ANADOLU UNIVERSITY

DEPT. OF ELECTRICAL & ELECTRONICS ENGINEERING

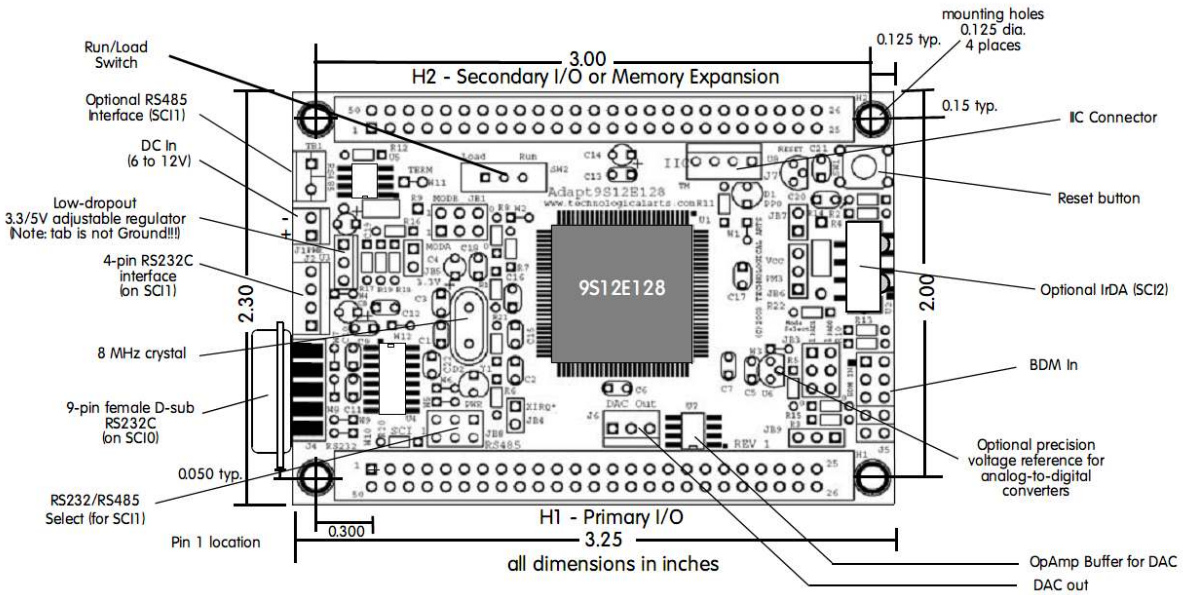
EEM489 MICROPROCESSORS II LABORATORY

LAB 3: APPLICATION SAMPLES

Oct 26th, 2009

TA: ŞÜKRÜ GÖRGÜLÜ

In the previous lab, you got familiar with CW development environment. This time you will learn how to control some more peripherals such as Analog-to-Digital Converter (ADC). Using ADC capability of your module, you will control a LED array and a sound buzzer over a potentiometer, which are all located on a demo-card that is connected to Adapt9s12e via the H1 expansion port.



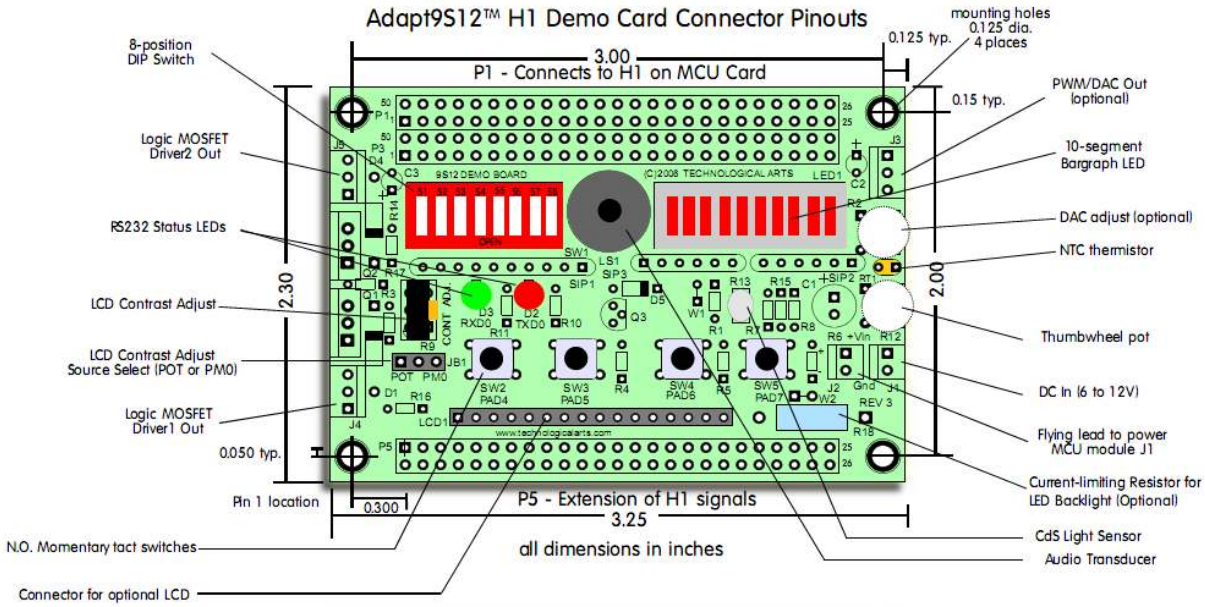
Adapt9S12E128™ CONNECTOR PINOUTS

As you can see H1 Demo Card is connected to Adapt9s12 module with H1 port which includes 50 pins within itself. Pin signals are tabulated in manuals which are located in folder '`eem489\E128\Adapt9S12E128Docs`'.

- **Adapt9s12 H1 Demo Card**

Demo card includes many components such as a dip-switch array, an audio transducer, a bar-graph LED array, a NTC thermistor, a thumbwheel pot, a CdS light sensor, an LCD control unit, and etc…

Manuals of 'Adapt9s12 H1 demo card' can be downloaded from web:
http://support.technologicalarts.ca/docs/Adapt9S12E/Demo_Card/



- **PART 1: Reading The Value Of The Pot On DemoCard**

In the first part of the experiment, you will need to read the value of the thumbwheel potentiometer (pot), and send it to the 10-segment LED array. The value of the pot can be measured by the Analog-to-Digital Converter (ADC) module. However, there are sixteen 10-bit ADC channels (PAD0/KWAD0 to PAD15/KWAD15). The pot is attached to one of these ADC channels thru the corresponding pin of the H1 port. Find it from the manuals, and detect which ADC channel will be used in ADC conversion. You will use a bean to configure ADC. Add it to your c project (Bean Selector → START → Measurement) and make necessary changes. Usage of ADC module is described in HELP pages of the processor expert for ADC bean. Check help documents of CW for detailed information. Typical usage of ADC bean is described there. You can set the conversion time to 20 microseconds (us).

You are expected to use ADC interrupt service to read the corresponding specific channel and put the value of the channel to the LED array. Hence, measured digital value of the pot will be visible upon the LED array.

You will use another bean to control I/O port which the LED array is connected to. Find the port from the manuals and make necessary changes in the bean.

Make your project, load and run. See the results.

- **PART 2: Generating Sound with the Audio Transducer**

Audio signals are generated as a mix of periodic signals which have frequencies in the range from 20 Hz to 20 KHz. To hear a pure audio signal, you will have to generate a square-wave.

In the second part of experiment, you will generate square waves with timer interrupt. Add a periodic timer interrupt bean to your project. Set its frequency to 1 KHz. In the timer interrupt, you will toggle the pin which the audio buzzer (SPKR) is connected-to. Find the corresponding signal-pin from the manuals, add another a bean to control that pin, and write the necessary code into the timer interrupt function. Remake your project, load it and run it. Use can use an oscilloscope to see the generated signal by probing the scope over the corresponding H1 pin holes.

(Note: Buzzer needs a bit more power than the LEDs. Hence, you may have to feed it with a change of power connection. Ask your training assistant for the additional power cable)

- **PART 3: Changing The Frequency Of The Timer Depending on the POT: A Real-time App.**

In this part, you will control the period of the timer interrupt with pot. However, you will have to use some low-level modifications here, because the timer interrupt bean doesn't have an enabled function to chance the period within the code. Think about it and find a way to change the period of your timer interrupt thru the program according to the value of the pot readings from ADC module.

The pot is expected to change the frequency of the square wave from 1000 Hz to 4000 Hz. Do the necessary calculations and add the code to change the period of the timer depending on the value of the pot.