dsPIC30f4012 Microcontroller

dsPIC30f4012 is manufactured by Microchip, and is about $6

Specifications:

- 28-pin, 16-bit microcontroller
- 24-bit wide instructions
- 16-bit wide data path
- 48 Kbytes on-chip flash program space
- Up to 30 MIPs operation
- 20 I/O pins
- 30 interrupt sources
- 16-bit Capture input functions
- 16-bit Compare/PWM output functions
- 2 UART modules with FIFO Buffers
- 1 CAN modules, 2.0B compliant
- 6 PWM output channels
- Trigger for A/D conversions

Parameter Name	Value
Architecture	16-bit
CPU Speed (MIPS)	30
Memory Type	Flash
Program Memory (KB)	48
RAM Bytes	2,048
Temperature Range C	-40 to 125
Operating Voltage Range (V)	2.5 to 5.5
I/O Pins	20
Pin Count	28
System Management Features	PBOR, LVD
POR	Yes
WDT	Yes
Internal Oscillator	7.37 MHz, 512 kHz
nanoWatt Features	Fast Wake/Fast Control
Digital Communication Peripherals	1-UART, 1-SPI, 1-I2C
Codec Interface	NO
Analog Peripherals	1-A/D 6x10-bit @ 1000(ksps)
CAN (#, type)	1 CAN
Capture/Compare/PWM Peripherals	4/2
Motor Control PWM Channels	6
Quadrature Encoder Interface (QEI)	1
Timers	5 x 16-bit 2 x 32-bit
Parallel Port	GPIO
Hardware RTCC	No
DMA	0

Getting Started with dsPIC30F Microcontroller

APPLICATIONS:

- **Motor Control**
  - Brushless DC, AC Induction and Switch Reluctance motors
- **Power Conversion and Monitoring**
  - Power conversion, inverters, and uninterruptible power supplies (UPS)
- **Internet Connectivity**
  - Ethernet and modem applications for Internet connectivity with TCP/IP
- **Speech and Audio**
  - Noise and echo cancellation, speech recognition and speech playback
- **Sensor Control**
  - Advanced sensor control and smart sensor interface modules
- **Automotive**
  - Certified high-temperature grades parts for automotive applications

For more detail see:

- Getting Started with dsPIC30F Digital Signal Controllers Users Guide
  - ww1.microchip.com/downloads/en/devicedoc/70151a.pdf

Programming dsPIC30F Microcontroller

Microchip development tools:

- **MPLAB IDE**
  - An integrated development environment (IDE) for C programming, debugging and uploading (free)
- **MPLAB ICD2 and PICkit**
  - In-circuit debugger tools that can be used for uploading programs into the dsPIC ()
- **Development boards**
  - Various development boards are available for different applications
    - dsPICDEM Starter Demo Boards
    - dsPICDEM General Purpose Development Boards
    - dsPICDEM Motor Control Development Systems
    - dsPICDEM Connectivity Development Boards
  - Other dsPIC30F controller boards:
    - **dsPIC30F4011 Controller Board: $33**
    - **dsPIC30F2010 Controller Board: $28**
      - www.futurlec.com/**dsPIC30F**4011_Controller_Tips.shtml

PICkit

dsPIC30F4012 chip

<u>For more detail see</u>:

- Getting Started with dsPIC30F Digital Signal Controllers Users Guide
  - ww1.microchip.com/downloads/en/devicedoc/70151a.pdf

Embedded Projects with dsPIC

- Professor Mark Csele's Home Page - Projects and Hobbies
  - 192.197.62.35/staff/mcsele/project.htm
- dsPIC programs and projects
  - www.baghli.com/dspic.php
- dsPIC-Servo Project - members.shaw.ca members.shaw.ca
  - members.shaw.ca/swstuff/dspic-servo.html
- Robots and PIC Projects by Bill and Mark Sherman
  - home.comcast.net/~botronics/robots.html
- Images for pic robotic projects - Report images
  - https://www.google.com/search?q=pic+robotic+projects&hl=en&client=firefox-a&hs=tz7&rls=org.mozilla:en-US:official&prmd=imvns&tbm=isch&tbo=u&source=univ&sa=X&ei=MN1MULzDAsW90QGz3oHYAQ&ved=0CCgQsAQ&biw=931&bih=706
- Pololu - PIC-Based, Obstacle-Avoiding Robot
  - www.pololu.com/docs/0J2
- RoverBot Project - Build a Robot - Robot Kit – PIC
  - www.midnightkite.com/RoverBot.html

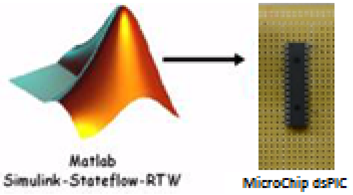
dsPIC Programming

Programming dsPIC in C

- Use MPLAB IDE to write the C code and compile it for the dsPIC
- Use MPLAB ICD2 to upload the compiled code to the dsPIC

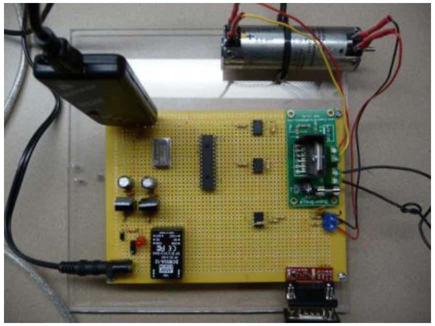
Programming dsPIC using Matlab products

- Embedded Target for dsPIC - A MATLAB toolbox for real-time control of dsPIC microcontrollers (Target Blockset for dsPIC, PIC24 and PIC32)
  - http://www.kerhuel.eu/RTWdsPIC/
  - The evaluation version is free, but allows only up to seven I/O pins



- SIM2LAB Rapid Embedded Solutions (Target Blockset for dsPIC33 family)
  - http://www.sim2lab.com/S2L_dsPIC33.html
  - The program is free
- MATLAB Device Blocks for MPLAB IDE
  - http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=SW007022
  - The demo version is free, but performs 30 sec wait before each program execution

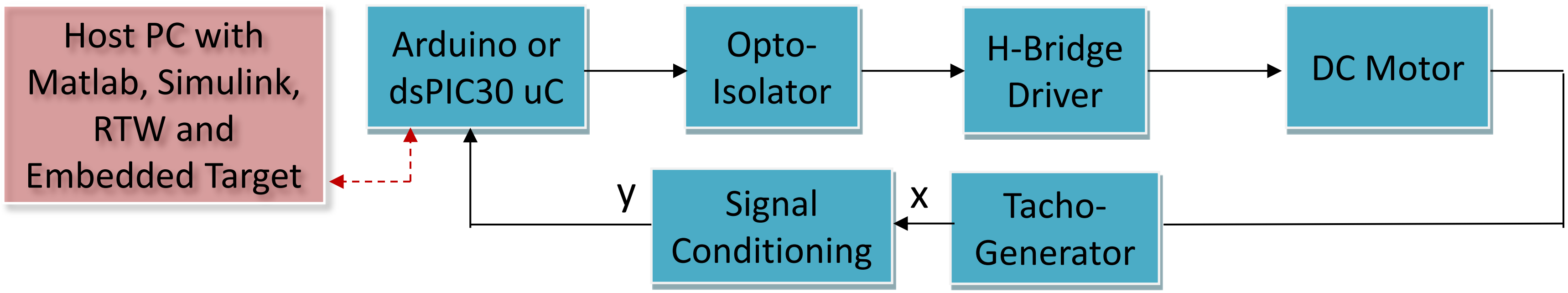
Embedded Control Lab Experimentation



Lab projects:

- The lab experiments are to be completed in groups of two in the Embedded Control Systems (ECS) lab (room E238)
  - Must get the TA's signature after completing each lab
- The lab projects involve modeling/identification and control of an integrated DC motor/tacho-generator, using two different microcontrollers
  - Arduino board (Atmega328p microcontroller)
  - dsPIC30F4012 microcontroller hardware
- Both lab experiments must follow the MAD (modeling, analysis, design) process, using Matlab, Simulink, and other Mathworks toolboxes
  - Details of the experiments are provided in the lab projects manuals
  - Approx. dc motor specs: +/- 4000 RPM, 12V, L=7mH, R=10 ohm, kt=3.45 oZ-in/A
  - Power-card specs: dc input 12V, dc outputs +5V, GND, +12V, 0V, -12V
- Submit a concise lab report for each experiment, including the collected data, data analysis, system identification, control design, simulations and plots, and hardware implementation and verification

Lab Project Set-Up



- The system consists of a dc motor, with an integrated tachometer (angular speed sensor), and is driven by an H-bridge driver
- The goal is to model and control the dc motor
- The system includes a microcontroller interfaced with the H-bridge driver and the tacho-generator, via an opto-isolator and a signal conditioning circuit
  - The opto-isolator is to protect the low-power processor from high-power motor
  - The signal conditioning circuit consists of a voltage divider and an op-amp, which is to scale and offset the measured speed to be compatible with the processor’s ADC

Signal Conditioning Circuit

- For an ADC reference voltage of 5v and a tachometer with maximum 10v output, one gets:

$$V_{REF}: V_y \in [0,5] volts; \quad V_{TACH}: V_x \in [-10,10] volts;$$

$$m = \frac{V_{ymax} - V_{ymin}}{V_{xmax} - V_{xmin}} = \frac{5}{20} = \frac{1}{4}; \quad b = V_{ymax} - mV_{xmax} = 5 - \frac{10}{4} = 2.5;$$

$$then: \ V_y = \frac{1}{4} V_x + 2.5$$

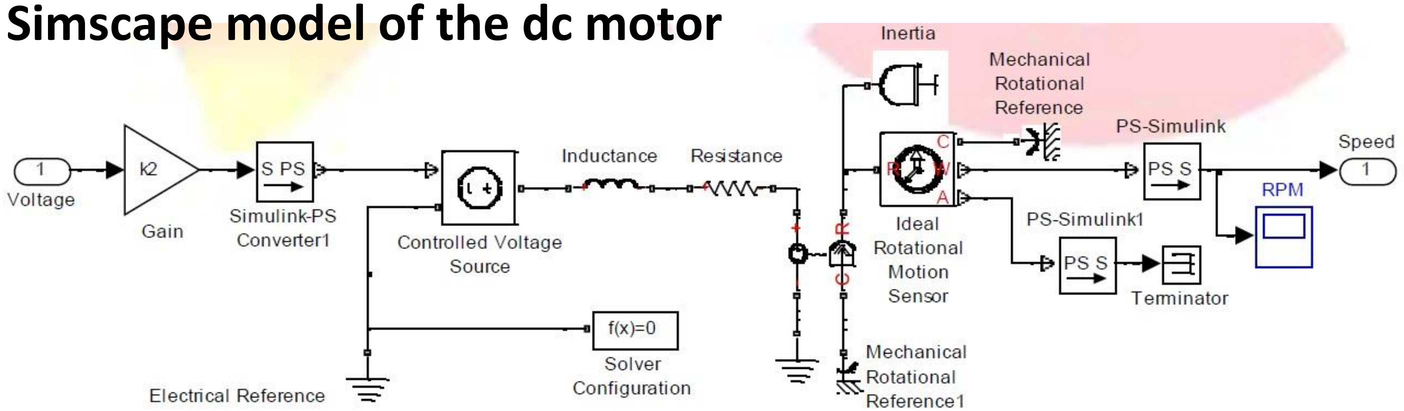
- The tacho signal needs to be quartered and given a 2.5v offset
- Inside the microcontroller, the inverse function must be used to determine the tachometer voltage from the ADC value:

$$V_y = \frac{1}{4} V_x + 2.5; \quad V_x = 4(V_y - 2.5)$$

Motor Control Modeling

- Tools for system modeling
  - System Identification Toolbox
    - Identify the system model (approx. linear model) from observed I/O data
      - The model includes motor, opto-isolator, H-bridge, tachometer, and signal conditioning circuit
  - Simulink Design Optimization Toolbox
    - Build a mathematical model of the system (possibly nonlinear) in Simulink or Simscape and adjust its parameter values to fit the I/O data
    - Linearize the nonlinear model and determine its transfer function, using Simulink Control Design Toolbox

Simscape model of the dc motor



Motor Control Design

- Tools for control design
  - Control System toolbox and Robust Control Toolbox
    - Design linear controllers for the linearized model of the system
      - Use SISOTool for lead-lag, PID, pole-placement control design
  - Simulink Design Optimization Toolbox
    - Design linear controllers for the Simulink or Simscape model of the system, using auto-tuning tools
      - For example PID auto-tuning
  - Optimization Toolbox
    - Design fixed-structure controller parameters by minimizing a cost
      - For example, minimize an ITAE cost to adjust the parameters of a lead-lag control

<u>See Simulink Design Optimization Toolbox example at:</u>

- Help → Simulink Design Optimization Toolbox → Demos → Rotational inverted pendulum

Control Implementation

- Arduino development board
  - Use Simulink and Arduino Target Blockset to write the program for I/O measurement, communication with the PC, via USB/serial port, and for uploading the code into Arduino

- dsPIC30f4012 microcontroller hardware
  - Use Simulink and Lubin Kerhuel's dsPIC Target Blockset to write the program for I/O measurement and uploading the code into dsPIC
    - MPLAB IDE must be installed for use by the dsPIC Target Blockset
    - MPLAB ICD2 must be available for program download via USB
  - Use Real-Term freeware for communication with the PC, via USB/serial port, for monitoring