FPGA-Based Real-Time Wrench Model of Direct Current Driven Magnetic Levitation Actuator

Fengqiu Xu , *Member, IEEE,* Yun Lv, Xianze Xu, and Venkata Dinavahi , *Senior Member, IEEE*

Abstract—**Magnetic levitation actuators (MLAs) are increasingly employed in the modern high-precision manufacturing industry. Since the current-wrench transformation matrix is calculated inaccurately in the controller, the existing MLA finds it hard to undertake high-precision multi-degrees-of-freedom (DOF) movement in the large stroke. A high-fidelity real-time model is required to design better controllers for MLAs. This paper proposes a universal FPGA-based real-time wrench model (RTWM); meanwhile an MLA capable of translation and rotation in the horizontal plane is employed as the modeling object. To obtain the high accuracy with excellent generality, the force and torque distribution is solved through the magnetic node, coordinate transformation, and Gaussian quadrature. The corresponding calculation components are developed on the field-programmable gate array (FPGA) by exploiting the fully pipelined arithmetic and parallel architecture processing. Utilizing the high level synthesis tool, the final register transfer level structure is optimized adequately to reduce the time and hardware overhead. In the experiment, the RTWM of the proposed MLA and an existing MLA from the literature are obtained together to highlight the generality, and the computational accuracy is compared with the finite element method (FEM) software Comsol Multiphysics and a boundary element method software package Radia, respectively. The results show that the relative deviation is less than 2% choosing the FEM results as benchmark.**

Index Terms—**Field-programmable gate array (FPGA), hardware implementation, magnetic levitation actuator (MLA), parallel processing, real-time systems, wrench model.**

I. INTRODUCTION

MAGNETIC levitation actuators (MLAs) are able to provide accurate and controllable force resulting from the interaction between the permanent magnet and the ironless coil [1], [2]. As the friction, backlash, vibration, and wear from moving parts are eliminated, the performance of the motion control systems are improved [3]. Therefore, MLAs have been widely

F. Xu, Y. Lv, and X. Xu are with the Electrical Information School, Wuhan University, Wuhan 430072, China (e-mail: hncxu@whu.edu.cn; yunlv@whu.edu.cn; xuxianze@whu.edu.cn).

V. Dinavahi is with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: dinavahi@ualberta.ca).
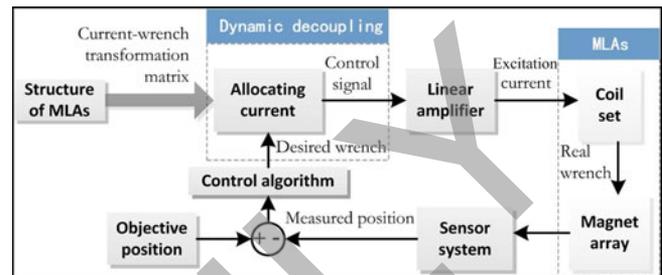
Fig. 1. Typical control block for MLAs with the dynamic decoupling unit.

employed in the modern industrial application, such as high precision positioning devices [4], [5], micromanipulations [6], [7], and haptic interactions [8], [9].

The main drawback of the MLA is that it cannot produce the accurate motion in a large stroke of rotation and translation simultaneously because it is difficult to decouple the magnetic force and torque in different directions. Fig. 1 is a typical control block for MLAs, and the desired force and torque acting on the translator is produced by allocating the exciting current of the coils in the dynamic decoupling unit. The elements of the current-wrench transformation matrix are required to be updated in each control sampling period [10]. However, it is difficult to construct the real-time wrench model (RTWM) which takes both the large translation and rotation into consideration precisely, thus producing the desired magnetic force and torque accurately when the translators undertake the multi-degrees-of-freedom (DOF) locomotion in a large range, which remains a considerable challenge.

Many designers employ delicate structures, such as magnetically levitated planar motor and magnetically levitated rotary machine, to enlarge the stroke and simplify the wrench model [11]–[13]. The MLA in [14] employs the one-dimensional (1-D) Halbach array and long racetrack coil. Since the magnetic interaction resulting from the short and arc side of the current carrying region can be ignored, the wrench model is simplified to an analytical expression to achieve the calculation in real-time. The work introduced in [15] proposes a magnetically levitated positioner using square coils and 1-D Halbach array. As the size of coils and magnets satisfy a certain relationship, the derived analytical expression of the electromagnetic force and torque would be streamlined, and has excellent accuracy with light computation burden. The magnetically levitated planar actuator in [16] utilizes the two-dimensional (2-D) Halbach array and herringbone coil set. Even if the interaction between

single magnet and coil are solved by the numerical method, where the symmetries existing in the MLA accelerate the computation. The above three magnetically levitated planar motors realize the large translational stroke, but their wrench models do not take the rotation of the translator into consideration, thus their range of rotation is confined in a small range where the magnetic fields and their gradients are nearly constant. On the other hand, the rotary MLA described in [17] is able to realize unlimited rotation around the vertical axis with the Halbach arrays mounted around the perimeter of the translator. Benefitting from the special structure, the torque with respect to the vertical axis can be expressed in an analytical style via the Maxwell equations and Lorenz force law in the polar coordinates. However, the wrench model deteriorates with the rotor translating in the horizontal plane, which makes the magnetically levitated rotary machine suffer from the drawback of short translational stroke. Thus, we can conclude that the existing RTWM cannot take all DOF into account at the same time [18]–[21], so the MLAs cannot undertake the accurate translation and rotation together in large range.

The motion control systems that are capable of realizing both the long-stroke translation and full rotation are required in many applications [22], [23], especially in semiconductor device fabrication and teleoperation. In order to realize the real-time computation of the magnetic force and torque distribution, look-up tables [24] and approximate analytical models [25] are two main approaches. Look-up tables demand the force and torque characteristics in advance thus a significant error can occur if the operation state is different from the predefined state. Additionally, the finite storing space of the look-up table affects the resolution of stored data. On the other hand, based on the axial symmetry of the circular coil, the approximate analytical method is able to construct the wrench model of the MLAs employing circular coil by considering the calculation results of some coefficients in the numerical integrals as constants. Nevertheless, the application scope of this method is limited as it cannot model the magnetic interaction resulting from rectangular or racetrack coils.

The numerical method is an alternative for modeling the electromagnetic actuators with unlimited translation and omnidirectional rotation [26], [27]. With such a modeling method, the electromagnetic force and torque are expressed as the summation of plenty of independent calculating units. However, the large computational burden existing in this modeling method hinders its utilization in the real controller. Currently, the general-purpose graphical processing units (GPGPU) [28], multicore digital signal processor (DSP) [29], and the field-programmable gate array (FPGA) [30], [31] are the alternatives to realize the computational acceleration with the parallel processing ability. Since the RTWM should be employed in the control system to calculate the current-wrench transformation matrix at each sampling cycle, the GPGPU cannot employed here as the most of its programs rely on a host CPU running an operating system, which cannot reduce the computational execution time down to microseconds and control the span of computation precisely. The multicore DSP is a real-time processor, however, the compute cores are too few for the numerical

computation in the wrench model and the interrupt employed to realize communication between the various cores hinder the response speed of the program. Thanks to the dramatic developments in the very large-scale integration (VLSI) technologies, the FPGA is considered as an advanced computational engine for real-time computation. With the following five advantages, FPGA is suitable for the implementation of the proposed RTWM:

1) with the nanosecond computational clock cycle, FPGA is an advanced real-time computational engine;
2) the FPGA possesses the inherent parallelism;
3) the reconfigurable architecture would realize the access of different calculation component concurrently;
4) it is very easy to realize the variable depth of pipelined implementation in the FPGA benefitting from the high level synthesis (HLS) and intellectual property (IP) cores provided by Xilinx, which is favorable to improve the computation efficiency;
5) with abundant resource of input/output ports, the FPGA is convenient to interface with the peripheral devices of the magnetically levitated system.

Therefore, aiming at a specific MLA, this paper studies the implementation method of RTWM on FPGA. The dc magnetic actuators with other design structure can also be modeled referring to this paper.

The rest of the paper is organized as follows. Section II introduces the numerical method to solve the force and torque briefly. The FPGA-based real-time digital hardware design for the corresponding computation is presented in Section III. In Section IV, the solution of the RTWM is compared with the finite element method (FEM) commercial software and boundary element method (BEM) programming package to highlight the calculation accuracy; the required computation resource, the calculation time step, and the validation of the obtained current vector based on the solution of RTWM are also discussed in this section. Furthermore, the RTWM for an existing MLA proposed in the previous literature is also tested to highlight the generality of the FPGA-based modeling method. Section V gives the conclusion of the paper.

II. NUMERICAL COMPUTATION OF FORCE AND TORQUE OF MLA

As a basic principle in the electromagnetism, the Lorenz force and torque integrals are applicable in the electromagnetic actuator. However, the rotation of the mover complicates the computation of integrals as identified in previous literature. In this paper, this issue is overcome through the numerical integral method.

In order to introduce the modeling method comprehensively, a typical MLA shown in Fig. 2 is used as the modeling object. Utilizing four 1-D Halbach magnet arrays containing 12 cuboid magnets and 8 independent iron-less racetrack coils in the translator and stator respectively, this MLA can rotate and translate in $\pm 10°$ and $\pm 10 \times \pm 10$ mm, respectively, in the horizontal plane. In such a movement range, the magnet array does not move away from the corresponding current region so that the coils
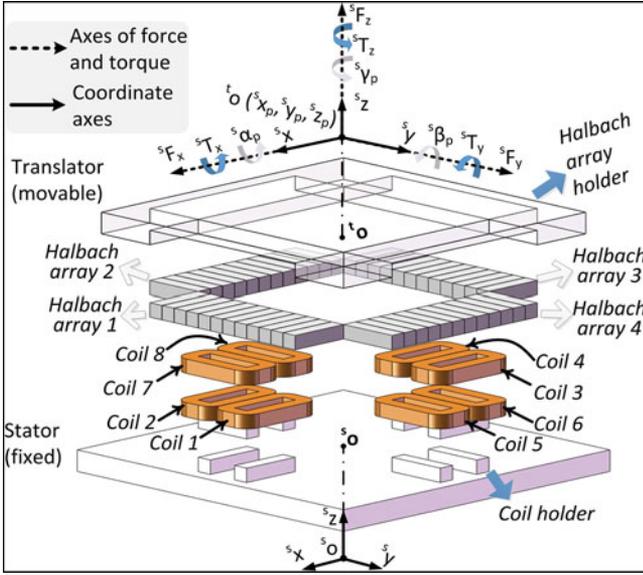
Fig. 2. Exploded view of the proposed MLA with the force and torque acting on the translator.

can provide sufficient force and torque to actuate the translator. With the translator coordinate system and the stator coordinate system defined as $\{t\}$ and $\{s\}$, respectively, the relative movement between them is given as a position and orientation vector (POV), $^s\mathbf{p} = [^s x_p, ^s y_p, ^s z_p, ^s \alpha_p, ^s \beta_p, ^s \gamma_p]$, and their relative integration is expressed as $^s\mathbf{w} = [^s F_x, ^s F_y, ^s F_z, ^s T_x, ^s T_y, ^s T_z]$. In the control system, POV is a vital variable on which the current-wrench transformation matrix $\mathbf{\Gamma}$ depends. In this MLA, we assume the capacitive sensors and 2-D optical encoders assembled in the center of the stator can be employed to measure the vertical or tilt motion and the horizontal translation or rotation, respectively.

With the sensor data obtained at each sampling cycle, current-wrench transformation matrix $\mathbf{\Gamma}$ is solved then used to calculate the current vector $\mathbf{I}$ containing the exciting current for each coil. Since there are eight coils in this MLA, the $\mathbf{\Gamma}$ is an 6×8 matrix. To minimize power dissipation, namely minimize the norm of current vector, $\mathbf{I}$ is solved by

$$\mathbf{I} = \mathbf{\Gamma}^+ \cdot \mathbf{w}_{\text{desire}} = \mathbf{\Gamma}^{\text{T}} \cdot \left(\mathbf{\Gamma} \cdot \mathbf{\Gamma}^{\text{T}}\right)^{-1} \cdot \mathbf{w}_{\text{desire}} \quad (1)$$

where $\mathbf{\Gamma}^+$ is the pseudoinverse of current-wrench transformation matrix, and the $\mathbf{w}_{\text{desire}}$ is the desired wrench vector for the translator [24], [32]. The contribution of the magnetic force and torque from the ith coil $[^s\mathbf{F}_i, ^s\mathbf{T}_i]$, defined as the wrench vector, is the ith column of and expressed below using the total current carrying volume as the integral region

$$^s\mathbf{F}_i = -\iiint_{\text{Vcoil}} {}^s\mathbf{J} \times {}^s\mathbf{B} \, dV \quad (2)$$

$$^s\mathbf{T}_i = -\iiint_{\text{Vcoil}} {}^s\mathbf{r} \times {}^s\mathbf{J} \times {}^s\mathbf{B} \, dV. \quad (3)$$

In (2) and (3), $^s\mathbf{J}$ is the current density, $^s\mathbf{r}$ is the moment arm, and $^s\mathbf{B}$ is the magnetic flux density. Because the shape of

the integral region is not constantly rectangular and the POV $^s\mathbf{p}$ is arbitrary, the solution of the integral in (2) and (3) is complicated and cannot always be analytical. Through the numerical approach based on magnetic node, Gaussian quadrature, and coordinate transformation, the Lorenz integral is transformed into the summation form as

$$^s\mathbf{F}_i = -\sum_{q=0}^{7} \sum_{g=0}^{n^3-1} \sum_{k=0}^{S-1} w_g \, {}^s\mathbf{J}_{q,g} \times {}_t^s\mathbf{R} \cdot {}^t\mathbf{B}_{q,g,k} \quad (4)$$

$$^s\mathbf{T}_i = -\sum_{q=0}^{7} \sum_{g=0}^{n^3-1} \sum_{k=0}^{S-1} w_g \, {}^s\mathbf{r}_{q,g} \times {}^s\mathbf{J}_{q,g} \times {}_t^s\mathbf{R} \cdot {}^t\mathbf{B}_{q,g,k} \quad (5)$$

where q, g, and k represent the index of coil segmentation, Gaussian node, and magnetic node, respectively. S is the amount of magnetic nodes in the translator, n is the order of Gaussian quadrature, ω_g is the weight of Gaussian quadrature, and $_t^s\mathbf{R}$ represents the rotation depending on the POV $^s\mathbf{p}$.

The wrench model of MLAs should be built by taking the design structure into full consideration. The detailed view of the actuating unit containing *Coil 1* and *Halbach array 1* are given in Fig. 3(a), where H_c, L_c, W_c, R_{out} describe the design parameter of the coil, H_m, L_m, W_m give the size of magnet, and $\tau_{c,x,1}$, $\tau_{c,y,1}$, $\tau_{m,x,1}$, and $\tau_{m,y,1}$ represent the position of coil and Halbach array in coordinate system $\{s\}$ and $\{t\}$, respectively.

Based on whether magnetization direction is orthogonal or not, the magnets in a Halbach array are divided into two groups as shown in Fig. 3(b). Sharing the same coordinate origin $^m o$, two coordinate systems $\{m_1\}$ and $\{m_2\}$ are built, in which the magnetic nodes are defined. Under this circumstance, the magnitude of these overlapping magnet nodes cannot be offset or superimposed. To obtain the magnetic field distribution, the contributions from these nodes are solved based on the method in [33], then summed up the results after the coordinate transformation.

For taking the arc section of the current carrying region into account, the coil is divided into eight parts as shown in Fig. 3(c). The nodes of Gaussian quadrature λ_g are able to be mapped in the coil region, which are named as coil nodes defined in coordinate system $\{c\}$ depicted in this figure. If n is equal to 3, the sectional view of the coil in the Fig. 3(c) on $^c x^c y$ plane or $^c y^c z$ plane contains nine coil nodes. The direction of current density on these coil nodes is decided by their location: The current direction is aligned with the $^c x$ or $^c y$ axis in the rectangular section; the current direction is the polar direction with the unit vector given as

$$\begin{cases} \widehat{x} = \cos\left(\frac{\pi}{4} + \frac{\pi}{4}\lambda_g + \theta_0\right) \\ \widehat{y} = \sin\left(\frac{\pi}{4} + \frac{\pi}{4}\lambda_g + \theta_0\right) \end{cases} \quad (6)$$

where θ_0 represent the original argument in each arc section.

III. REAL-TIME WRENCH MODEL ON FPGA

The current-wrench transformation matrix reflecting the force and torque contribution of each coil underneath the translator are required to determine the values of exciting current at

Fig. 3. (a) Zoomed view of *Coil 1* and *Halbach array 1* in MLA. (b) Exploded view of the 1-D Halbach magnet array. (c) Sectional view of the coil.

Fig. 4. Design procedure of RTWM containing the physical model, offline program, and the real-time digital hardware implementation.

The design procedure of the FPGA program for RTWM is depicted in Fig. 4. With the physical model of the MLA obtained, the wrench model was implemented on the central processing unit (CPU) through the programming environment such as MATLAB or Visual Studio to evaluate the effect of the offline program in advance. Then, the FPGA program is designed based on the offline calculation program via the Xilinx Vivado HLS tool. The detailed implementation method is described below.

A. Force and Torque Computation on CPU

To achieve the computation, the position of magnetic nodes and coil nodes are required to be extracted from the physical model into the computer memory for processing as shown in the left block of Fig. 4. The interaction between all of the coils and the magnet arrays are solved through the flowchart in the middle block which mainly contains following three steps.

1) Transforming the coil nodes into the same coordinates with magnetic nodes via the coordinate transformation.
2) Solving the magnetic flux density excited by the magnetic nodes with the method proposed in [33].
3) Calculating the summation equation described in (4) and (5).

At the end of the flowchart, the result is added to the corresponding memory employed to store the output of the wrench model. All magnetic nodes and coil nodes are required to be traversed in the computation.

B. Real-Time Computation With HLS Tool

FPGA programs work on the complex digital circuits in register-transfer level (RTL) structure using hardware description language (HDL), which is time-consuming and error-prone for the design of the complicated algorithm of specific tasks. High-level synthesis (HLS) introduced by Xilinx is able to translate C specification into an RTL implementation. Since C language has much higher abstraction than HDL, the coding effort for the achievement of (4) and (5) are substantially reduced. Furthermore, HLS provides abundant optimization methods for the RTL implementation including the pipeline, which is suitable for this paper.

As depicted in the right block of Fig. 4, the main loop in the offline program is specified to an RTL specification directly through the HLS, which is employed as an IP core in the final hardware implementation on FPGA. The execution time of the

every sampling time. The aforementioned method has potential to model the MLA with arbitrary POV, but the complicated operation process cannot obtain the real-time solution. As given in (4) and (5), these equations for magnetic force and torque possess the form of multiple summations, therefore, the wrench model is able to share the similar calculation units, and suitable to be implemented into a pipelined program. Additionally, with the independent coils employed in MLA, the inherent parallelism can realize the acceleration of computation. Taking these characteristic into account, FPGA not only has the parallel and the pipelined execution modes, but also owns the real-time data processing ability, which is suitable to accomplish this calculation task.

Fig. 5. Solution of magnetic flux density resulting from two magnetic nodes. (a) Diagram of one coil node and two magnetic nodes in MLA. (b) Flowchart of computation.

Fig. 6. Optimizing method to solve the coordinates of coil nodes in the arc current carrying region.

obtained IP core is estimated by

$$T_{\text{IP}} = (n_{\text{latency}} + n_{\text{interval}} \cdot N) \cdot T_{\text{clk}} \quad (7)$$

where T_{clk} means the clock period, n_{latency} and n_{interval} represent the number of the required clock cycles for the inherent latency and interval, and N is the execution times of the pipelined IP core. Actually, the main time overhead of the RTWM is resulting from T_{IP}, and there are three optimizing methods to reduce the computation time by analyzing the expression of (7).

1) Reducing the Interval Time of Acquired IP Core: The efficiency of the pipelined RTL structure can be increased significantly if the n_{interval} in (7) is equal to 1. Thus, the main loop of flowchart in the middle block of Fig. 4 cannot contain the circular statements such as `for` and `do-while`.

2) Ignoring the Effects of the Magnets far Away From the Object Coil: The interaction between the magnet and coil can be ignored when they are far from each other. As shown in Fig. 5(a), only the contribution of the *Halbach Array 1* is taken into consideration to solve the wrench vector of *Coil 1*. Furthermore, the magnetic flux density resulting from the two magnet nodes from the neighboring magnets are solved concurrently. One coil is divided into 8 segments, the 12 magnets exist in the Halbach array, and each magnet contains 8 magnetic nodes with N given as

$$N = 8 \cdot n^3 \cdot 12 \cdot 8 \div 2 = 384 \cdot n^3 \quad (8)$$

where n is the Gaussian order. The calculation process to solve the magnetic flux density at one step is described in Fig. 5(b).

3) Eliminating Trigonometric Functions for Solving the Coordinates of Coil Nodes: The final RTL structures for the trigonometric functions cost plenty of computation resources and result in significant latency. If the Gaussian quadrature order is known, the solution of the required trigonometric functions can be stored in register to calculate the position of coil node as depicted in Fig. 6. This figure zooms the Section ⑦ in Fig. 3(c) with the set $\{g_1, g_2\}$ employed to represent the index of Gaussian node. Since the Gaussian nodes λ_g, deciding the position of coil nodes, are constant if the order of Gaussian quadrature n is known, the results of sin and cos functions are
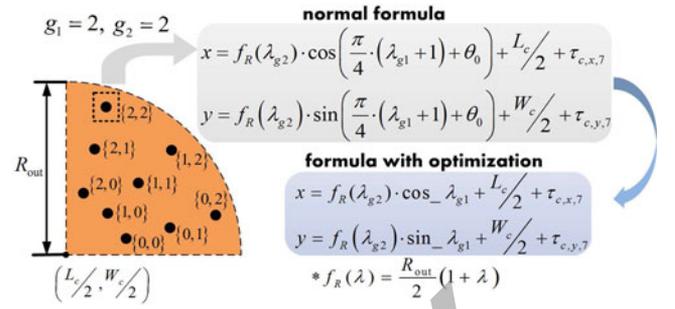
written as the corresponding parameter, $\cos\_\lambda_{g1}$ and $\sin\_\lambda_{g1}$, thus the computation of trigonometric functions are avoided. On the other hand, the coordinates of the coil nodes in other arc sections can be derived due to the $\cos\_\lambda_{g1}$ and $\sin\_\lambda_{g1}$ directly only by changing the sign.

C. FPGA Implementation and Latency Analysis

The detailed digital hardware implementation and the latency analysis for the RTWM are described in Fig. 7. The RTWM module contains two types of IP cores as depicted in Fig 7(a): The IP core exported by the HLS, and the **Sum** IP core provided by the LogicCore. The HLS IP core calculates the $[\mathbf{F}_{i,j}, \mathbf{T}_{i,j}]^{\text{T}}$ in (9) representing the relative interaction between two magnetic nodes and one coil node depicted in Fig. 5. Configured in pipeline, the HLS IP core needs an integer $j \in [0, N-1]$ provided by a counter, excepting the "clk," "rst," "start" signals and the POV data. The solutions are sent to six independent **Sum** IP cores to do the subsequent summation operation once the output is available. After the task of the HLS IP core is finished, the wrench vector is read from the **Sum** IP cores group after a constant latency

$$[\mathbf{F}_i(\mathbf{p}), \mathbf{T}_i(\mathbf{p})]^{\text{T}} = \sum_{j=0}^{N-1} [\mathbf{F}_{i,j}(\mathbf{p}), \mathbf{T}_{i,j}(\mathbf{p})]^{\text{T}}. \quad (9)$$

Furthermore, the finite state machine (FSM) for solving the wrench vector based on the RTWM is given in Fig 7(b). The computing program is triggered by a "time_step" signal, then the POV is read into the program and the RTWM module is invoked. After the RTWM module finishes the calculation, the register is updated and these IP cores are reset.

The FPGA-based RTWM is a two-stage pipelined structure. The total execution period of the RTWM, T_{sample}, is depicted in Fig. 7(c) and calculated in (10) with n_{HLS} and n_{sum} representing the inherent latency of HLS IP core and **Sum** IP core, respectively,

$$T_{\text{sample}} = (n_{\text{HLS}} + N + n_{\text{sum}}) \cdot T_{\text{clk}}. \quad (10)$$

Based on the inherent independence of the coils in the MLA, eight RTWMs are implemented on the FPGA in parallel to solve the wrench vector of each coil. These wrench vectors construct the final current-wrench transformation matrix. To maintain the
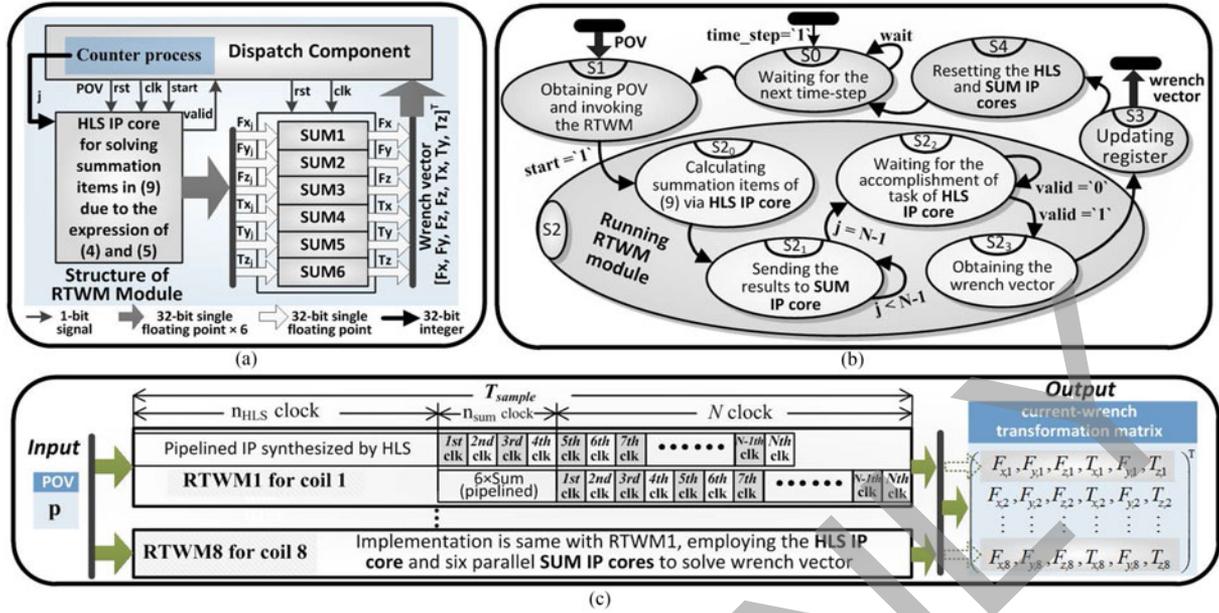
Fig. 7. (a) Digital hardware implementation of RTWM on FPGA. (b) FSM of computing wrench vector based on RTWM. (c) Latency analysis of FPGA-based RTWM.
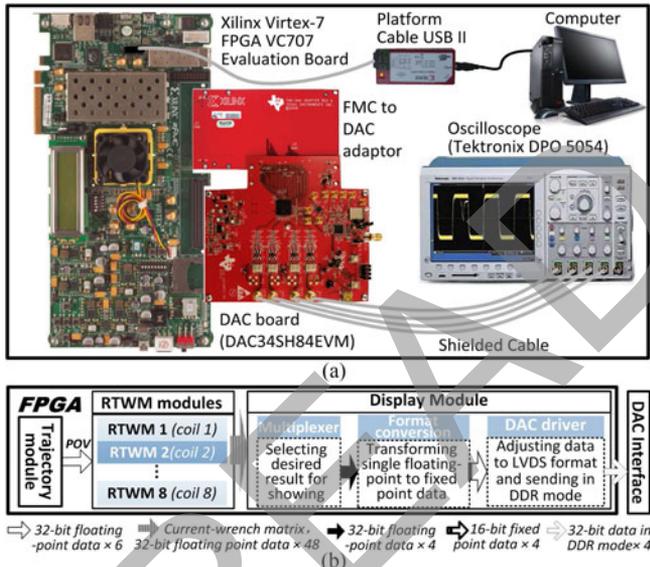
Fig. 8. (a) Hardware environment for the implementation of RTWM. (b) Computation modules in the FPGA implementation.

computation accuracy and efficiency, all of the IP cores used in the FPGA implementation are single precision floating-point.

IV. VALIDATION OF FPGA-BASED RTWM

In order to verify the calculation accuracy and efficiency of the proposed RTWM, the force and torque specification of the proposed magnetically levitated planar motor depicted in Fig. 2 are solved. The design parameters of the 1-D Halbach magnet array and the coil are listed in the appendix.

The hardware design is targeted to Xilinx Virtex-7 XC7VX485T with 100 MHz clock frequency shown in Fig. 8(a),

and the detailed implementation of the FPGA is given in Fig. 8(b). Eight RTWM components are designed and working in parallel to solve the wrench vector of the corresponding coils. The real-time solution is observed on the oscilloscope after it is transferred into the analog signals via the Texas Instruments DAC34SH84 evaluation module. As a low-power, high-dynamic range, quad-channel, and 16-b digital-to-analog converter (DAC), the DAC34SH84 employs the 32-b double data rate low voltage difference signal input data bus. In this case, the corresponding drive module is necessary in the FPGA to adjust the data format and synchronize the data clock, so that the display module in Fig. 8(b) contains the components of multiplexer, format conversion, and DAC driver. In this experiment, since the eight coils are independent with each other, we only observe the wrench vector produced by the *Coil 1* in Fig. 2 to validate the solution of RTWM in the following test. The *Coil 1* is excited by dc power with the 1 A constant current.

A. Computation Efficiency and Accuracy Analysis

The geometric center of the translator and the stator coincide and the corresponding axes align when POV $^s\mathbf{p}$ is equal to $\mathbf{0}$. We assume the trajectory of the translator meets (11). The units of $^s x$, $^s y$, and $^s z$ are mm, the units of $^s\alpha$, $^s\beta$, and $^s\gamma$ are rad, and t is time in second. Under this circumstance, the translation and rotation range of this MLA in $^s x^s y$ plane is $\pm 10 \times \pm 10$ mm and $\pm 11.25°$, respectively,

$$\begin{cases} ^s x = 10 \cdot t \cdot \cos\left(2\pi \cdot t\right) \\ ^s y = 10 \cdot t \cdot \sin\left(2\pi \cdot t\right) \\ ^s z = 12.5 + 2.5 \cdot t \end{cases} \text{ and } \begin{cases} ^s\alpha = 0 \\ ^s\beta = 0 \\ ^s\gamma = \frac{\pi}{16} \cdot \sin\left(2\pi \cdot t\right) \end{cases}.$$
(11)
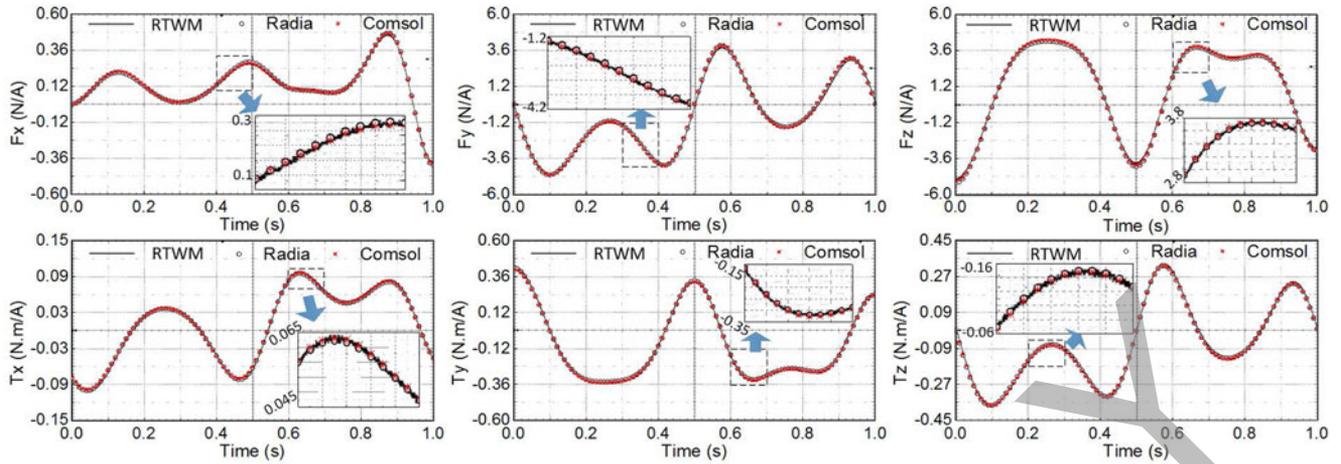
Fig. 9. Wrench vector of the *Coil 1* in the proposed MLA solved by RTWM, Comsol Multiphysics, and Radia.

TABLE I
COMPARISON OF COMPUTATION ACCURACY BETWEEN RTWM, COMSOL MULTIPHYSICS, AND RADIA

Elements in wrench vector	RMSE and RD with respect to Comsol			
	Solution of RTWM		Solution of Radia	
	RMSE(X_{RMSE})	RD(η_{RD})	RMSE(X_{RMSE})	RD(η_{RD})
$^s F_x$	3.70 (mN/A)	1.82%	4.61 (mN/A)	2.26%
$^s F_y$	39.2 (mN/A)	1.53%	45.5 (mN/A)	1.77%
$^s F_z$	43.5 (mN/A)	1.39%	46.4 (mN/A)	1.48%
$^s T_x$	0.72 (N.mm/A)	1.21%	0.79 (N.mm/A)	1.34%
$^s T_y$	3.80 (N.mm/A)	1.48%	3.84 (N.mm/A)	1.50%
$^s T_z$	3.15 (N.mm/A)	1.54%	3.67 (N.mm/A)	1.79%

1) Efficiency Analysis: In this case study, the Gaussian order is chosen as 4, thus N is calculated below due to (8)

$$N = 384 \cdot 4^3 = 24576. \tag{12}$$

Benefiting from the optimization method proposed in Section III, the n_{HLS} shown in Fig. 7 is reduced to 228 referring to the report of HLS tool. With the clock period is 10 ns, it is reasonable to choose n_{sum} as 4 in the FPGA implementation. Therefore, the T_{sample} given in Fig. 7 is 248.08 μs due to (10). With the idle time being 1.92 μs, the time step of the RTWM is 250 μs, so that the control frequency of the controller for this MLA could be set to 4 kHz employing this modeling method.

2) Accuracy Analysis: The RTWM results are compared with Comsol Multiphysics, a universal FEM software commonly employed in the simulation of electromagnetic device [34]–[38], and the Radia, a BEM program package, to highlight the accuracy. The three sets of results are shown in Fig. 9 and the period of simulation is 1 s. The output of the RTWM is the waveform which contains 4001 sampling points captured from the oscilloscope, and the solution of Comsol and Radia are superimposed on it to show the consistency. To distinguish the difference between the three solutions, the partial views of these curves in 0.1 s are inserted in each subfigure. Furthermore, because both the FEM and the BEM required a long solving time,

the sampling points of Comsol and Radia are decreased to 101. Even if the computation burden is reduced, the Radia and Comsol still spend more than 30 min and 6 h, respectively, to finish the computation.

Utilizing the solution of Comsol as the benchmark, the root mean square error (RMSE) of the solution of RTWM and Radia are listed in Table I, which are calculated by

$$X_{\text{RMSE}} = \sqrt{\frac{\sum_{i=0}^{100} (X_{\text{obs},i} - X_{\text{ref},i})^2}{101}} \tag{13}$$

where $X_{\text{obs},i}$ represents each element in the wrench vector from the RTWM or Radia at each sampling point, $X_{\text{ref},i}$ is the force and torque solved by Comsol. Meanwhile, the relative deviation (RD), written as η_{RD} and solved by (14), are also listed in Table I to reflect the difference between these solutions

$$\eta_{\text{RD}} = \sqrt{\frac{\sum_{i=0}^{100} (X_{\text{obs},i} - X_{\text{ref},i})^2}{\sum_{i=0}^{100} X_{\text{ref},i}^2}} \times 100\%. \tag{14}$$

3) Validation of RTWM for the Solution of the Current Vector: To evaluate the decoupling unit based on the RTWM, we assume that the MLA requires a wrench vector, $[0\,\text{N}, 0\,\text{N}, 20\,\text{N}, 0\,\text{N·mm}, 0\,\text{N·mm}, 50\,\text{N·mm}]$, acting on the mover when it moves along the trajectory of (11). After the current-wrench transformation matrix is solved by the RTWM, the current vector $\mathbf{I}$ is calculated through (1) at each time-step, and the results are shown in Fig. 10. Since the simulation time step is 250 μs, there are 4001 sampling points of the current value for each coil.

The condition number of the current-wrench transformation matrix at each time-step provides a measure of levitation stability and motion control performance. When the mover leaves the motion range, the condition number becomes large and the required current value would be unreasonable. Observing the solution, all of the required current values in Fig. 10 do not go beyond 2 A. These current values are reasonable and can be realized by the general current amplifier. Therefore, we conclude
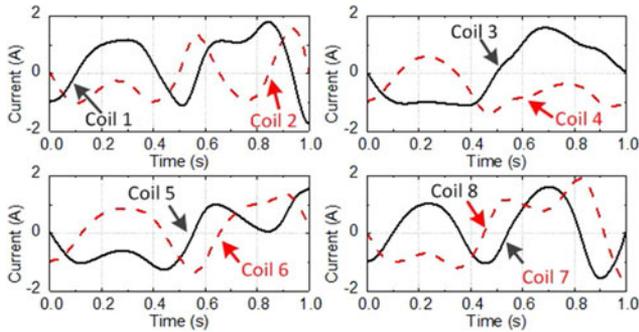
Fig. 10.   Current value of each coil in the proposed MLA.
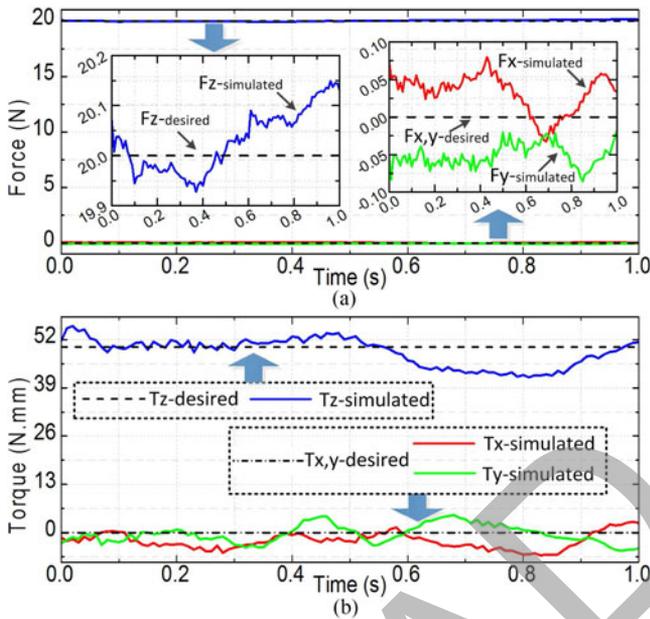
Fig. 11.   Force and torque solved by the finite element analysis with the obtained current vector. (a) Force elements. (b) Torque elements.
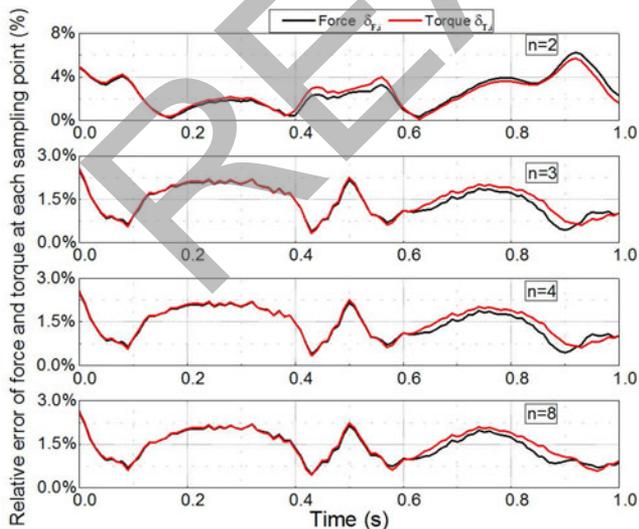
Fig. 12.   Relative error of magnetic force and torque magnitude at each sampling point due to different Gaussian order.

TABLE II
FPGA RESOURCE UTILIZATION FOR THE HARDWARE IMPLEMENTATION

FPGA Resources	Utilization of Hardware Modules		
	Trajectory	RTWM Module	Display
LUT	6433 (2.1%)	272 230 (89.7%)	1606 (0.5%)
FF	4546 (0.7%)	187 428 (30.9%)	680 (0.1%)
DSPs	56 (2%)	1414 (50.5%)	16 (0.6%)
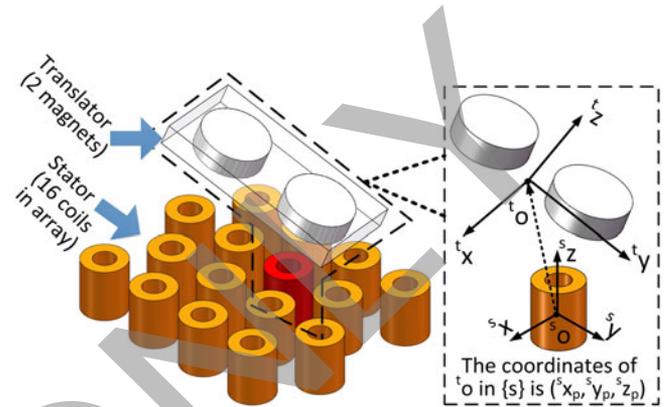
Fig. 13.   Structure diagram of the MLA proposed in [24].

the proposed MLA is capable of providing the sufficient force and torque in the translation and rotation range.

Additionally, the current results given in Fig. 10 are also verified by Comsol; 101 sampling points in each current data array are selected uniformly as the exciting condition, then the produced force and torque are solved via the FEM software. The simulation results presented as the solid lines in Fig. 11 approximate to the desired values given as dash lines, but some errors exist. We believe there are three reasons that cause the fluctuations. First, the unclear boundary condition of the MLA results in the variation in the solution of finite-element analysis. Second, the Comsol and the proposed RTWM use different methods to calculate the force and torque. Third, FEM is sensitive to minor perturbations in the model dimensions, finite element mesh, and solution parameters. However, despite the existing fluctuation in Fig. 11, the MLA is able to produce the required force and torque with the current values given in Fig. 10, and the disturbance in the wrench vector can be inhibited in the real device with feedback or other advanced control methods.

B. Relationship Between Computation Efficiency and Computation Accuracy

Since N in the RTWM is proportional to the cubic order of the Gaussian quadrature as depicted in (8), order of Gaussian quadrature n decides the computation time of each sampling point directly. The relationship between the computation accuracy and the efficiency is discussed here.

Assuming the POV varies due to the trajectory given in (11), the relative error of the force and torque between the RTWM
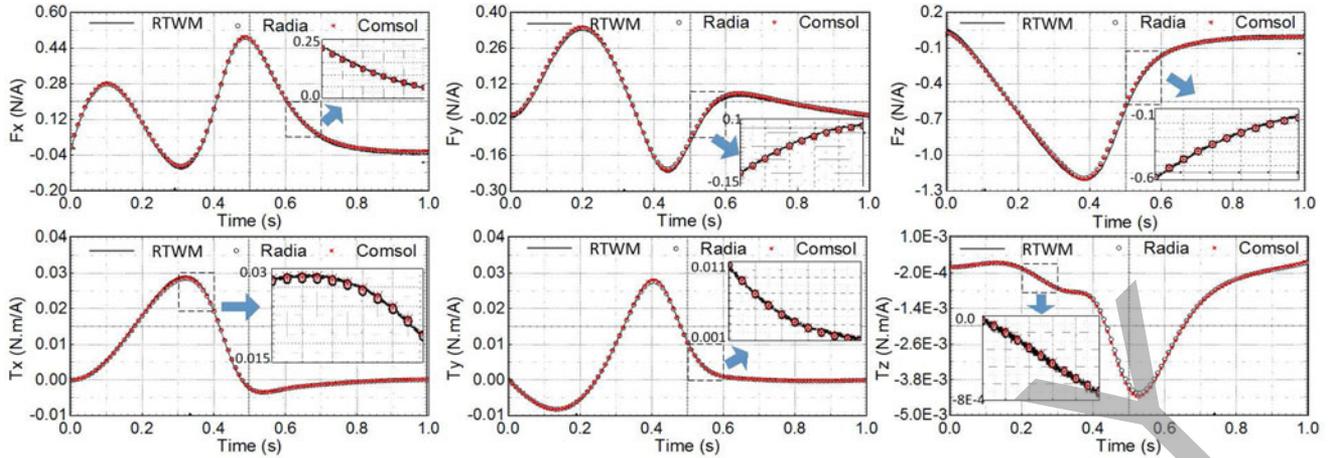
Fig. 14. Wrench vector of the marked coil in the MLA of [24] solved by RTWM, Comsol Multiphysics, and Radia.

and Comsol is obtained at each sampling point through

$$(\delta_{\mathbf{F},i}, \delta_{\mathbf{T},i}) = \left(\frac{\|\mathbf{F}_{\text{obs},i} - \mathbf{F}_{\text{ref},i}\|}{\|\mathbf{F}_{\text{ref},i}\|}, \frac{\|\mathbf{T}_{\text{obs},i} - \mathbf{T}_{\text{ref},i}\|}{\|\mathbf{T}_{\text{ref},i}\|} \right) \times 100\% \tag{15}$$

and there are 101 sampling points in these calculations. Fig. 12 shows the variation of relative error with respect to the POV due to the Gaussian quadrature order employed in the RTWM.

With the results depicted in Fig. 10, we can conclude that the computation accuracy does not improve definitely, but the summation items amount N increases significantly once n is larger than 3. Thus, we should choose the value of n reasonably in the RTWM to meet the accuracy requirement of the matching controller for the proposed MLA. If n is equal to 4, the time step of the computation is $250 \, \mu s$ as aforementioned, so the RTWM can be employed in a control unit with a 4 kHz sampling frequency. Actually, to maintain a higher control frequency, n can reduce to 3. Therefore, the time step reduces to $110 \, \mu s$ due to (8), and such an RTWM can be employed in the controller with 9 kHz sampling frequency.

C. Utilization of Computation Resource

As the wrench model involves plenty of floating-point calculations, the limitation of the computation resource in the FPGA is a challenge for the implementation of the RTWM. In Table II, we describe the occupied hardware resources, containing lookup table (LUT), flip flop (FF), and (DSP), of the XilinxVC707 Evaluation board for these modules. The trajectory, RTWM, and display modules in this table are corresponding to the content given in Fig. 8(b). The IP cores in the RTWM modules synthesized by HLS cost the majority of hardware resource in the FPGA. Under this circumstance, there exist difficulties in constructing the wrench model of some MLAs with large-scale coil set. We assume there are two methods to solve this issue.

1) Employing Multiple FPGAs to Implement the RTWM: Utilizing multiple FPGAs as processing units, more RTWM modules can be synthesized in parallel. In this case, the

computation efficiency is improved so the MLAs with large-scale coil set can be modeled in real-time.

2) Reducing the Dimension of the Current-Wrench Transformation Matrix: The RTWM only needs to solve the wrench vector of part of coils near to the magnet array, rather than all coils. Thus, the excitation coils in the stator should be selected dynamically at each control cycle.

D. Generalization of the Proposed RTWM

The utilization of the proposed RTWM will not be restricted by the design structure and motion range of the actuator. In addition to the design structure given in Fig. 2, many other multi-DOF MLAs without iron proposed in the previous literature can also be modeled and solved in real-time. To verify the generalization, the MLA proposed in [24] with large translation and rotation ranges in all directions is chosen as the test object.

$$\begin{cases} ^s x = 80 \cdot t \cdot \cos(2\pi \cdot t) \\ ^s y = 80 \cdot t \cdot \sin(2\pi \cdot t) \\ ^s z = 30 + 25 \cdot t \end{cases} \text{ and } \begin{cases} ^s \alpha = \frac{\pi}{6} \cdot t \\ ^s \beta = \frac{\pi}{6} \cdot t \\ ^s \gamma = 2\pi \cdot t \end{cases} . \tag{16}$$

The prototype of this MLA is depicted in Fig. 13 and its detailed size can be found in [24]. A certain trajectory is given by (16) using the same units of (11). We assume the translator moves along this trajectory. Referring to the approach aforementioned, the FPGA-based RTWM for this MLA is obtained, and the wrench vector of the coil in the dash block of Fig. 13 is shown in Fig. 14. Likewise, Fig. 14 shows excellent computation accuracy by superimposing the captured waveforms with the FEM and BEM results. The partial views of these curves in $0.1 \, s$ are also given to show the details. The relative errors of the force and torque components are less than 0.6% and 1.8%, respectively calculated by (14) using Comsol as benchmark.

To solve the current-wrench transformation matrix, 16 independent RTWM modules are employed for each coil. In every RTWM, 64×2 magnetic nodes represent the two magnets in the mover, and $4^3 \times 4$ coil nodes exist in the coil as it is divided into four arc segments with Gaussian order n set as 4. Thus, N

is given by

$$N = 64 \cdot 2 \cdot 4^3 \cdot 4 = 32768. \qquad (17)$$

In the FPGA implementation, n_{HLS} and n_{sum} are 170 and 4, respectively, so the T_{sample} in Fig. 7(c) is 329.42 μs due to (10). With the idle time being 3.91 μs, the time step of the RTWM is 333.33 μs. Additionally, the occupy ratio of the FPGA resources for the 16 RTWM components, including LUT, FF, DSPs, are 262,912(86.6%), 207,672(34.2%), and 2,512(89.7%), respectively.

With the testing results given above, we conclude that the proposed RTWM possesses the good generalization and can be employed in the functioning MLA with different structure.

V. CONCLUSION

The FPGA-based RTWM for the MLA was proposed in this paper. Based on the magnetic nodes, Gaussian quadrature, and coordinate transformation the wrench model was capable of solving the current-wrench transformation matrix of long stroke multi-DOF MLAs. The RTWM was implemented on the FPGA by taking advantage of the inherent pipelined and parallel characteristics to accelerate the computation, and the detailed design procedure based on the HLS was introduced. The RTWM was validated by comparing the solutions with commercial FEM software, Comsol Multiphysics and BEM programming package, Radia, and the results highlighted the excellent computation accuracy. The factors affecting the computation efficiency and hardware resource overhead are also discussed to guide the design of RTWM for the other MLAs. Additionally, the generality of the proposed RTWM is verified by modeling an MLA with different structure, and the good computation accuracy of the real-time solution is also highlighted by the comparison.

APPENDIX

The design parameters of the proposed MLA are given as follows: $R_{out} = 10$ mm, $L_c = 40$ mm, $W_c = 10$ mm, $H_c = 10$ mm, $L_m = 40$ mm, $W_m = 10$ mm, $H_m = 10$ mm, $\tau_{c,x,1} = 80$ mm, $\tau_{c,y,1} = 15$ mm, $\tau_{m,x,1} = 80$ mm, $\tau_{m,y,1} = 0$ mm, the remanence is 1T, and the turns of the coil is 300.

REFERENCES

[1] C. Hu, Z. Wang, Y. Zhu, M. Zhang, and H. Liu, "Performance-oriented precision LARC tracking motion control of a magnetically levitated planar motor with comparative experiments," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5763–5773, Sep. 2016.

[2] H. Zhang, B. Kou, Y. Jin, and H. Zhang, "Modeling and analysis of a new cylindrical magnetic levitation gravity compensator with low stiffness for the 6-DOF fine stage," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3629–3639, Jun. 2015.

[3] S. Li, L. Harnefors, and M. Iwasaki, "Analysis, and advanced control in motion control systems—Part I," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 3966–3970, Sep. 2012.

[4] W. J. Kim, S. Verma, and H. Shakir, "Design and precision construction of novel magnetic-levitation-based multi-axis nanoscale positioning systems," *Precis. Eng.*, vol. 31, no. 4, pp. 337–350, Oct. 2007.

[5] Z. Zhang and C. H. Menq, "Six-axis magnetic levitation and motion control," *IEEE Trans. Robot.*, vol. 23, no. 2, pp. 196–205, Apr. 2007.

[6] J. de Boeij, E. Lomonova, and J. Duarte, "Contactless planar actuator with manipulator: A motion system without cables and physical contact between the mover and the fixed world," *IEEE Trans. Ind. Appl.*, vol. 45, no. 6, pp. 1930–1938, Nov./Dec. 2009.

[7] M. Mehrtash, N. Tsuda, and M. B. Khamesee, "Bilateral macro-micro teleoperation using magnetic levitation," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 3, pp. 459–469, Mar. 2011.

[8] P. Estevez, A. Mulder, and R. H. M. Schmidt, "6-DoF miniature maglev positioning stage for application in haptic micro-manipulation," *Mechatronics*, vol. 22, no. 7, pp. 1015–1022, Oct. 2012.

[9] P. Berkelman and M. Dzadovsky, "Extending the motion ranges of magnetic levitation for haptic interaction," in *Proc. 3rd Joint Eurohaptics Conf. Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst.*, Mar. 2009, pp. 517–522.

[10] J. W. Jansen, C. M. M. van Lierop, and E. A. Lomonova, "Magnetically levitated planar actuator with moving magnets," *IEEE Trans. Ind. Appl.*, vol. 44, no. 4, pp. 1108–1115, Jul./Aug. 2008.

[11] H. Zhu, C. K. Pang, and T. J. Teo, "A flexure-based parallel actuation dual-stage system for large-stroke nanopositioning," *IEEE Trans. Ind. Electron.*, vol. 64, no. 7, pp. 5553–5563, Jul. 2017.

[12] L. Guo, H. Zhang, M. Galea, J. Li, and C. Gerada, "Multiobjective optimization of a magnetically levitated planar motor with multilayer windings," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3522–3532, Jun. 2016.

[13] I. U. R. Usman and X. Lu, "Force ripple attenuation of 6-DOF direct drive permanent magnet planar Levitating synchronous motors," *IEEE Trans. Magn.*, vol. 51, no. 12, Dec. 2016, Art. 8208708.

[14] C. Schaffel, M. Katzschmann, and H. U. Mohr, "High-resolution positioning device," U.S. Patent 9 479 040, 2016-10-25.

[15] H. Zhu, T. J. Teo, and C. K. Pang, "Design and modeling of a six-degree-of-freedom magnetically levitated positioner using square coils and 1-D Halbach arrays," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 440–450, Jan. 2017.

[16] J. M. M. Rovers, J. W. Jansen, J. C. Compter, and E. A. Lomonova, "Analysis method of the dynamic force and torque distribution in the magnet array of a commutated magnetically levitated planar actuator," *IEEE Trans. Ind. Electron.*, vol. 59, no. 5, pp. 2157–2166, May 2012.

[17] M. Dyck, X. Lu, and Y. Altintas, "Magnetically levitated rotary table with six degrees of freedom," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 1, pp. 530–540, Jan. 2017.

[18] J. Peng, Y. Zhou, and G. Liu, "Calculation of a new real-time control model for the magnetically levitated ironless planar motor," *IEEE Trans. Magn.* vol. 49, no. 4, pp. 1416–1422, Apr. 2013.

[19] B. Kou, F. Xing, L. Zhang, C. Zhang, and Y. Zhou, "A real-time computation model of the electromagnetic force and torque for a Maglev planar motor with the concentric winding," *Appl. Sci.*, vol. 7, no. 1, pp. 98, Jan. 2017.

[20] T. J. Teo, H. Zhu, and C. K. Pang, "Modeling of a two degrees-of-freedom moving magnet linear motor for magnetically levitated positioners," *IEEE Trans. Magn.*, vol. 50, no. 12, pp. 1–12, Dec. 2014.

[21] K. J. Meessen, J. J. H. Paulides, and E. A. Lomonova, "Force calculations in 3-D cylindrical structures using Fourier analysis and the Maxwell stress tensor," *IEEE Trans. Magn.*, vol. 49, no. 1, pp. 536–545, Jan. 2012.

[22] Y. Jiang, Y. Zhu, K. Yang, C. Hu, and D. Yu, "A data-driven iterative decoupling feedforward control strategy with application to an ultraprecision motion stage," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 620–627, Jan. 2015.

[23] A. T. A. Peijnenburg, J. P. M. Vermeulen, and J. V. Eijk, "Magnetic levitation systems compared to conventional bearing systems," *Micro Electron. Eng.*, vol. 83, no. 4, pp. 1372–1375, Apr. 2006.

[24] P. Berkelman and M. Dzadovsky, "Magnetic levitation over large translation and rotation ranges in all directions," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 1, pp. 44–52, Feb. 2013.

[25] F. Liu, M. Zhang, Y. Zhu, and C. Hu, "A real-time model of ironless planar motors with stationary circular coils," *IEEE Trans. Magn.*, vol. 51, no. 7, Jul. 2015, Art. 8203110.

[26] M. Miyasaka and P. Berkelman, "Magnetic levitation with unlimited omnidirectional rotation range," *Mechatronics*, vol. 24, no. 3, pp. 252–264, Apr. 2014.

[27] K. Bai, K. M. Lee, and J. Lu, "A magnetic flux model based method for detecting multi-DOF motion of a permanent magnet spherical motor," *Mechatronics*, vol. 39, no. 7, pp. 217–225, Nov. 2016.

[28] M. L. Balter, A. I. Chen, T. J. Maguire, and M. L. Yarmush, "Adaptive kinematic control of a robotic venipuncture device based on stereo vision, ultrasound, and force guidance," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 1626–1635, Feb. 2017.

[29] G. Wang, Y. Cheng, L. Zhang, and Z. Zhai, "Real-time image detail enhancement implementing on multi-core DSP platform," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 7035–7045, Dec. 2014.

[30] F. E. Fleming and C. S. Edrington, "Real-time emulation of switched reluctance machines via magnetic equivalent circuits," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3366–3376, Jun. 2016.

[31] M. Dagbagi, A. Hemdani, L. Idkhajine, M. W. Naouar, E. Monmasson, and I. Slama-Belkhodja, "ADC-based embedded real-time simulator of a power converter implemented in a low-cost FPGA: Application to a fault-tolerant control of a grid-connected voltage-source rectifier," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1179–1190, Feb. 2016.

[32] C. M. M. van Lierop, J. W. Jansen, E. A. Lomonova, A. A. H. Damen, P. P. J. van den Bosch, and A. J. A. Vandenput, "Commutation of a magnetically levitated planar actuator with moving-magnets," *IEEE Trans. Ind. Appl.*, vol. 128, no. 12, pp. 1333–1338, Dec. 2008.

[33] F. Bancel, "Magnetic nodes," *J. Phys. D, Appl. Phys.*, vol. 32, no. 17, pp. 2155–2161, Mar. 1999.

[34] D. V. Daniel, N. Tanase, E. S. Apostol, I. Chirita, and C. Ilie, "An overview regarding the analytical vs. numerical computation for a PMB used for FESS," in *Proc. 10th Int. Symp. Adv. Topics Elect. Eng.*, Mar. 2017, pp. 458–462.

[35] P. Haidl, A. Buchroithner, M. Bader, M. Zisser, B. Schweighofer, and H. Wegleiter, "Improved test rig design for vibration control of a rotor bearing system," in *Proc. 23rd Int. Congr. Sound Vibration*, Jul. 2016, pp. 1–8.

[36] R. Olaru, A. Arcire, C. Petrescu, M. M. Mihai, and B. Girtan, "A novel vibration actuator based on active magnetic spring," *Sensors Actuators A, Phys.*, vol. 264, no. 1, pp. 11–17, Sep. 2017.

[37] T. Espenhahn, D. Berger, S. Hameister, R. Hühne, L. Schultz, and K. Nielsch, "Design and validation of switchable tracks for superconducting levitation systems," *IEEE Trans. Appl. Supercond.*, vol. 27, no. 4, Jun. 2017, Art. no. 3600605.

[38] D. Fodorean, L. Idoumghar, M. Brevilliers M, P. Minciunescu, and C. Irimia, "Hybrid differential evolution algorithm employed for the optimum design of a high-speed PMSM used for EV propulsion," *IEEE Trans. Ind. Electron.*, vol. 64, no. 12, pp. 9824–9833, Dec. 2017.

Yun Lv was born in Suizhou, China. She received the B.Eng. degree in measurement and control technology and instrument from Wuhan University, Wuhan, China, in 2016. She is currently working toward the M.Eng. degree in instrument and meter engineering with the Electrical Information School, Wuhan University.

Her research interests include magnetically levitated technology and electromagnetic simulation.

Miss Lv was the recipient of the excellent Bachelor's degree thesis award of Hubei Province in 2016.

Xianze Xu received the B.Eng. degree in mechanical engineering from Hefei University of Technology, Hefei, China, in 1990, the M.Eng. and Ph.D. degrees in the mechanical engineering from Wuhan University of Technology, Wuhan, China, in 1992 and 2002, respectively.

He is currently a Professor with the Electrical Information School, Wuhan University, Wuhan, China. His research interest includes precision instruments design, manufacture, and control.

Fengqiu Xu (S'15–M'18) was born in Fengcheng, China. He received the B.Eng. degree in measurement and control technology and instruments and the Ph.D. degree in signal and information processing, both from Wuhan University, Wuhan, China, in 2011 and 2016, respectively.

In 2016, he was a Visiting Ph.D. student with the Department of Electrical and Computer Engineering, University of Alberta Edmonton, AB, Canada. He is currently a Postdoctoral with the Electrical Information School, Wuhan University. His research interests include magnetically levitated technology, real-time simulation of electrical machines, and finite element analysis.

Venkata Dinavahi (S'94–M'00–SM'08) received the B.Eng. degree in electrical engineering from Visveswaraya National Institute of Technology, Nagpur, India, in 1993, the M.Tech. degree in electrical engineering from the Indian Institute of Technology Kanpur, Kanpur, India, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2000.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems and power electronic systems, electromagnetic transients, device-level modeling, large-scale systems, and parallel and distributed computing.