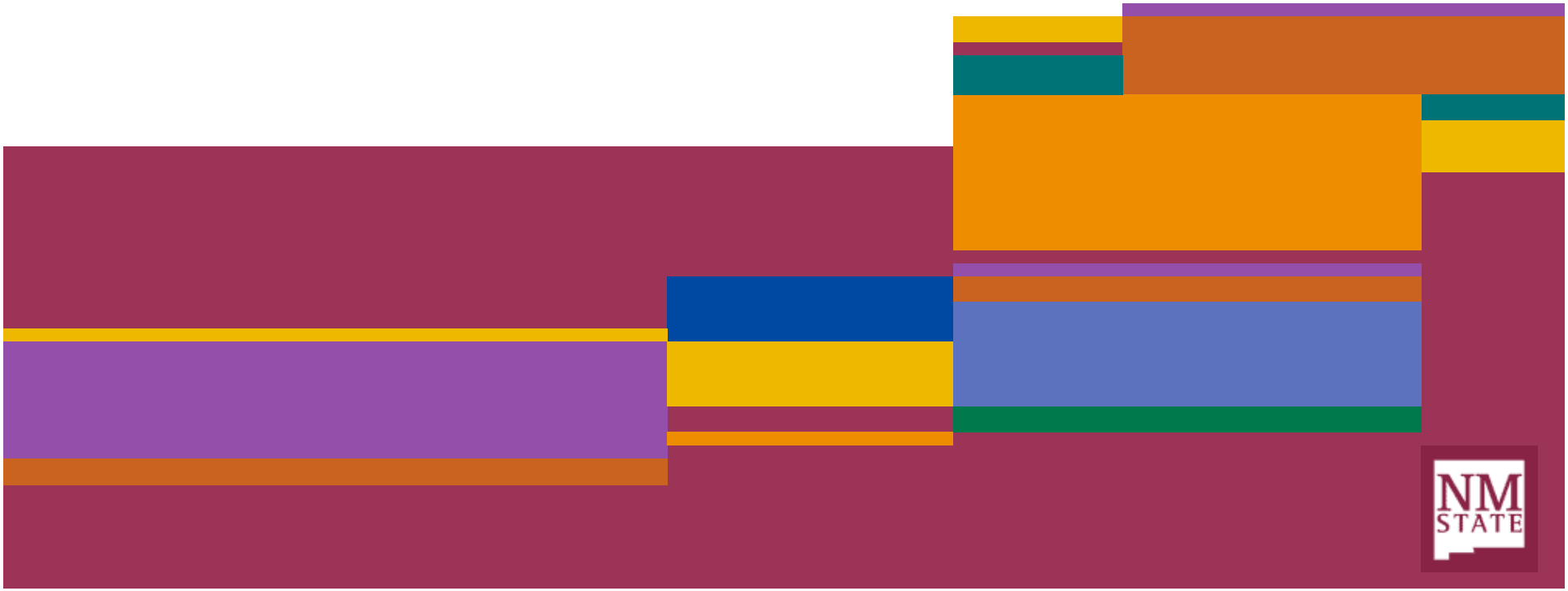
Generalized Extension Language (GEL) and Applications



References

- More detailed information on GEL can be found in Chapter 12 of the Code Composer Studio (CCS) User's Guide, application note **SPRU328B.pdf** (see course webpage)

- Additional information can be found in CCS's help system
  - Help → Contents → Creating Code and Building Your Project → Automating Tasks with the General Extension Language

Introduction

- General Extension Language (GEL) is an interpretive language similar to C that lets you create functions to extend CCS's capabilities
  - Functions are created using GEL syntax and separately loaded into CCS
- With GEL, we can create a Graphical User Interface (GUI) that can displays a window on the host PC
  - GUI: sliders, dialog boxes, menu actions, output windows
  - Memory can be accessed in real-time using GEL
- Actions to the GUI can be used to control the DSK

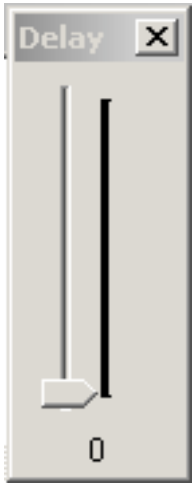
Example: Real-Time Echo with Selectable Delay

- As a simple example on the use of GEL for creating a GUI, consider the following difference equation

$$y[n] = 0.5x[n] + 0.5x[n-D]$$

  which implements an echo

- We would like to make the echo delay, D a variable and control it (real-time) using a GUI slider
  - We also assume that $0 \leq D \leq$ MAX_DELAY



user_data.h

```
<SNIP>

/* Global variables here as extern, initializations in
initialize_program.c */

#define MAX_DELAY 32767 /* buffer length, max should be 32767 */
extern short buffer[];  /* buffer to store samples */
extern short *OldestSamplePtr;   /* ptr to oldest sample */
extern short delay;       /* desired echo delay */

#include "DSPFunctionsFixedPoint/util.h"

<SNIP>
```

initialize_program.c

```
short buffer[MAX_DELAY];   /* buffer to store samples */
short *OldestSamplePtr;    /* ptr to oldest sample in buffer */
short delay;               /* echo delay */

void initialize_program()
{
    short i;

    delay = 0;                  /* initial echo delay in samples*/
    OldestSamplePtr = buffer; /* setup ptr clear buffer out */
    for(i=0;i<MAX_DELAY;i++)
        buffer[i] = 0;
}
```

process_signal.c

```
void process_signal(short inputRight, short inputLeft, short
*outputRight, short *outputLeft)
{
    short delay_sample;

    // Process right channel sample
    *OldestSamplePtr = inputRight;

    delay_sample=tap((MAX_DELAY-1),buffer,OldestSamplePtr,delay);

    *outputRight = inputRight/2 + delay_sample/2;

    cdelay((MAX_DELAY-1), buffer, &OldestSamplePtr);

    // Process left channel sample
    *outputLeft = inputLeft;
}
```

echo.gel

```
menuitem "Echo"; // GEL menu item called Echo
// min, max, increment, skip, variable name
slider Delay(0, 32766, 100, 1000, d)
{
        // delay is a global variable
        delay = d;
}
```

Steps

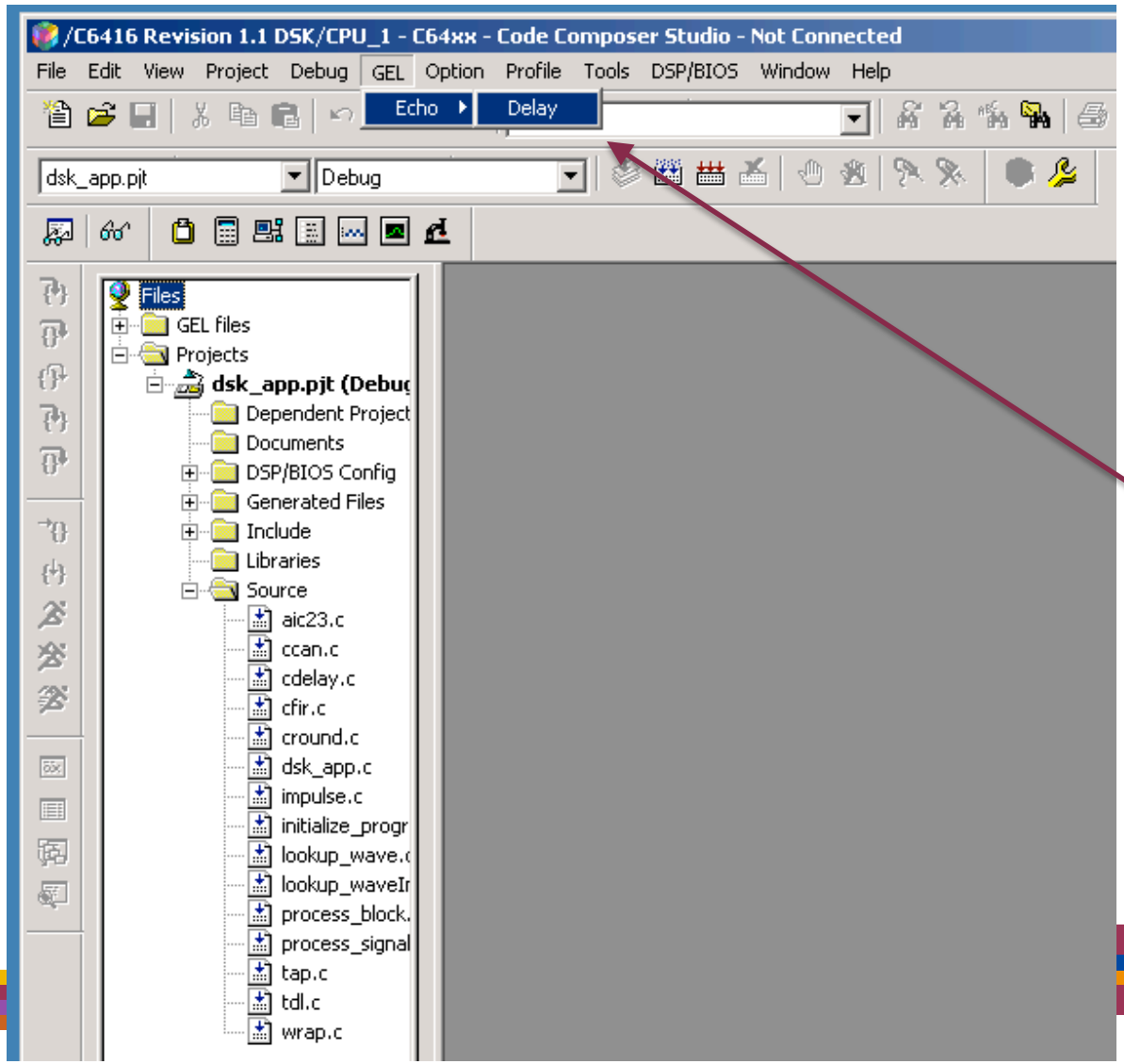
1) Build the project as usual and download to target H/W

2) Load the GEL file (File -> Load GEL...)

3) Display the slider (GEL -> Echo -> Delay)

4) Run the program

Displaying the Delay Slider



Delay Slider