theta)\Big), \\ \dot{\theta} = \frac{r}{2\delta}(\upsilon_R - \upsilon_L), \dot{\phi}_R = \upsilon_R, \dot{\phi}_L = \upsilon_L, \dot{\upsilon}_R = a_R, \dot{\upsilon}_L = a_L. \end{array}\right\} \tag{4}$$

III. Design of Acceleration-based Controllers

Consider a workspace that is only partially known to the robot and contains static obstacles. The personal transporter needs to navigate its way to its destination while avoiding the obstructions that are identified by the omnidirectional sensor.

Definition 3.1: The target for the kPT robot is a disk with center (τ_1, τ_2) and radius r_s. It is described as the set

$$B := \{(z_1, z_2) \in \mathbb{R}^2 : (z_1 - \tau_1)^2 + (z_2 - \tau_2)^2 \le r_s^2\}. \tag{5}$$

Definition 3.2: The k^{th} solid stationary obstacle is a disk with center (o_{k1}, o_{k2}) and radius $r_{O_k} > 0$. It is described as the set

$$O_k := \{(z_1, z_2) \in \mathbb{R}^2 : (z_1 - o_{k1})^2 + (z_2 - o_{k2})^2 \le r_{O_k}^2\} \tag{6}$$

for $k \in \{1, 2, 3, ..., q\}$.

A. Lyapunov Function Components

The following functions are created as part of the attractive and repulsive potential fields functions, known as the Lyapunov function,

1) Target Attraction: The following is a radically unbounded target attraction function that will ensure that the kPT robot converges to its target

$$V(\mathbf{x}) := \frac{1}{2}\Big((x - \tau_1)^2 + (y - \tau_2)^2 + \upsilon_R^2 + \upsilon_L^2\Big). \tag{7}$$

2) Stationary Obstacle Avoidance: To avoid collisions with the k^{th} static solid obstacle, for $k \in \{1, 2, 3, ..., q\}$, governed by equation (6), the following avoidance function is used:

$$E_k(\mathbf{x}) = \frac{1}{2}\Big((x - o_{k1})^2 + (y - o_{k2})^2 - (r_s + r_{O_k})^2\Big). \tag{8}$$

3) Artificial Obstacles Avoidance: It is noted that the angular velocities are restricted to a certain limit to suit a practical model. Thus, the angular velocities of the two-driven wheels are bounded as $|\upsilon_R(t)| < \upsilon_{max}$ and $|\upsilon_L(t)| < \upsilon_{max}$, where υ_{max} is the maximum angular velocity of the front right and left wheels. The limitations on the velocities of the wheels can only be incorporated in LbCS using artificial obstacles. The following avoidance functions are designed to account for the velocity limits that would be used in the design of repulsive potentials functions,

$$U_1(\mathbf{x}) = \frac{1}{2}\left(\upsilon_{\max}^2 - \upsilon_R^2\right) \tag{9a}$$
$$U_2(\mathbf{x}) = \frac{1}{2}\left(\upsilon_{\max}^2 - \upsilon_L^2\right). \tag{9b}$$

B. A Lyapunov Function

Let α_1, α_2, and β_k be positive control parameters, the total potentials for system (4) is as follows:

$$L(\mathbf{x}) = V(\mathbf{x}) + R(\mathbf{x})\left(\sum_{k=1}^{q}\frac{\beta_k}{E_k(\mathbf{x})} + \sum_{k=1}^{2}\frac{\alpha_k}{U_k(\mathbf{x})}\right) \tag{10}$$

where

$$R(\mathbf{x}) := \frac{1}{2}\Big((x-\tau_1)^2 + (y-\tau_2)^2\Big), \tag{11}$$

and

$$\beta_k = \begin{cases} \zeta(r_d^2 - d_{O_k}^2)^2, & \text{if } d_{O_k} \le r_d \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

$R(\mathbf{x})$ is an auxiliary function which ensures Lyapunov function to vanish at the target, while the obstacle avoidance parameter is β_k which is a saturation function that indicates that the kPT robot is navigating in a partially known environment, let $\zeta > 0$. It is noted that d_{O_k} is the Euclidean norm of the closest points on the boundary of the circular obstacle defined by equation (8) and the boundary of the circular disk with a radius r_ψ as described by equation (1).

C. Controller Design

Along the course of system's (4), $\dot{L}(x)$ upon collecting terms with υ_R, and υ_L could be arranged as

$$\dot{L}(x) = f\upsilon_R + g\upsilon_L,$$

where the functions f and g, on suppressing $\mathbf{x}$, are defined as:

$$f = \frac{r}{2\delta}\frac{\partial L}{\partial x}(\delta\cos\theta - \lambda\sin\theta) + a_R\Big(1 + \frac{\alpha_1 R}{U_1^2}\Big) + \frac{r}{2\delta}\frac{\partial L}{\partial y}(\delta\sin\theta + \lambda\cos\theta) \tag{13}$$

and

$$g = \frac{r}{2\delta}\frac{\partial L}{\partial x}(\delta\cos\theta + \lambda\sin\theta) + a_L\Big(1 + \frac{\alpha_2 R}{U_2^2}\Big) \quad \frac{r}{2\delta}\frac{\partial L}{\partial y}(\delta\sin\theta - \lambda\cos\theta) \tag{14}$$

Introducing the convergence parameters $\sigma_1 > 0$ and $\sigma_2 > 0$ such that

$$\dot{L} = -\sigma_1\upsilon_R^2 - \sigma_2\upsilon_L^2 \le 0,$$

then the nonlinear controllers of system (4) are obtained as:

$$\left.\begin{aligned} a_R = \;& \frac{-U_1^2}{U_1^2 + \alpha_1 R(\mathbf{x})}\Big(\frac{r}{2\delta}\frac{\partial L}{\partial x}\Big)(\delta\cos\theta - \lambda\sin\theta) \\ & -\frac{U_1^2}{U_1^2 + \alpha_1 R(\mathbf{x})}\Big(\frac{r}{2\delta}\frac{\partial L}{\partial y}\Big)(\delta\sin\theta + \lambda\cos\theta) \\ & -\frac{U_1^2}{U_1^2 + \alpha_1 R(\mathbf{x})}\Big(\sigma_1\upsilon_R + \Big(\frac{r}{2\delta}\Big)\frac{\partial L}{\partial\theta}\Big), \\ a_L = \;& \frac{-U_2^2}{U_2^2 + \alpha_2 R(\mathbf{x})}\Big(\frac{r}{2\delta}\frac{\partial L}{\partial x}\Big)(\delta\cos\theta + \lambda\sin\theta) \\ & -\frac{U_2^2}{U_2^2 + \alpha_2 R(\mathbf{x})}\Big(\frac{r}{2\delta}\frac{\partial L}{\partial y}\Big)(\delta\sin\theta - \lambda\cos\theta) \\ & -\frac{U_2^2}{U_2^2 + \alpha_2 R(\mathbf{x})}\Big(\sigma_2\upsilon_L - \Big(\frac{r}{2\delta}\Big)\frac{\partial L}{\partial\theta}\Big). \end{aligned}\right\} \tag{15}$$

Note, system (4) is ensured to be stable by the Direct Method of Lyapunov.

IV. Simulation

The results obtained were validated using computer simulations generated using the Wolfram Mathematica 12.3 programme. To produce the desired simulation results of system (4), a series of Mathematica instructions were ran sequentially. The RK4 method (Runge-Kutta Method) was used to numerically simulate the system. The assigned values of control and convergence parameters were obtained through brute-force technique.

Example 4.1: Consider a kPT robot attempting to move from its initial configuration to a specified final configuration while avoiding an obstacle in its route. The starting configuration, orientation, and final configuration (target) of the kPT robot with the stationary obstacle position are depicted in Figure 3(a). Table I shows the numerical values of the kPT robot's initial states, restrictions, control and convergence parameters. As time evolves, the robot comes closer to its destination, as depicted in Figure 3(a). The forward, backward, turning, and **zero-turn radius** movements are shown in this example. Figure 3(b) shows the evolution of the monotonically decreasing $L(\mathbf{x})$ and its time derivative. It shows that the kPT robot is approaching its destination. The angular velocities, υ_R and υ_L of the kPT robot are shown in Figure 3(c). The kPT robot's wheels have negative velocities, indicating that it is in reverse mode, and deceleration indicates that it is approaching its goal.

TABLE I

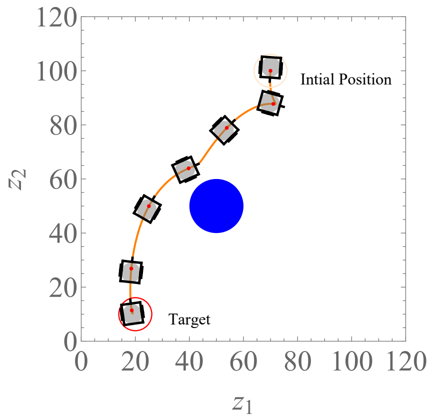
Example 4.1. Numerical values of the initial states, constraints, and control and convergence parameters of the kPT robot.

Initial Configuration	
Rectangular position	$(x_0, y_0) = (70, 100)$
Initial orientation	1.5 radians
Constraints	
Dimensions	$\delta = 4, r = 2, \lambda = 1, \eta = 4, \xi = 3.5$
Target location	$(\tau_1, \tau_2) = (20, 10)$
Omnidirectional Detection range	$r_d = 10$
Stationary obstacle Position	$(o_{11}, o_{12}) = (50, 50)$
Radius of fixed obstacle	$r_{O_1} = 10$
Maximum angular velocities	$\upsilon_{R max} = \upsilon_{L max} = 1$
Control and Convergence parameters	
Artificial obstacle avoidance	$\alpha_1 = \alpha_2 = 0.001$
Obstacle avoidance amplification	$\zeta = 0.0001$
Convergence	$\sigma_1 = \sigma_2 = 500$

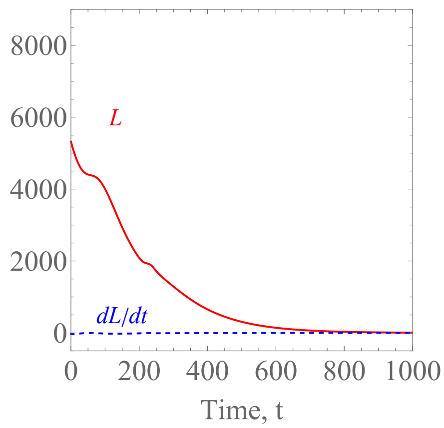
Example 4.2: This example considers the navigation of the kPT robot in an environment consisting of four randomly generated static obstacles. Stationary obstacle positions are randomly generated with the radius fixed between 2 and 10 units. Other initial configurations, constraints, and control parameters are kept the same as in Table I. Figure 4 shows the kPT robot avoiding randomly generated static obstacles present in its path while moving from its initial position to its final destination.

V. Electrical Design

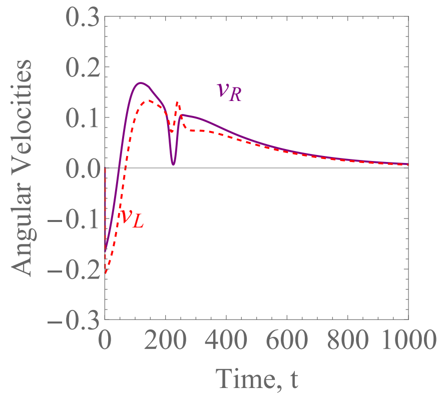
The kPT robot control topology, which is designed to be implemented on an FPGA board is described in this section. The



(a) Position and orientation at different times.



(b) Lyapunov function and its derivative.



(c) Angular velocities.

Fig. 3. (a) Position and orientation of the kPT robot at $t = 0, 72, 150, 250, 370, 600$ and 1200. The trajectory of the robot is shown in orange. (b) Monotonically decreasing Lyapunov function and its time derivative. (c) The front right and left wheels angular velocities of the kPT robot showing rapid deceleration as it approaches the target.

design of the controller as an Application Specific Instruction Set Embedded Processor (ASIP) is presented in Figure 5. The embedded processor design can be implemented in hardware on a VLSI chip or on an FPGA board for fast prototyping.

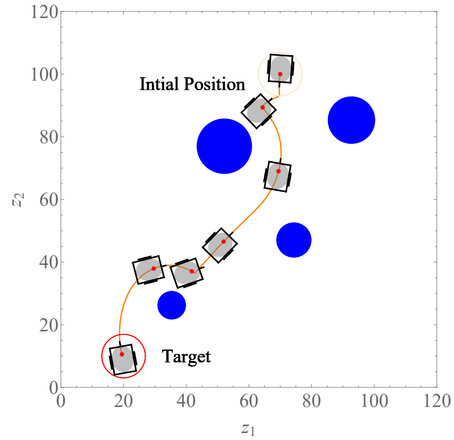


Fig. 4. Position and orientation of the kPT robot at $t = 1, 80, 230, 345, 670, 979,$ and 2000.

Register Transfer-level (RTL) design of the Assistive Robot system is provided in this section. Figure 6 shows the Finite

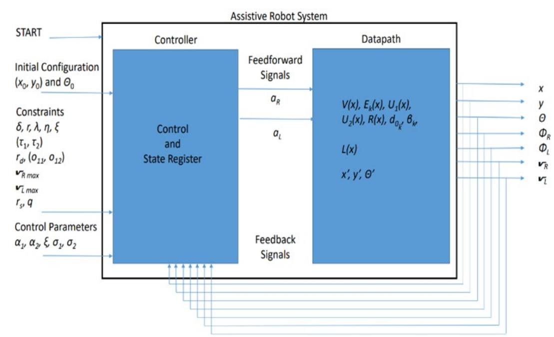
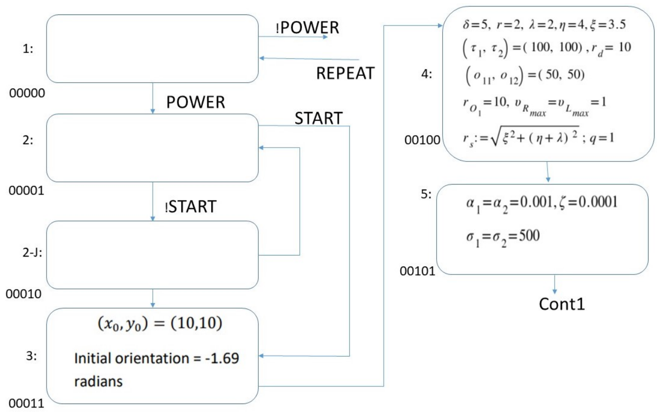


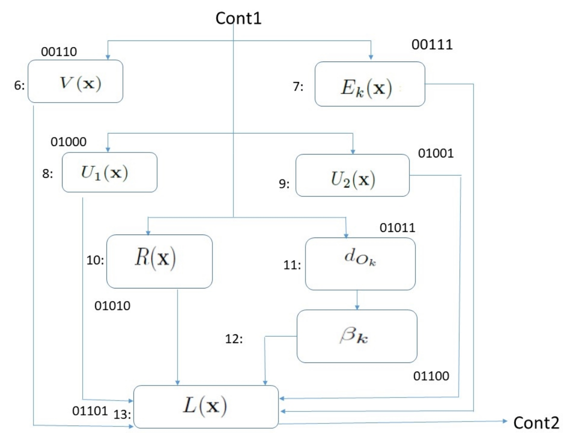
Fig. 5. The Assistive Robot high-level architecture.

State Machine with Datapath of the control algorithm, in which instructions and data are identified by separate unique states.

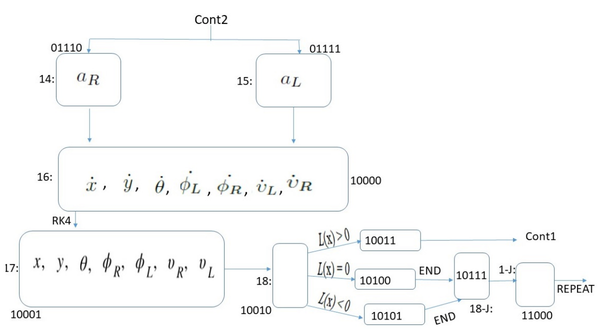
The complete design of the system is shown in Figure 7, where data and constant parameters are replaced by registers and instructions are replaced by control signals. Input control signals such as RBs, RBwa, RBwe, CALUs, Mre, and Mwe are sent out by the controller to configure the logic represented by the datapath and the memory. Output control signals such as CALUz are feedback signals returned by the datapath to the controller to indicate the status of execution of operations. The overall design consist of a control logic and state register unit (controller), a program counter (PC), instruction register (IR), multiplexer to route memory addresses, program memory, data memory, register bank, and custom arithmetic and logic unit (custom ALU) capable of performing floating point operations. Multiple custom ALU's can be used to perform parallel computation to speed up the next state processing and generation.



(a) Stage 1



(b) Stage 2



(c) Stage 3

Fig. 6. Finite State Machine with Datapath.

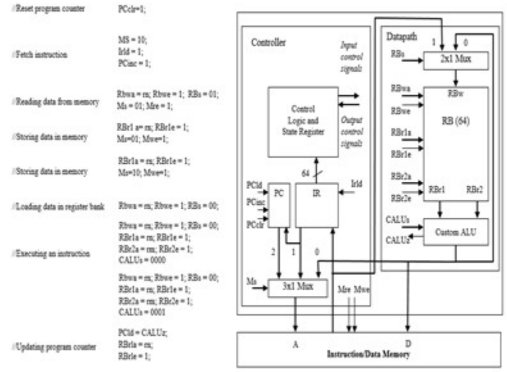


Fig. 7. Combined controller and datapath design.

VI. Conclusion

This paper is a design implementation study of a new autonomous kids personal transporter robot, which is equipped with an omnidirectional obstacle sensor of limited detection range. The kPT robot could provide entertainment for children and support parenting by carrying children around various activities where caregivers do not have to carry the child physically. The acceleration-based controllers were given preference over velocity controllers to provide a comfortable ride. The nonlinear time-invariant stabilizing acceleration controllers of the robot's front right and left wheels were constructed using the LbCS, allowing the robot to travel from an initial position to a destination location in the face of static impediments using its kinematic equations. Numerical proofs and computer-based solutions were used to demonstrate the functionality of the acceleration controllers. However, there is a risk of introducing algorithm singularities, which is a disadvantage of LbCS. In the future, a hybrid system will be created by merging the controllers described in this paper with a heuristic method, inheriting the benefits of LbCS while excluding its drawbacks. The ASIP processor design of the proposed kPT robot for acceleration feedback control laws is presented, which could be implemented in hardware like a VLSI chip, an ASIC chip, or an FPGA board for fast prototyping. Industries can adopt the model proposed in this paper for manufacturing autonomous vehicles for children to provide an assistive and entertaining means which can be used for various recreational activities involving children.

References

[1] I. AD. Nesnas, L. M. Fesq, and R. A. Volpe. Autonomy for space robots: Past, present, and future. *Current Robotics Reports*, 2(3):251–263, 2021.

[2] R. Chand, M. R. Valluri, and MGM Khan. Digital signature scheme over lattices. In *2021 25th International Conference on Circuits, Systems, Communications and Computers (CSCC)*, pages 71–78. IEEE, 2021.

[3] J. M. Anderson, K. Nidhi, K. D. Stanley, P. Sorensen, C. Samaras, and A. Oluwatola. *Autonomous vehicle technology: A guide for policymakers*. Rand Corporation, 2014.

[4] I. Leite, C. Martinho, and A. Paiva. Social robots for long-term interaction: a survey. *International Journal of Social Robotics*, 5(2):291–308, 2013.

[5] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini. Tactile sensing—from humans to humanoids. *IEEE transactions on robotics*, 26(1):1–20, 2009.
[6] E. Wang, C. Lee, and Y. Li. Assessing the impact of industrial robots on manufacturing energy intensity in 38 countries. *Energy Economics*, 105:105748, 2022.
[7] A. H. Himel, F. A. Boby, S. Saba, T. M. Akhund, N. Ullah, and K. Ali. Contribution of robotics in medical applications a literary survey. In *Intelligent Sustainable Systems*, pages 247–255. Springer, 2022.
[8] Greg Basky. Spike in e-scooter injuries linked to ride-share boom. *CMAJ*, 192(8):E195–E196, 2020.
[9] Laura Marchal-Crespo, Jan Furumasu, and David J Reinkensmeyer. A robotic wheelchair trainer: Design overview and a feasibility study. *Journal of neuroengineering and rehabilitation*, 7(1):1–12, 2010.
[10] H. Soh and Y. Demiris. Involving young children in the development of a safe smart paediatric wheelchair. In *ACM/IEEE HRI-2011 Pioneers Workshop*, 2011.
[11] S. A. Kumar, J. Vanualailai, B. Sharma, and A. Prasad. Velocity controllers for a swarm of unmanned aerial vehicles. *Journal of Industrial Information Integration*, 22:100198, 2021.
[12] S. A. Kumar, B. Sharma, J. Vanualailai, and A. Prasad. Stable switched controllers for a swarm of UGVs for hierarchal landmark navigation. *Swarm and Evolutionary Computation*, 65:100926, 2021.
[13] S. A. Kumar, J. Vanualailai, B. Sharma, A. Chaudhary, and V. Kapadia. Emergent formations of a Lagrangian swarm of unmanned ground vehicles. In *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–6, 2016.
[14] S. A. Kumar and J. Vanualailai. A Lagrangian UAV swarm formation suitable for monitoring exclusive economic zone and for search and rescue. In *Proceedings of the 2017 IEEE Conference on Control Technology and Applications*, pages 1874–1879, Kohala Coast, Hawai'i, USA, August 2017.
[15] A. Devi, J. Vanualailai, S. A. Kumar, and B. Sharma. A cohesive and well-spaced swarm with application to unmanned aerial vehicles. In *Proceedings of the 2017 International Conference on Unmanned Aircraft Systems*, pages 698–705, Miami, FL, USA, June 2017.
[16] A. Prasad, B. Sharma, J. Vanualailai, and S. A. Kumar. A geometric approach to target convergence and obstacle avoidance of a nonstandard tractor-trailer robot. *International Journal of Robust and Nonlinear Control*, 30(13):4924–4943, 2020.
[17] A. Prasad, B. Sharma, J. Vanualailai, and S. A. Kumar. Stabilizing controllers for landmark navigation of planar robots in an obstacle-ridden workspace. *Journal of Advanced Transportation*, 2020, 2020.
[18] S. A. Kumar, J. Vanualailai, and A. Prasad. Distributed velocity controllers of the individuals of emerging swarm clusters. In *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–6, 2020.
[19] B. Sharma, A. Prasad, R. Chand, S. Kumar, J. Raj, and J. Vanualailai. Landmarks as navigation-aids for multiple robots. In *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*, pages 8–14, 2020.
[20] B. Sharma, J. Vanualailai, and S. Singh. Lyapunov-based nonlinear controllers for obstacle avoidance with a planar n-link doubly nonholonomic manipulator. *Robotics and Autonomous Systems*, 60(12):1484–1497, 2012.
[21] S. A. Kumar, J. Vanualailai, and B. Sharma. Lyapunov functions for a planar swarm model with application to nonholonomic planar vehicles. In *2015 IEEE Conference on Control Applications (CCA)*, pages 1919–1924, 2015.
[22] S. A. Kumar, J. Vanualailai, and B. Sharma. Lyapunov-based control for a swarm of planar nonholonomic vehicles. *Mathematics in Computer Science*, 9(4):461–475, October 2015.
[23] A. Prasad, B. Sharma, J. Vanualailai, and S. A. Kumar. Motion control of an articulated mobile manipulator in 3d using the lyapunov-based control scheme. *International Journal of Control*, pages 1–15, 2021.
[24] S. A. Kumar, J. Vanualailai, and A. Prasad. Assistive technology: autonomous wheelchair in obstacle-ridden environment. *PeerJ Computer Science*, 7:e725:1–23, 2021.
[25] R. Chand, S. A. Kumar, and R. P. Chand. Navigation of an n-link revolute robotic arm via hierarchal landmarks. In *2021 3rd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pages 188–192, 2021.
[26] R. P. Chand, S. A. Kumar, and R. Chand. LbCS navigation controllers of twining Lagrangian swarm individuals. In *2021 3rd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pages 183–187, 2021.
[27] A. Prasad, B. Sharma, and S. A. Kumar. Strategic creation and placement of landmarks for robot navigation in a partially-known environment. In *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–6, 2020.
[28] S. A. Kumar, K. Chand, L. I. Paea, I. Thakur, and M. Vatikani. Herding predators using swarm intelligence. In *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–6, 2021.
[29] R. P. Chand, S. A. Kumar, R. Chand, and R. Tamath. Lyapunov-based controllers of an n-link prismatic robot arm. In *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–5, 2021.
[30] R. Chand, S. A. Kumar, R. P. Chand, and S. Reddy. A car-like mobile manipulator with an n-link prismatic arm. In *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–6, 2021.
[31] R. Chand, R. P. Chand, and S. A. Kumar. Switch controllers of an n-link revolute manipulator with a prismatic end-effector for landmark navigation. *PeerJ Computer Science*, 8:e885:1–27, 2022.
[32] B. Sharma, J. Vanualailai, and S. Singh. Tunnel passing maneuvers of prescribed formations. *International Journal of Robust and Nonlinear Control*, 24(5):876–901, 2014.
[33] L. Arneill. Spin and zig zag with the power wheels wild thing! {VIDEO REVIEW}. https://www.growingyourbaby.com/power-wheels-wild-thing/, 2016.