Research On Applying Mplab Device Block For Simulink In Teaching Digital Control Systems

Nguyen Huu Chan Thanh[1,*]

[1]Vietnam Aviation Academy, Vietnam
*Corresponding Author/Email: thanhnhc@vaa.edu.vn

Manuscript received: November 1 , 2024 / Revised: November 20, 2024 / Accepted: November 28, 2024

ABSTRACT

A low-cost rapid-prototyping system utilizing dsPIC microcontrollers, based on the MPLAB Device Blocks for Simulink development software, is introduced in this paper. This system provides an efficient and practical educational platform for students and researchers to understand and implement digital control systems in real time. By integrating Simulink with dsPIC, the system enables users to design, simulate, and deploy control algorithms without requiring extensive knowledge of embedded programming, thus bridging the gap between theoretical learning and practical implementation. The development software utilizes a graphical interface with block diagrams, allowing users to create control models easily. The real-time workshop (RTW) within Simulink generates optimized C code from these models, which is then automatically compiled, assembled, and downloaded onto the dsPIC microcontroller. This automation simplifies the entire process, eliminating the need for manual coding and debugging. The ability to implement real-time algorithms directly onto a hardware platform enhances the hands-on learning experience for students at both undergraduate and postgraduate levels. Clearly, this programming method is very useful for controllers of modern transportation vehicles such as drones and UAVs. Moreover, the system is cost-effective, making it accessible to educational institutions and small research labs. It supports a wide range of applications, including motor control, signal processing, and robotics, making it an invaluable tool for teaching and experimenting.

KEYWORDS: Rapid Prototyping, Fast Prototyping, Graphical Programming, Digital Control System, Real-time Control.

NOMENCLATURE

DSP = Digital Signal Processors
RTW = Real-Time Workshop
UAV = Unmanned Aerial Vehicle
PID = Proportional-Integral-Derivative

1. Introduction

The advancement of digital integrated circuit manufacturing technology, combined with modern control algorithms, has led to the development of high-performance digital signal processors (DSPs). These controllers offer excellent noise resistance, flexibility in updating and improving control programs, and the ability to meet the demands of both consumer and industrial applications.

However, a major challenge is that generating code for complex control algorithms embedded in DSPs requires a significant amount of time for programming and debugging. This difficulty extends the time from initial concept to market availability (time to market), reducing competitiveness and increasing research and development costs.

To overcome the issue of code generation time, major DSP manufacturers such as Texas Instruments, Microchip, and ST have developed and provided customers with useful fast code generation tools. In addition to common assembly and C languages, these tools, known as Fast Prototyping, Rapid Prototyping, or Graphical Programming, enable quick development across various platforms.

Some well-known graphical programming languages worldwide include:

- Texas Instruments C2000, C5000, C6000: Generates code for Texas Instruments DSPs running on the MATLAB Simulink platform.
- MPLAB Device Blockset: Generates code for Microchip DSCs running on the MATLAB Simulink platform.
- Waijung Blockset: Generates code for ST's ARM microcontrollers running on the MATLAB Simulink platform.
- Analog Devices, Freescale, Infineon: Generate code for DSPs from these manufacturers running on the MATLAB Simulink platform.
- LabVIEW: Generates code for devices from National Instruments.

In Vietnam, the dsPIC series is an affordable and widely used microcontroller line with high processing speed, diverse hardware modules, and easily accessible programming tools at a low cost. Therefore, this paper primarily focuses on applying the MPLAB Device Blockset for rapid code generation on dsPIC chips. This provides students and learners with a powerful tool for implementing digital control projects [1-3].

2. Using MPLAB Device Blockset for Simulink

2.1 System and software requirements

The following is a list of operating system and required softwares installations: [1]

- Windows 7 or later
- MATLAB 2010a or later
- Simulink
- Toolboxes: MATLAB Coder, Simulink Coder, Embedded Coder, Fixed-Point Designer, Signal Processing Toolbox, DSP System Toolbox, Control System Toolbox
- MPLAB X IDE
- MPLAB XC16 Compiler
- MPLAB XC32 Compiler
- MPLAB Device Blockset for Simulink

2.2 DSP KITs and tools requirements

The following is a list of Microchip DSP kits and required programming circuits:

- Microchip DSC KIT or DIY KIT,
- Programmer and debugger: PICKIT 3 or ICD 3.

2.3 Functions of the modules in MPLAB Device Blockset for Simulink [2]

2.3.1 System Configuration:

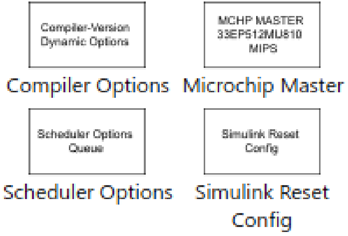

Figure 1: System Configuration Module [1]

- Compiler Options: Select the compiler.
- Microchip Master: Configure the chip and oscillator crystal.
- Scheduler Option: Configure multi-tasking.
- Simulink Reset Config: Allows resetting directly from Simulink

2.3.2 System Functions:

Figure 2: System Functions Module [1]

- Idle Task: Calls a subroutine.
- Interrupt: Calls a subroutine to handle interrupts.
- Reset: Defines conditions that will reset the chip.

2.3.3 System Info:

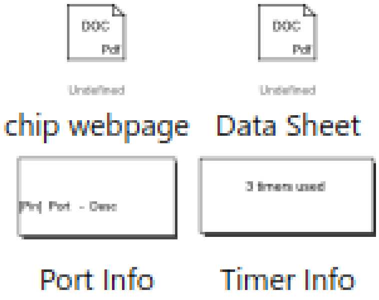

Figure 3: System Info Module [1]

- Data Sheet: Opens the chip's datasheet.
- Port Info: Lists the used pins of the chip.
- Timer Info: Lists the used timers of the chip.
- Chip Webpage: Redirects to the HTML page related to the selected chip.

2.3.4 Analog IO:

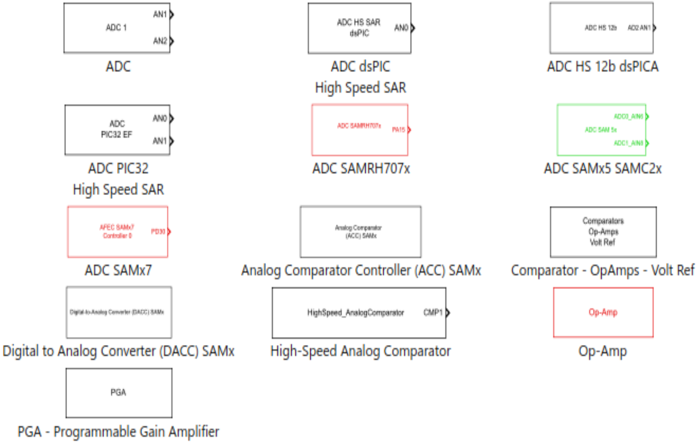

Figure 4: Analog IO Module [1]

- ADC: Configures the ADC module.
- ADC PIC32 High Speed SAR: Configures the ADC module for 32-bit PIC microcontrollers.
- High Speed Analog Comparator: Configures the comparator module on the chip…

2.3.5 Bus I2C, Bus SPI, QEI:

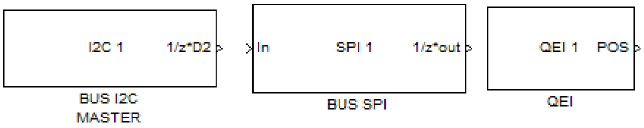

Figure 5: Bus I2C, SPI, QEI Module [1]

- I2C: Configures the I2C communication module.
- SPI: Configures the SPI communication module.
- QEI: Configures the encoder reading module.

2.3.6 Bus UART:

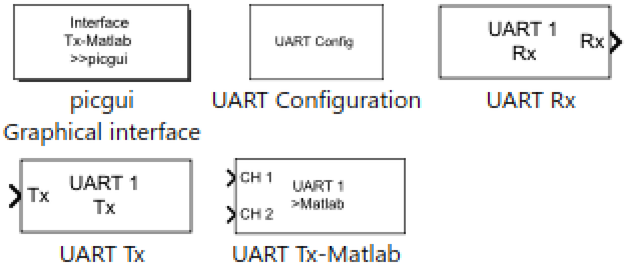

Figure 6: Bus UART Module [1]

- UART Configuration: Configures UART communication parameters such as baud rate, number of bits, etc.
- UART Rx: Configures the number of receiving channels from UART.
- UART Tx: Configures the number of transmitting channels to UART.
- UART Tx MATLAB: Configures the number of channels sent to MATLAB via UART.
- PicGUI: Opens a graphical interface in MATLAB to view signals sent from the chip.

2.3.7 Digital IO:

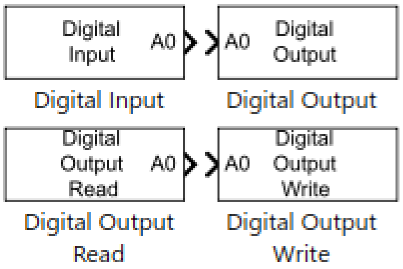

Figure 7: Digital IO Module [1]

- Digital In: Selects digital input pins.
- Digital Output: Selects digital output pins.
- Digital Output Read: Read from the digital output buffer register.
- Digital Output Write: Write to the digital output buffer register.

2.3.8 PWM IO:

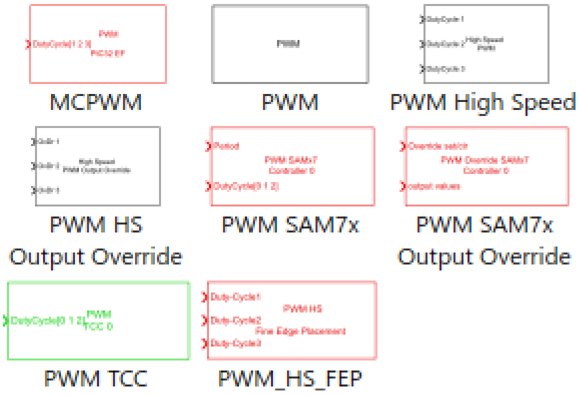

Figure 8: PWM IO Module [1]

- MCPWM: Configures the PWM module for motor control.
- PWM: Configures the Capture/Compare/PWM module.
- PWM HS: Configures the high-speed PWM module on 32-bit PIC microcontrollers…

2.3.9 Pulse Input/Output:

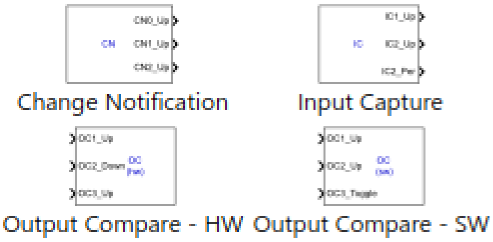

Figure 9: Pulse Input/Output Module [1]

- Change Notification: Configures input for detecting digital signal changes.
- Input Capture: Configures input for pulse capturing.
- Output Compare: Configures hardware-based digital output comparison.
- Output Compare - SW: Configures software-based digital output comparison using a timer.

2.3.10 Profiling:

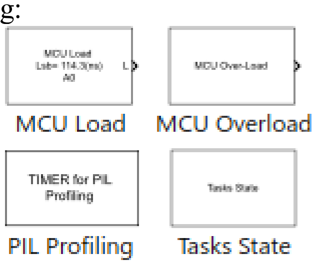

Figure 10: Profiling Module [1]

- MCU Load: Reports the execution time of the algorithm in real-time.
- MCU Overload: Reports algorithm overload in real-time.
- PIL Profiling: Lists executed tasks and the required execution time in the algorithm.
- Tasks State: Selects the priority task for execution.

2.4 Based modules in Simulink

Combining MPLAB Device Blocks modules with based Simulink blocks such as Source, Gain, PID, Discrete, and Subsystem helps clearly express control ideas and makes debugging easier if errors occur.

3. Experimental a BUCK chopper circuit control by graphical programming method

The hardware of the chopper circuit designed for teaching graphical programming has been implemented on a test board as an illustrative example, as shown in Figure 11. The output load is a 1KΩ resistor. The control circuits use an N-FET connected to ground and a P-FET connected to an isolated power source through an optocoupler. The voltage feedback divider circuit is connected to pins, allowing integration with different DSP platforms.

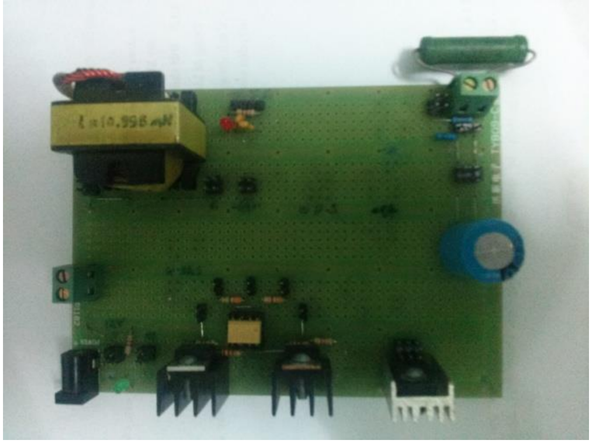

Figure 11: Chopper circuit hardware on the test board

In this paper, a Buck chopper circuit [3] will be used to step down the voltage from 12V to 5V, along with a dsPIC 30F4011 controller, to demonstrate the graphical programming method for real-time control. The detailed component connection diagram is shown in Figure 12.

After assembling the hardware, the control concept is programmed in MATLAB/Simulink, as shown in Figure 13. Once completed, the program is compiled and automatically loaded into the dsPIC.

The operator can fine-tune and observe waveforms at desired points on the PC screen via UART communication instead of using a physical oscilloscope. Figure 14 shows the output waveform corresponding to the PID controller in real-time with sampling time 1ms.

4. Conclusions

This paper provides detailed information and explanations on the graphical programming tool based on MATLAB for Microchip's dsPIC series. The functions and usage of the MPLAB Device Block for Simulink modules are clearly explained.

A specific Buck chopper circuit was then controlled to illustrate the concept of combining blocks to implement a PID control algorithm embedded in the dsPIC 30F4011. The control parameters and output signals were displayed directly on a PC screen via UART, replacing the need for an expensive oscilloscope.

The results of this paper can be used for further in-depth research, as academic teaching material for related courses, or as a source of scientific inspiration.

Acknowledgements

I would like to express my sincere gratitude to my family, institution, and colleagues for their continuous support, encouragement, and facilitation of my scientific research.

References

Online Document of Microchip Company (2023), http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=SW007023

Online Tool of Microchip Company (2024), http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=SW007023

Hartono Darmawaskita, "AN 216: DC/DC Converter Controller Using A PICmicro Microcontroller", *Application note of Microchip Technology Inc*.

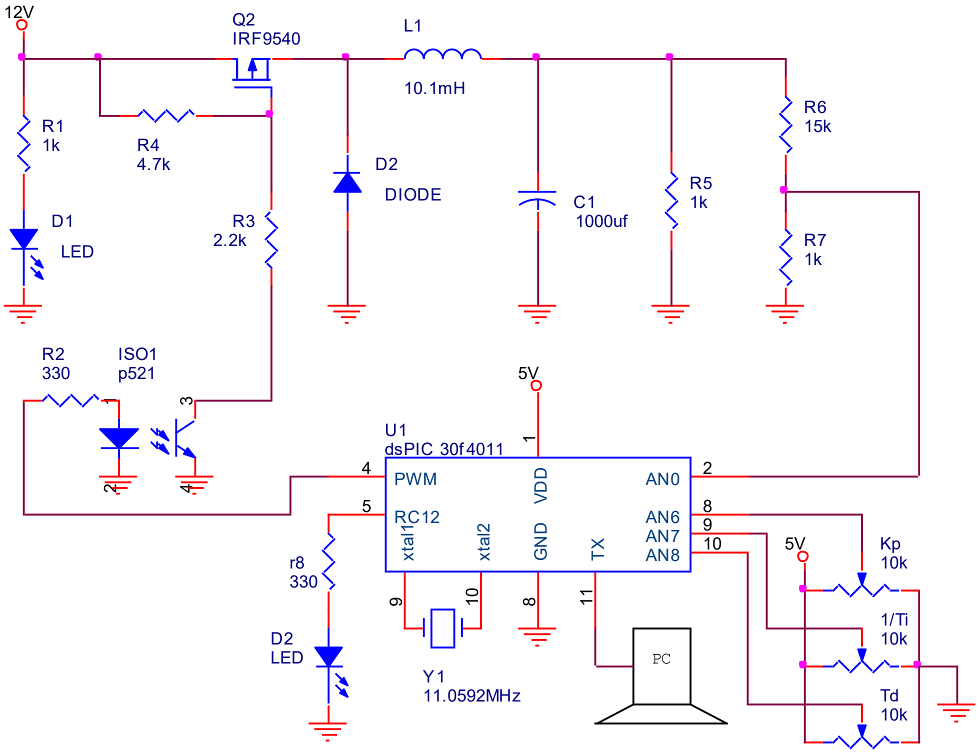

Figure 12: Buck chopper circuit control connection diagram

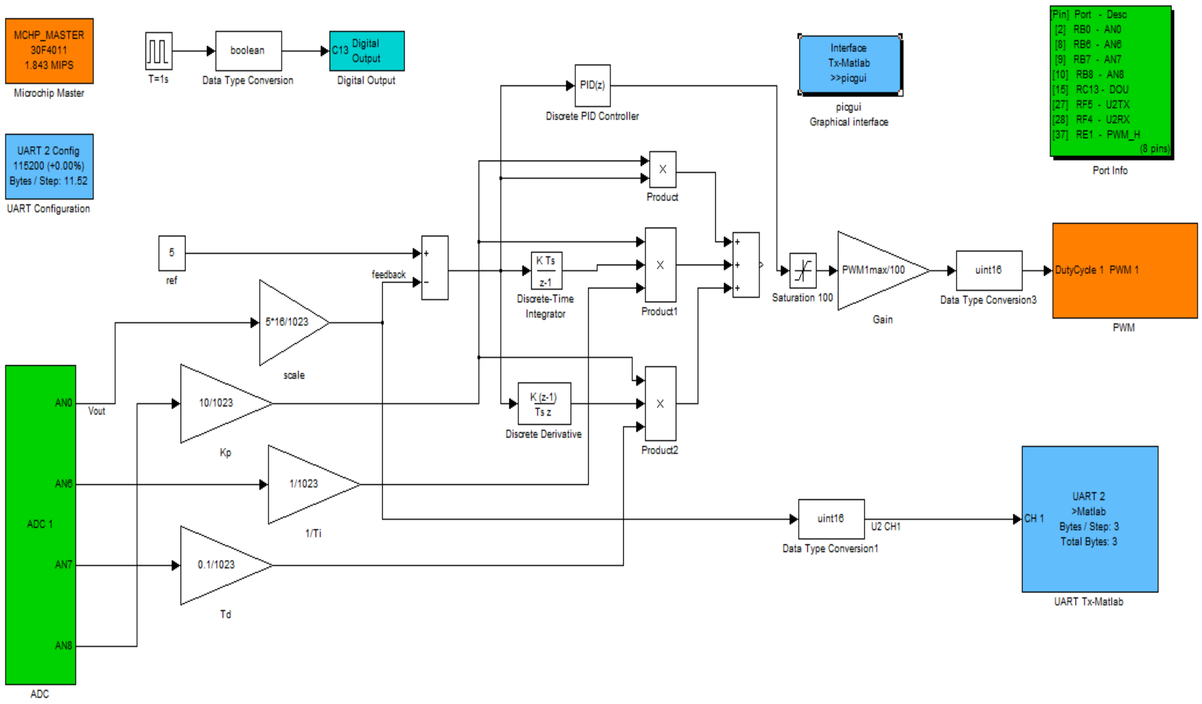

Figure 13: Graphical programming diagram for Buck circuit control in Simulink

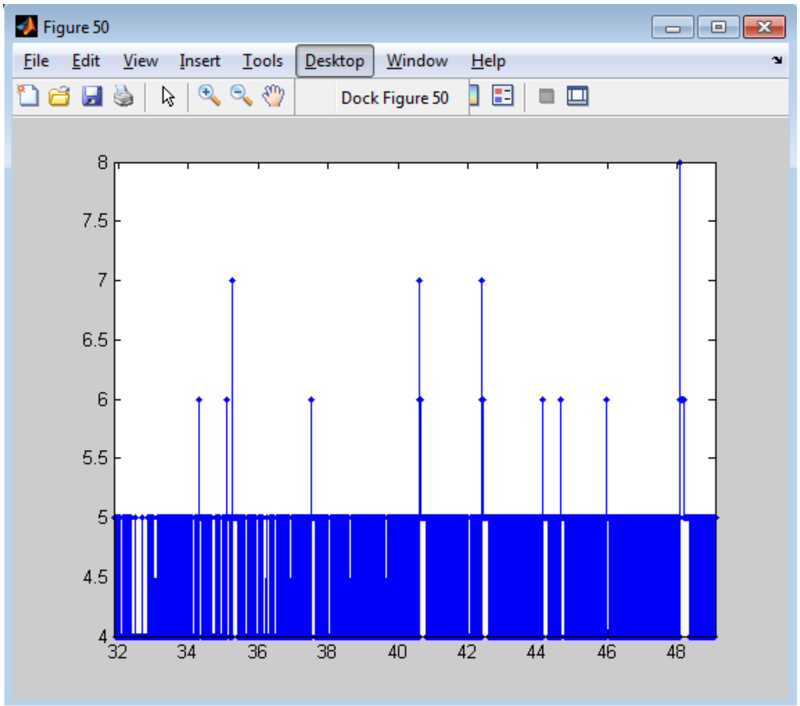

Figure 14: Simulink Real-time output voltage waveform with proportional-integral-derivative (PID) controller