DSP Based Digital Controller Design and Implementation for Energy Systems

Ahmed Talal Hashmat
Department of Electrical Engineering
Memorial University of Newfoundland
St. John's, NL, Canada
athashmat@mun.ca

Tahmid Tajwar
Department of Electrical Engineering
Memorial University of Newfoundland
St. John's, NL, Canada
tahmidt@mun.ca

Ashraf Ali Khan
Department of Electrical Engineering
Memorial University of Newfoundland
St. John's, NL, Canada
ashrafak@mun.ca

Abstract-- This paper presents a comprehensive study on the design and implementation of DSP-based digital controllers for energy systems, with a specific focus on utilizing the TMS320F28335 microcontroller from Texas Instruments. Energy systems play a critical role in various domains and improving their control strategies using advanced digital signal processing (DSP) techniques is of utmost importance. The paper begins with an overview of energy systems and the significance of efficient control mechanisms. It then explores the fundamental concepts of DSP and highlights the relevance of using the TMS320F28335 microcontroller for digital control applications in energy systems. The microcontroller offers a powerful combination of performance, features, and flexibility, making it suitable for implementing complex control algorithms.

Keywords— DSP, Microcontroller, PWM, ADC, Buck-Boost, PID

I. I. Introduction

In recent years, the field of energy systems has witnessed significant advancements driven by the increasing demand for efficient and reliable power generation, distribution, and conversion. To meet these demands, advanced control strategies that can optimize system performance, enhance stability, and improve energy utilization have become imperative. In this context, digital signal processing (DSP) techniques have emerged as a powerful tool for designing and implementing digital controllers in energy systems.

The paper investigates various digital control algorithms commonly used in energy systems, such as proportional-integral-derivative (PID) control, adaptive control, and model predictive control (MPC), and discusses their implementation on the TMS320F28335 microcontroller. The TMS320F28335 microcontroller is a highly capable and versatile platform that combines a powerful 32-bit DSP core with a rich set of peripherals [1], making it well-suited for complex control applications in energy systems.

The primary objective of this paper is to explore the potential of DSP-based digital controllers in improving the efficiency, stability, and reliability of energy systems. By leveraging the computational power and flexibility of the TMS320F28335 microcontroller, we aim to design and implement advanced digital control algorithms that can address the specific control requirements of different energy system components.

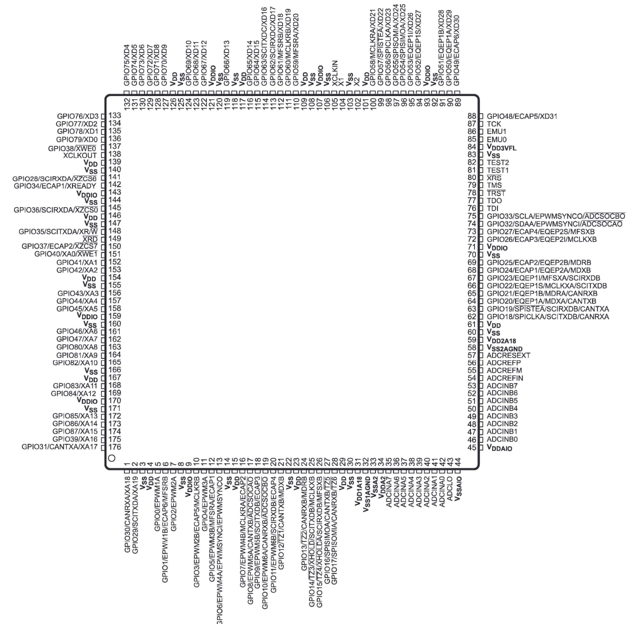

Fig. 1. Block representation of the TMS320F28335 Micro-Controller [2].

II. Microcontroller: TMS320F28335

In terms of hardware platforms, the TMS320F28335 microcontroller from Texas Instruments has gained significant attention in the literature. Researchers have utilized this microcontroller's capabilities, such as its high-performance DSP core, integrated peripherals, and real-time control features, for implementing DSP-based digital controllers in energy systems. These studies have demonstrated the suitability of the TMS320F28335 microcontroller for developing complex control algorithms and achieving efficient control of energy system components.

The TMS320F28335 is a member of the Texas Instruments (TI) C2000 microcontroller family, specifically designed for real-time control applications. It is designed to excel in real-time control applications, making it suitable for tasks such as motor control, power electronics, and digital signal processing (DSP). It offers high-performance digital signal processing capabilities, including built-in motor control peripherals and a floating-point unit (FPU) for efficient mathematical operations.

In addition to that communication interfaces such as UART, SPI, I2C, timers, and general-purpose input/output (GPIO) pins make it easier to interface with external devices and sensors commonly used in real-time control applications. These all is made possible by its high-performance 32-bit C28x core, offering significant processing power and high-speed execution for demanding real-time control tasks. Despite the progress made in the field, there are still challenges and areas for further exploration. For instance, the integration of emerging technologies like the Internet of Things (IoT) and artificial intelligence (AI) into DSP-based digital control systems for energy systems presents exciting research opportunities. Additionally, the development of optimized hardware architectures and the implementation of real-time communication protocols can further enhance the performance and capabilities of DSP-based digital controllers. [1-3]

TABLE I. MICRO-CONTROLLER FEATURES.[3]

Component	Specification
CPU	C28x
Frequency (MHz)	150
Flash memory (kByte)	512
RAM (kByte)	68
ADC resolution (Bps)	12
Total processing (MIPS)	150
Features	External memory interface
UART	3
CAN (#)	2
Sigma-delta filter	0
PWM (Ch)	12
Number of ADC channels	16
Direct memory access (Ch)	6
SPI	1
QEP	2
Communication interface	CAN, I2C, SPI, UART

III. BUCK-BOOST CONVERTER

A Buck-Boost Converter is a type of DC-DC converter that can step up or step down the input voltage to a desired output voltage level.

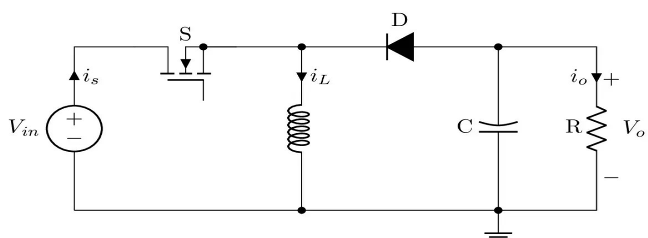

Fig. 2. Buck-Boost Converter

Let's consider the following parameters:

V_{in} = *Input voltage*

V_{out} = *Output voltage*

I_{in} = *Input current*

I_{out} = *Output current*

T_{on} = *Switch ON time*

T_{off} = *Switch OFF time*

$$D = \text{Duty cycle}\left(\frac{T_{on}}{\left(T_{on} + T_{off}\right)}\right) \quad 3.1$$

During the ON time (T_{on}) of the switch, the input voltage is connected to the inductor (L) and the output capacitor (C). During the OFF time (T_{off}), the inductor is connected to the output capacitor and the load.

The basic principle of the Buck-Boost Converter can be described as follows:

A. During Ton:

The current through the inductor is given by:

$$I_L = I_{in} + \frac{\left(V_{in} \times T_{on}\right)}{L} \quad 3.2$$

B. During Toff:

The current through the inductor is given by:

$$I_L = I_{out} - \frac{\left(V_{out} \times T_{off}\right)}{L} \quad 3.3$$

Output voltage is represented as:

$$V_{out} = V_{in} \times \frac{(1 - D)}{D} \quad 3.4$$

This equation gives you the output voltage of the Buck-Boost Converter as a function of the input voltage (*Vin*) and the duty cycle (*D*).

IV. ANALOG TO DIGITAL CONVERTERS (ADC)

DSP deals with digital signals, which are discrete-time and quantized representations of continuous analog signals. ADCs convert analog signals into digital form.

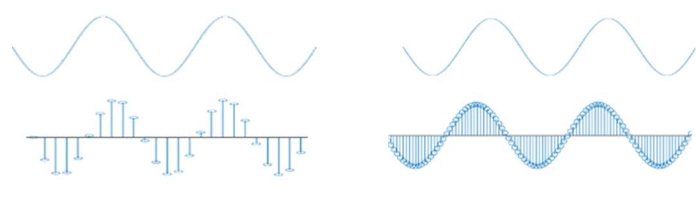

Fig. 3. Sampling rate representation.

The resolution for an ADC depends on the number of bits it has resolution = $1/2^N$ where, N = number of bits. Thus, a 12bit ADC (like that on the TMS320F28335 has a resolution of $1/2^{12}$ => one part in 4096. If the reference voltage is 3V, the voltage resolution is about 3V/4096 = 0.732 mV. Thus, to detect a change in an unknown voltage, it will have to change by more than 0.732 mV.

In a PID controller, the error (e) is the difference between the desired setpoint (SP) and the actual process variable (PV).

The controller's job is to minimize this error by adjusting the control variable (in this case, the duty cycle of the PWM signal).

Assuming,

$e(t)$ = *Error at time t* $\left(e(t) = SP - PV\right)$

Kp = *Proportional gain*

Ki = *Integral gain*

Kd = *Derivative gain*

$D(t)$ = *Duty cycle at time t*

The control output (in this case, duty cycle) is calculated based on the formula:

$$D(t) = Kp \times e(t) + Ki \times \int e(t)\,dt + Kd \times \frac{de(t)}{dt} \qquad 4.1$$

In the formula, the three terms represent the proportional, integral, and derivative actions, respectively. The integral term accumulates the past error values over time, and the derivative term considers the rate of change of the error.

To derive the duty cycle (D) directly from the error (e) without going through the PID control algorithm, we can rearrange the equation as follows:

$$D(t) = (1/Kp) \times \left[e(t) - Ki \times \int e(t)\,dt - Kd \times \frac{de(t)}{dt} \right] \qquad 4.2$$

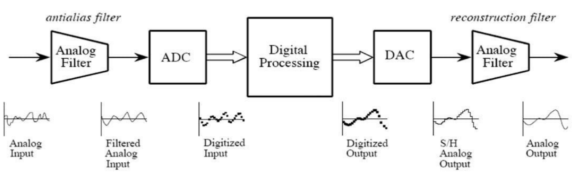

Fig. 4. ADC Process Diagram.[10]

In the case of TMS320F28335 Functions of the ADC module include:

- •12-bit ADC core with built-in dual sample-and-hold (S/H)
- Simultaneous sampling or sequential sampling modes
- Analog input: 0 V to 3 V
- Fast conversion time runs at 25 MHz, ADC clock, or 12.5 MSPS.
- 16-channel, multiplexed inputs
- Sequencer can be operated as two independent 8-state sequencers or as one large 16-state.

The digital value of the input analog voltage is derived by:

$$Digital\ Value = 4096 \times \frac{Input\ Analog\ Voltage\ - ADCLO}{3}$$

$$when\ 0\ V < input < 3\ V$$

A. Registers used involves:

- *ADCTRL3* operating in *SMODE,* Emulation-suspend mode.
- *ADCTRL1.bit.CPS* the CPS part in the control register represents the pre-scaler option.
- *ADCTRL1.bit.CONT_RUN* This bit determines whether the sequencer operates in continuous conversion mode or start-stop mode. In the continuous conversion mode, there is no need to reset the sequencer however, the sequencer must be reset in the start-stop mode to put the converter in state CONV00.
- *ADCREFSEL* used for ADC Voltage generation circuit options.

V. PWM Signal Generation

Pulse Width Modulation (PWM) is a technique to encode information in the form of varying pulse widths of a periodic signal. The duty cycle determines the average power delivered or the intensity of the output.

PWM signals are typically generated using a microcontroller, a dedicated PWM generator, or software implementation. The desired output is translated into a digital value representing the duty cycle.

$$D = \left(\frac{T_{on}}{T} \right) \times 100\% \qquad 5.1$$

The TMS320F28335 offers various clock sources such as internal oscillators or external crystals. We first need to configure the clock source for the microcontroller. The clock source needs to be appropriate based on our requirements. The on-chip oscillator and phase-locked loop (PLL) block in the TMS320F28335 provide the clocking signals for the device, as well as control for low-power mode (LPM) entry. The PLL has a 4-bit ratio control to select different CPU clock rates.[3]

We next Enable and configure the PWM module of the TMS320F28335. The specific registers and settings may vary depending on the microcontroller variant and the specific PWM module we intend to use.

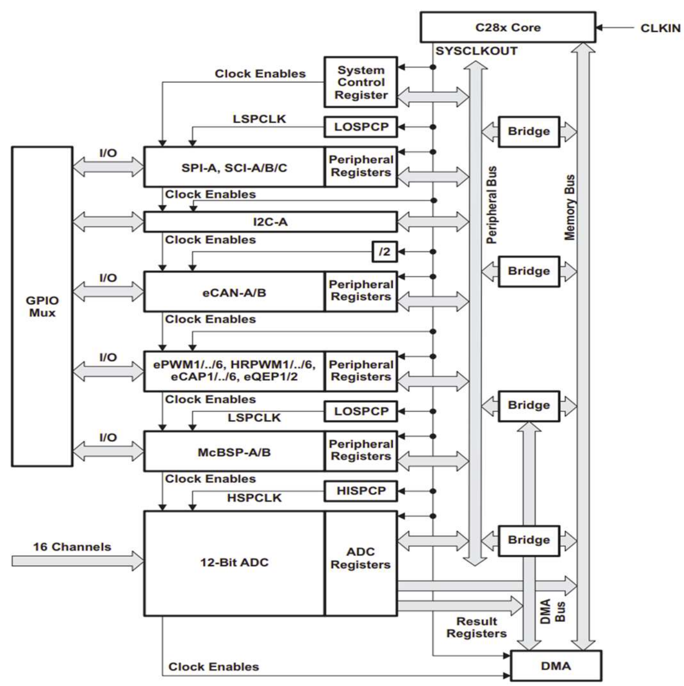

Fig. 5. Register Sequence for TMS320F28335.[2]

The desired period of the PWM signal needs to be determined. The period represents the time it takes for the PWM waveform to complete one cycle.

Appropriate registers need to be configured for the PWM period based on our desired frequency, which is determined by the duty cycle, it also determines the percentage of time the PWM signal is in the "on" state compared to the total period. Adjusting the duty cycle will enable us to control the power or intensity of the output signal. Most microcontrollers have dedicated registers to set the duty cycle for each PWM channel.

Once we have configured the PWM module, we need to set the necessary registers to initiate the PWM signal generation. This involves enabling the PWM output, starting the PWM timer, and configuring any additional parameters. If we need to change the duty cycle dynamically during runtime, we can update the appropriate registers to modify the PWM output.

In the case of TMS320F28335, the ePWM module represents one complete PWM channel composed of two PWM outputs: EPWMxA and EPWMxB.

Two PWM outputs (EPWMxA and EPWMxB) that can be used in the following configurations:

- Two independent PWM outputs with single-edge operation
- Two independent PWM outputs with dual-edge symmetric operation
- One independent PWM output with dual-edge asymmetric operation
- Asynchronous override control of PWM signals through software.
- Programmable phase-control support for lag or lead operation relative to other ePWM modules.

A. The registers that have been used are:

- TBCTL which is a time-based control register
- TBPRD This is period based register
- TBPHS is a phase register
- TBCTR is a counter register
- CMPCTL is a counter-compare control register
- AQCTLA is an Action-Qualifier control register for Output A

The ePWM modules are chained together via a clock synchronization scheme that allows them to operate as a single system when required

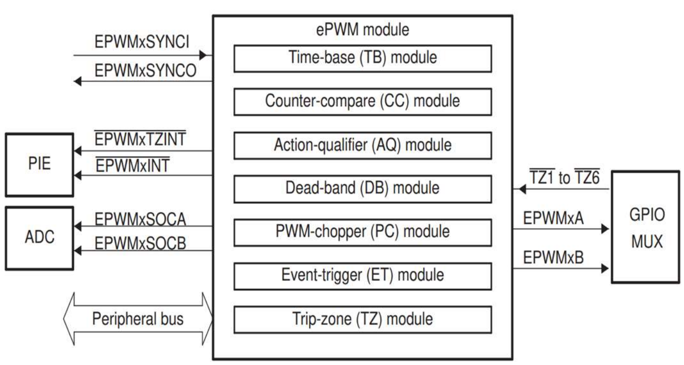

Fig. 6. Submodules and Signal Connections for an ePWM Module

Each ePWM module supports the following features, such as Dedicated 16-bit time-base counter with period and frequency control.

- Dead-band generation with independent rising and falling edge delay control.
- Programmable trip zone allocation of both cycle-by-cycle trip and one-shot trip on fault conditions.
- A trip condition can force either high, low, or high-impedance state logic levels at PWM outputs.
- Programmable event pre-scaling minimizes CPU overhead on interrupts. PWM chopping by high-frequency carrier signal, which is useful for pulse transformer gate drives.

VI. Methodology

Implementation of a Buck-Boost converter with the Micro-Controller was done for the initial phase, the below block diagram represents the pins used in the Micro-controller to activate the converter as demanded.

The pins 41, 42 and 5 are used to send the necessary signals. Where 41, and 42 are the ADC pins and 5 is an ePWM pin.

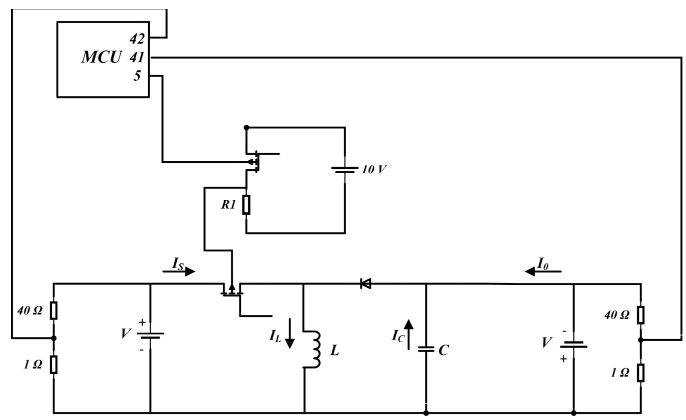

Fig. 7. Integration of the Buck-Boost Converter with the Micro-Controller.

VII. Simulation Results

When, V_{ref} is set to 24V the results obtained are shown by the figures below:

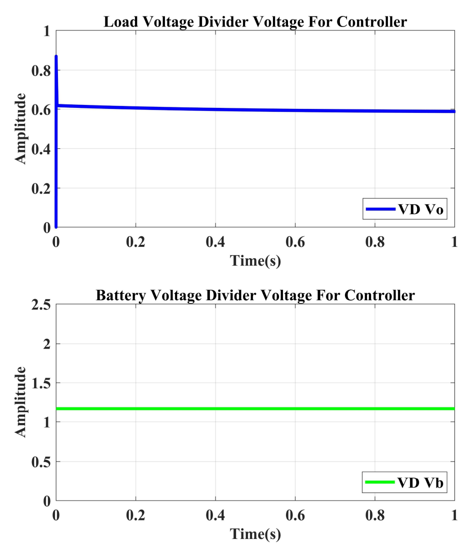

Fig. 8. Battery and load Voltage Divider Voltage for 24V

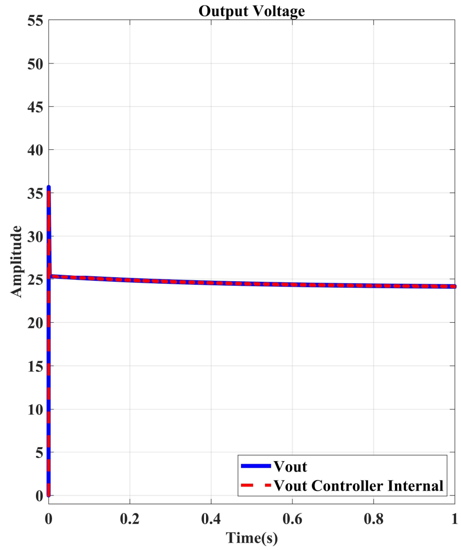

Fig. 9. Voltage sensed by controller vs actual voltage for 24V.

When V_{ref} is set to 96V the results obtained are shown by the figures below:

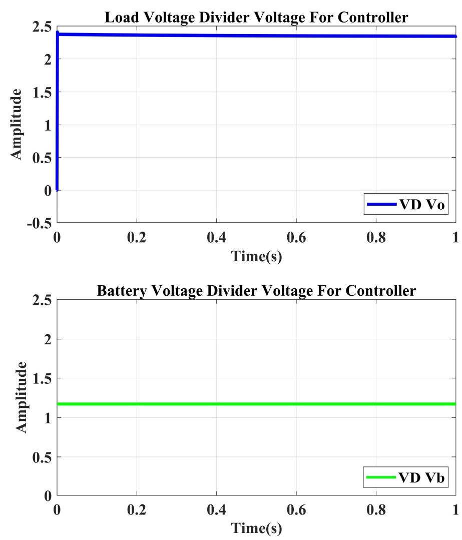

Fig. 10. Battery and load Voltage Divider Voltage for 96V

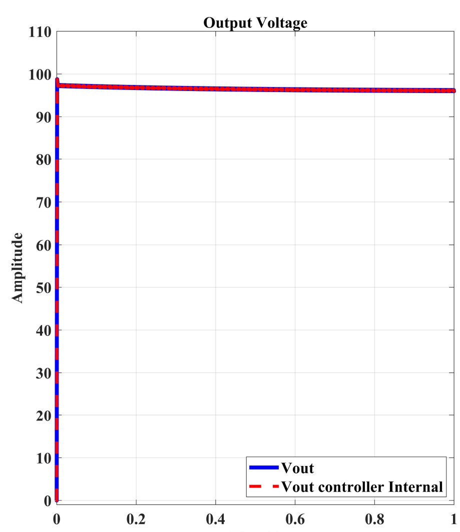

Fig. 11. Voltage sensed by controller vs actual voltage for 96V.

VIII. Conclusion

The project explores various control strategies, such as feedback control, feed-forward control, and adaptive control, to determine the most suitable approach for each energy system. The selection of control algorithms, including Proportional-Integral-Derivative (PID) control, model predictive control (MPC), and fuzzy logic control, will be based on their ability to optimize system performance and stability. The implementation phase will focus on integrating the digital controller into real-world energy systems, considering the hardware and software requirements for DSP implementation. By utilizing state-of-the-art DSP platforms, we aim to achieve real-time control capabilities, allowing for efficient and precise adjustment of system parameters.

References

[1] https://www.ti.com/product/TMS320F28335

[2] https://www.ti.com/lit/ds/symlink/tms320f28335.pdf?ts=1688377211295&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FTMS320F28335

[3] https://www.ti.com/lit/ug/sprui07/sprui07.pdf?ts=1688344929597&ref_url=https%253A%252F%252Flink.csdn.net%252F%253Ftarget%253Dhttp%253A%252F%252Fwww.ti.com%252Flit%252Fpdf%252Fsprui07

[4] R. San Vicente, R. Cortes, J. Robles and J. E. Chong-Quero, "DSP-microcontroller implementations of a simplified algorithm for synchrophasor calculation," 2005 2nd International Conference on Electrical and Electronics Engineering, Mexico City, Mexico, 2005, pp. 408-411, doi: 10.1109/ICEEE.2005.1529656.

[5] S. Banerjee and A. Sinha, "Performance analysis of different DSP algorithms on advanced microcontroller and FPGA," 2009 International Conference on Advances in Computational Tools for Engineering Applications, Beirut, Lebanon, 2009, pp. 609-613, doi: 10.1109/ACTEA.2009.5227848.

[6] A. Bianchi and M. Valiani, "DSP-Based and Microcontroller-Based Variable Frequency Drives for Domestic Washing Machine," 2007 IEEE Industry Applications Annual Meeting, New Orleans, LA, USA, 2007, pp. 1052-1055, doi: 10.1109/07IAS.2007.163.

[7] R. S. Bajwa, N. Schumann and H. Kojima, "Power analysis of a 32-bit RISC microcontroller integrated with a 16-bit DSP," Proceedings of 1997 International Symposium on Low Power Electronics and Design, Monterey, CA, USA, 1997, pp. 137-142, doi: 10.1145/263272.263309.

[8] L. -B. Chen, Y. -C. Liu, C. -H. Chen, C. -F. Kao and I. -J. Huang, "Parameterized embedded in-circuit emulator and its retargetable debugging software for microprocessor/microcontroller/DSP processor," 2008 Asia and South Pacific Design Automation Conference, Seoul, Korea (South), 2008, pp. 117-118, doi: 10.1109/ASPDAC.2008.4483923.

[9] R. Pittini, Z. Zhang and M. A. E. Andersen, "An interface board for developing control loops in power electronics based on microcontrollers and DSPs Cores -Arduino /ChipKit /dsPIC /DSP /TI Piccolo," 2013 IEEE 14th Workshop on Control and Modeling for Power Electronics (COMPEL), Salt Lake City, UT, USA, 2013, pp. 1-7, doi: 10.1109/COMPEL.2013.6626423.

[10] S. Boksuwan and T. Benjanarasuth, "Code generation of fractional filters for dsPIC microcontrollers," TENCON 2011 - 2011 IEEE Region 10 Conference, Bali, Indonesia, 2011, pp. 1275-1279, doi: 10.1109/TENCON.2011.6129012.

[11] R. Mittal and V. K. Jain, "Fault Detection in Three Phase Power Transmission Lines, a TI Microcontroller Implementation, and a VLSI Architecture," 2021 North American Power Symposium (NAPS), College Station, TX, USA, 2021, pp. 01-06, doi: 10.1109/NAPS52732.2021.9654580.