



DesignNews

IoT Device Prototyping with Microchip Curiosity Development Boards

Day 1:

Curiosity Firmware Development Using the CCS C Compiler

Sponsored by



Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.



Fred Eady

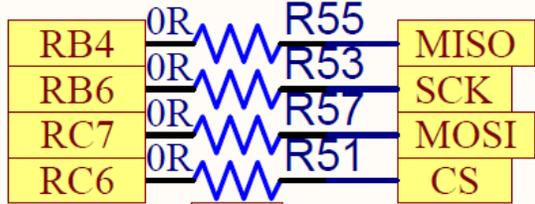
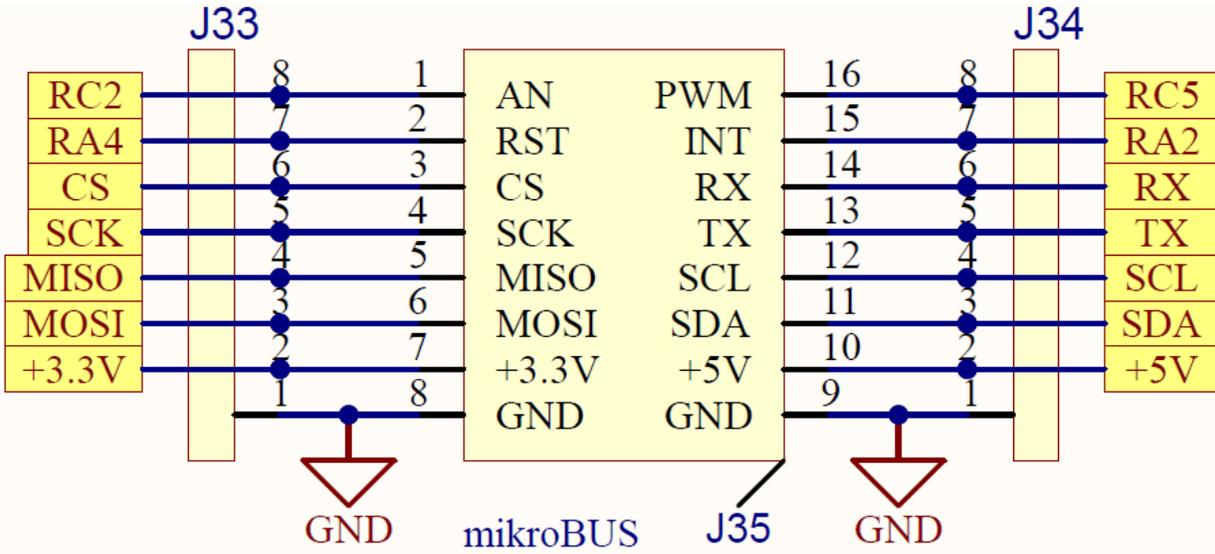
Visit 'Lecturer Profile' in your console for more details.

AGENDA

- **Coding with the CCS C Compiler's Magic Wand**
- **High Endurance Flash**

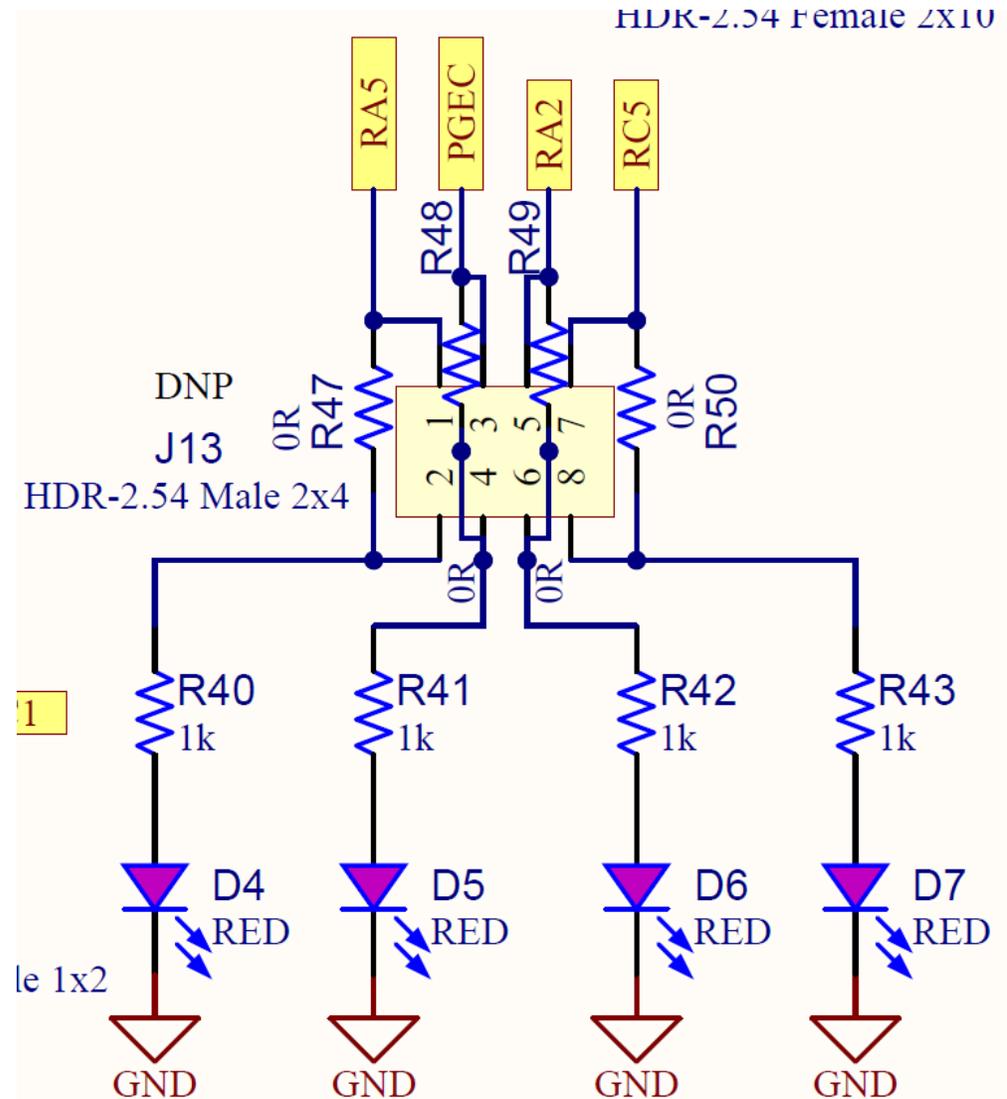


Bricks in the Wall

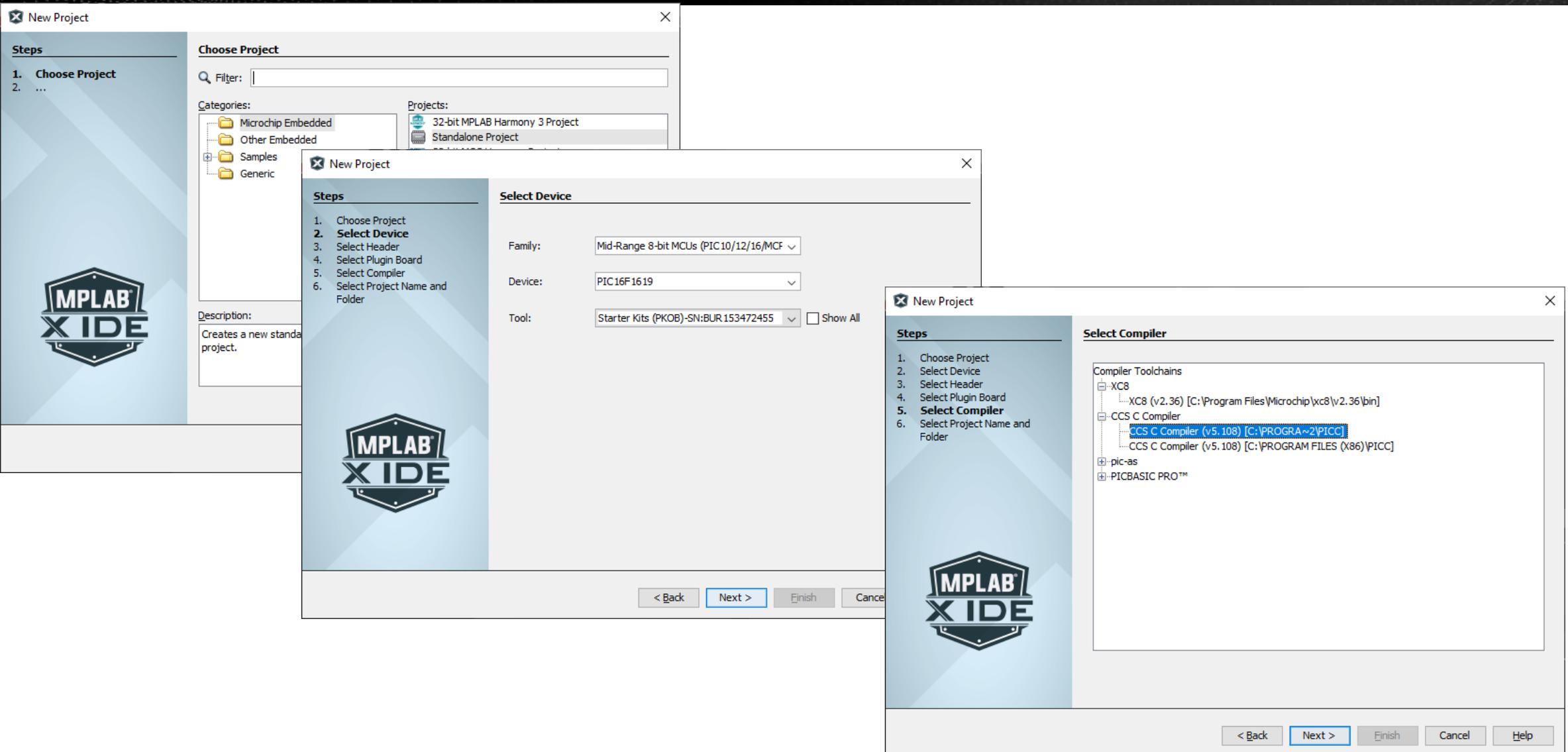


J39 HDR-2.54 Male 2x4

J40 HDR-2.54 Male 2x4



EUSART the CCS Way – Create the MPLABX Project



New Project

Steps

1. Choose Project
2. ...

Choose Project

Filter:

Categories:

- Microchip Embedded
- Other Embedded
- Samples
- Generic

Projects:

- 32-bit MPLAB Harmony 3 Project
- Standalone Project

Description: Creates a new standard project.

MPLAB X IDE

New Project

Steps

1. Choose Project
2. **Select Device**
3. Select Header
4. Select Plugin Board
5. Select Compiler
6. Select Project Name and Folder

Select Device

Family: Mid-Range 8-bit MCUs (PIC10/12/16/MCF)

Device: PIC16F1619

Tool: Starter Kits (PKOB)-SN: BUR153472455 Show All

< Back Next > Finish Cancel

New Project

Steps

1. Choose Project
2. Select Device
3. Select Header
4. Select Plugin Board
5. **Select Compiler**
6. Select Project Name and Folder

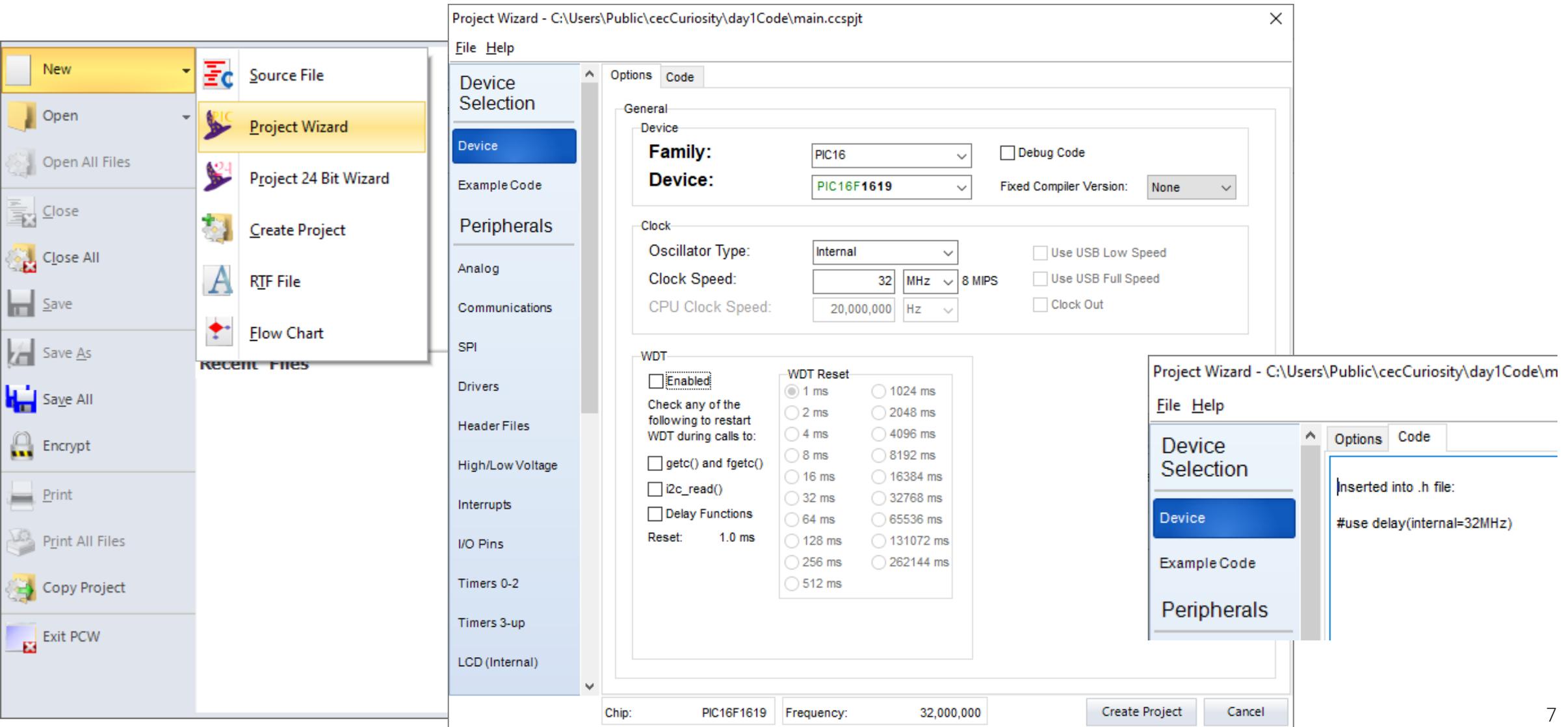
Select Compiler

Compiler Toolchains

- XC8
 - XC8 (v2.36) [C:\Program Files\Microchip\xc8\v2.36\bin]
- CCS C Compiler
 - CCS C Compiler (v5.108) [C:\PROGRA~2\PICC]**
 - CCS C Compiler (v5.108) [C:\PROGRAM FILES (X86)\PICC]
- pic-as
- PICBASIC PRO™

< Back Next > Finish Cancel Help

EUSART the CCS Way – Create the CCS Project



The screenshot displays the CCS Project Wizard interface. The main window is titled "Project Wizard - C:\Users\Public\cecCuriosity\day1Code\main.ccsproj". The "Device Selection" tab is active, showing the following configuration:

- Family:** PIC16
- Device:** PIC16F1619
- Fixed Compiler Version:** None
- Oscillator Type:** Internal
- Clock Speed:** 32 MHz (8 MIPS)
- CPU Clock Speed:** 20,000,000 Hz
- WDT:** Enabled, 1.0 ms Reset
- WDT Reset:** 1 ms (selected), 2 ms, 4 ms, 8 ms, 16 ms, 32 ms, 64 ms, 128 ms, 256 ms, 512 ms, 1024 ms, 2048 ms, 4096 ms, 8192 ms, 16384 ms, 32768 ms, 65536 ms, 131072 ms, 262144 ms

The "Code" tab is also visible, showing the following code snippet:

```
Inserted into .h file:  
#use delay(internal=32MHz)
```

The "Device Selection" sidebar on the left lists various hardware components: Device, Example Code, Peripherals, Analog, Communications, SPI, Drivers, Header Files, High/Low Voltage, Interrupts, I/O Pins, Timers 0-2, Timers 3-up, and LCD (Internal).

The "Recent Files" sidebar on the left lists: New, Open, Open All Files, Close, Close All, Save, Save As, Save All, Encrypt, Print, Print All Files, Copy Project, and Exit PCW.

The "Project Wizard" sidebar on the left lists: Source File, Project Wizard, Project 24 Bit Wizard, Create Project, RTF File, and Flow Chart.

The "Project Wizard" sidebar on the right lists: Device Selection, Example Code, and Peripherals.

The "Project Wizard" sidebar on the bottom lists: Chip: PIC16F1619, Frequency: 32,000,000, Create Project, and Cancel.

EUSART the CCS Way - With a little bit of CCS I²C

Project Wizard - C:\Users\Public\cecCuriosity\day1Code\main.ccsproj

File Help

Device Selection

Options Code

Communications

Use RS-232

Port Count: 1 (selected) 2 3 4

Type: UART1 (selected) Standard RS232 (selected) RS485

Restart WDT on RS232

Baud: 9600 Parity: None Invert Float_high Errors External interrupt

Transmit Pin: B7 Receive Pin: B5

Enable Pin: None Receive Enable Pin: None

Bits: 8 Buffer Size: 0 Stream: PORT1 ID: 10

I²C

Use I²C

SDA: B4 SCL: B6

Master (selected) Slave Fast (selected) Slow

Restart WDT on I²C Force Hardware

Slave Address:

Chip: PIC16F1619 Frequency: 32,000,000

Create Project Cancel

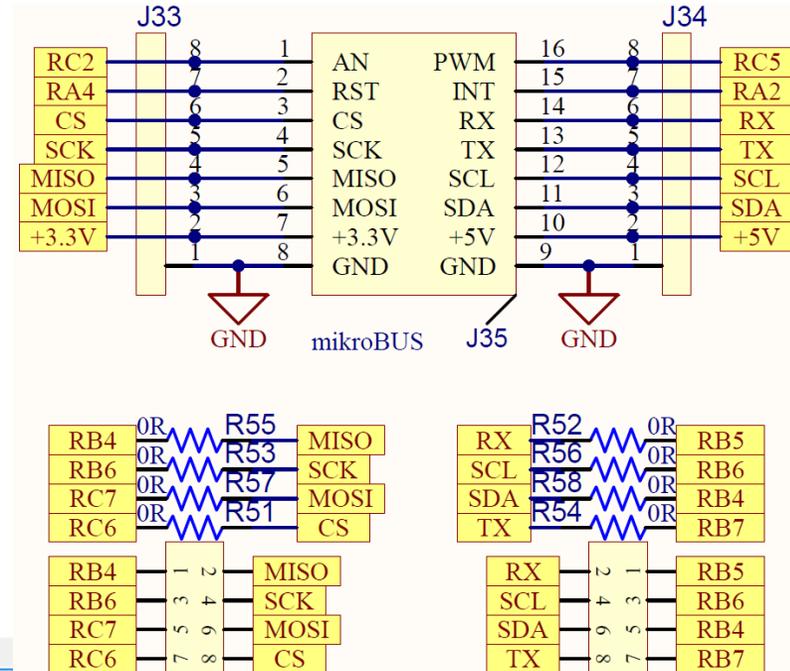
Users\Public\cecCuriosity\day1Code\main.ccsproj

Options Code

Inserted into .h file:

```
#pin_select U1TX = PIN_B7
#pin_select U1RX = PIN_B5

#use rs232(baud=9600,parity=N,xmit=PIN_B7,rcv=PIN_B5,bits=8,stream=PORT1)
#pin_select SDA1=PIN_B4
#pin_select SCL1=PIN_B6
#use i2c(Master,Fast,sda=PIN_B4,scl=PIN_B6,force_hw)
```



Chip: J39 HDR-2.54 Male 2x4 J40 HDR-2.54 Male 2x4

Create Project Cancel

EUSART the CCS Way – Create the CCS Project

The image shows three sequential screenshots of the CCS Project Wizard's 'Fuses' configuration screen. The first screenshot shows the 'Options' tab with 'Add Default Fuses to Header File' checked. The second screenshot shows the 'Fuses' section with 'Include Fuses' checked and various options like 'Master Clear pin enabled' and 'Fail-safe clock monitor enabled' selected. The third screenshot shows the resulting C code for the fuses, including definitions for #FUSES MCLR, #FUSES NOBROWNOUT, #FUSES FCMEN, #FUSES NOWRT, #FUSES PPS1WAY, #FUSES ZCDDIS, #FUSES STVREN, #FUSES BORV19, #FUSES LVP, #FUSES WDTSW, #FUSES WDTWIN_SW, and #FUSES WDTCLK_SW.

Project Wizard - C:\Users\Public\cecCuriosity\day1Code\main.ccsproj

File Help

Options Code

Options

Function Generation

Opening brace on the following line

Opening brace on the same line

Restart WDT during calls to DELAY

One fuse per line with comments

Add Default Fuses to Header File

Timers 0-2

Timers 3-up

LCD (Internal)

LCD (External)

Capacitive Touch

RTCC

CCP/Vref

Comparator

USB

TCP/IP

MODBus

CANBus

Bootloader

Advanced

Options

Fuses

Chip: PIC16F1619 Frequency: 32,000,000 Create Project Cancel

Project Wizard - C:\Users\Public\cecCuriosity\day1Code\main.ccsproj

File Help

Options Code

Fuses

Include Fuses

Power Up Timer

Master Clear pin enabled

Code protected from reads

No brownout reset

Output clock on OSC2

Internal External Switch Over mode enabled

Fail-safe clock monitor enabled

Program memory not write protected

Allows only one reconfiguration of peripheral pins

Zero-cross detect circuit is disabled at POR

Stack full/underflow will cause reset

Brownout reset at 1.9V

Low-Power Brownout reset is enabled

Low Voltage Programming on B3(PIC16) or B5(PIC18)

Watch Dog Timer Postscale settable in software

Watchdog Window is settable in software

WDT clock source settable in software

Chip: PIC16F1619 Frequency: 32,000,000 Create Project Cancel

Project Wizard - C:\Users\Public\cecCuriosity\day1Code\main.ccsproj

File Help

Options Code

Inserted into .h file:

```
#FUSES MCLR //Master Clear pin enabled
#FUSES NOBROWNOUT //No brownout reset
#FUSES FCMEN //Fail-safe clock monitor enabled
#FUSES NOWRT //Program memory not write protected
#FUSES PPS1WAY //Allows only one reconfiguration of peripheral pins
#FUSES ZCDDIS //Zero-cross detect circuit is disabled at POR
#FUSES STVREN //Stack full/underflow will cause reset
#FUSES BORV19 //Brownout reset at 1.9V
#FUSES LVP //Low Voltage Programming on B3(PIC16) or B5(PIC18)
#FUSES WDTSW //Watch Dog Timer Postscale settable in software
#FUSES WDTWIN_SW //Watchdog Window is settable in software
#FUSES WDTCLK_SW //WDT clock source settable in software
```

Chip: PIC16F1619 Frequency: 32,000,000 Create Project Cancel

EUSART the CCS Way – Code the CCS Firmware

```

void main()
{
    enable_interrupts(INT_RDA);
    enable_interrupts(GLOBAL);

    while(TRUE)
    {
        if(CharInRing())
        {
            output_toggle(LED_D6);
            ringBite = readring();
            printf("%c",ringBite);
        }
    }
}

//*****
//* RCVR BUFFER VARIABLES
//* USART RECEIVE BUFFERS SETUP
//* 1,2,4,8,16,32,64,128 or 256 BYTES
//*****
#define UART1_RX_BUFFER_SIZE      256
#define UART1_RX_BUFFER_MASK ( UART1_RX_BUFFER_SIZE - 1 )

uint8_t  UART1_RxHead;
uint8_t  UART1_RxTail;
uint8_t  UART1_RxBuf[UART1_RX_BUFFER_SIZE];
uint8_t  tmphead;
uint8_t  tmptail;
char rxBite;
char ringBite;

#INT_RDA
void RDA_isr(void)
{
    rxBite = getc();
    // calculate buffer index
    tmphead = ( UART1_RxHead + 1 ) & UART1_RX_BUFFER_MASK;
    // store new index
    UART1_RxHead = tmphead;

    if ( tmphead == UART1_RxTail)
    {
        UART1_RxHead = 0;
        UART1_RxTail = 0;
        return;
    }
    // store received data in ring buffer

    UART1_RxBuf[tmphead] = rxBite;
}

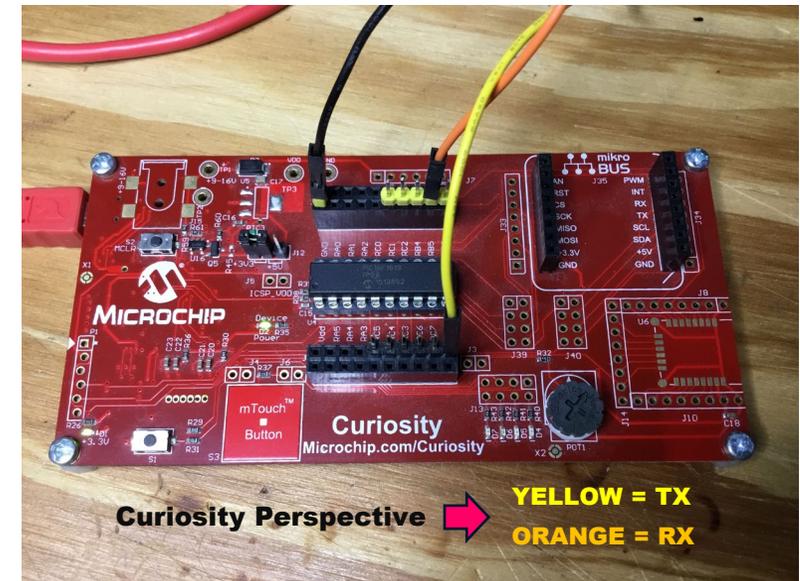
```

```

//*****
//* CHECK FOR CHARACTER IN RING
//*****
uint8_t CharInRing(void)
{
    return(UART1_RxHead != UART1_RxTail);
}

//*****
//* GET BYTE FROM RX RING BUFFER
//*****
uint8_t readring(void)
{
    // calculate buffer index
    tmptail = ( UART1_RxTail + 1 ) & UART1_RX_BUFFER_MASK;
    // store new index
    UART1_RxTail = tmptail;
    return UART1_RxBuf[tmptail];
}

```



EUSART the CCS Way – Execute the CCS Code



I²C the CCS Way – Code the CCS Firmware

```

void main()
{
    init_ext_eeprom();
    write_ext_eeprom(0x0000,0xAA);
    delay_ms(10);
    i2cBite = read_ext_eeprom(0x0000);
    if(i2cBite == 0xAA)
    {
        printf("%x",i2cBite);
        output_high(LED_D6);
    }
    while(1);
}

#define EEPROM_ADDRESS int
#define EEPROM_SIZE 2048
char i2cBite;

void init_ext_eeprom()
{
    output_float(PIN_B4); //SDA
    output_float(PIN_B6); //SCL
}

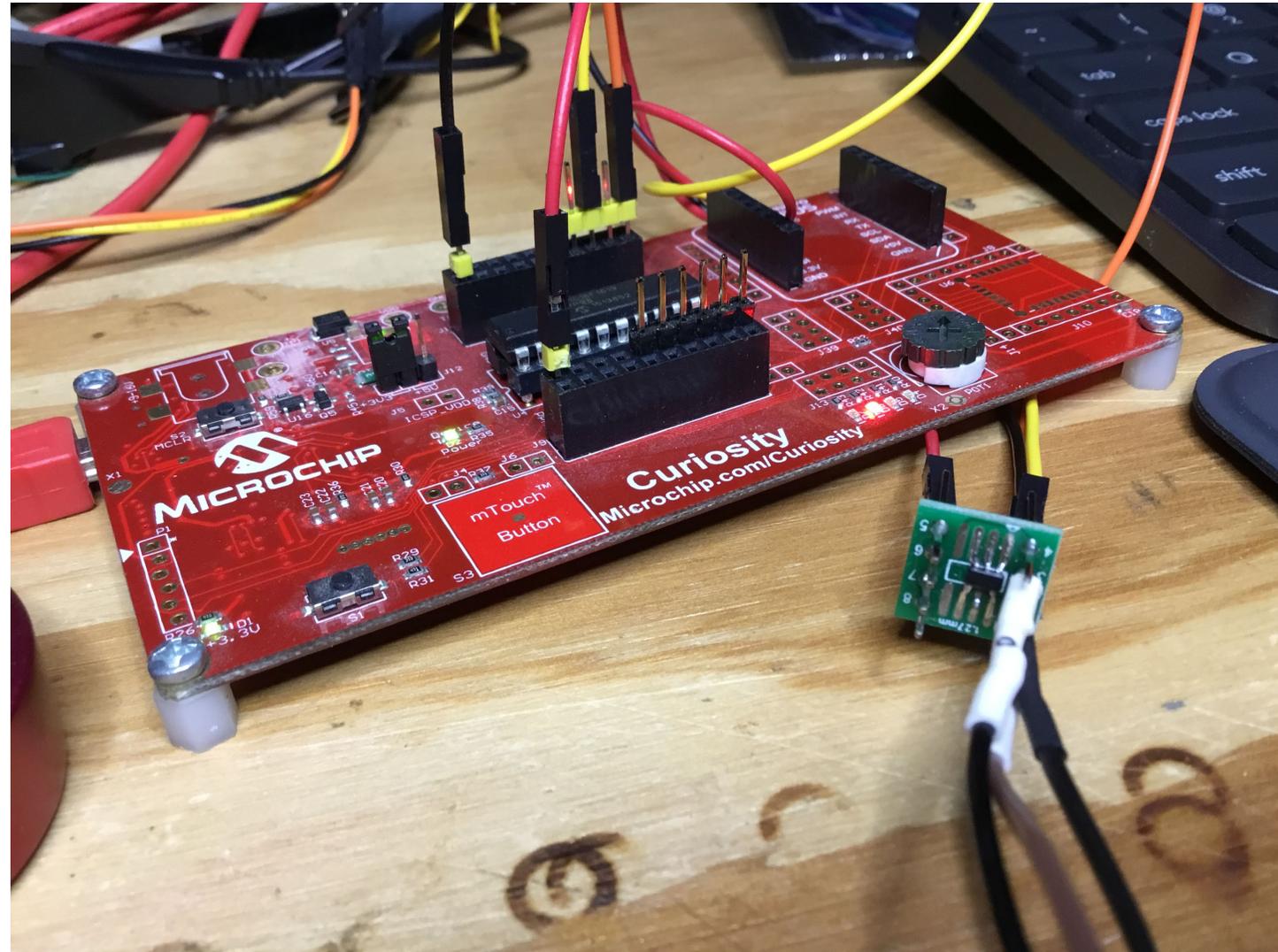
void write_ext_eeprom(int address, BYTE data)
{
    i2c_start();
    i2c_write(0xa0);
    i2c_write(address>>8);
    i2c_write(address);
    i2c_write(data);
    i2c_stop();
}

```

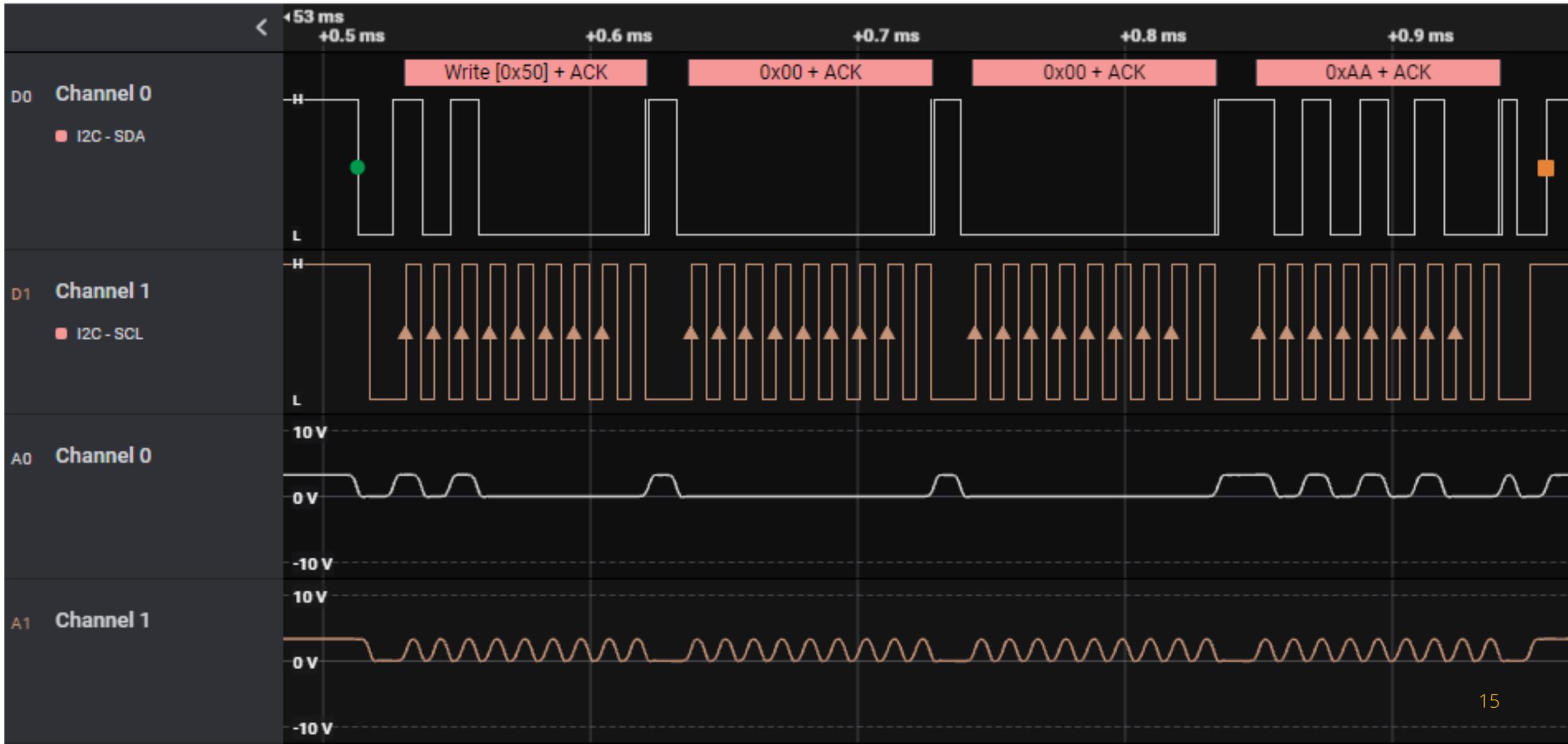
```

BYTE read_ext_eeprom(int address)
{
    BYTE data;
    i2c_start();
    i2c_write(0xa0);
    i2c_write(address>>8);
    i2c_write(address);
    i2c_start();
    i2c_write(0xa1);
    data=i2c_read(0);
    i2c_stop();
    return(data);
}

```



I²C the CCS Way – Write EEPROM



I²C the CCS Way – Read EEPROM



High Endurance Flash – Select the XC8 Compiler



New Project

Steps

1. Choose Project
2. Select Device
3. Select Header
4. Select Plugin Board
5. **Select Compiler**
6. Select Project Name and Folder

Select Compiler

Compiler Toolchains

- XC8
 - XC8 (v2.36) [C:\Program Files\Microchip\xc8\v2.36\bin]**
- + CCS C Compiler
- + pic-as
- + PICBASIC PRO™

< Back Next > Finish Cancel Help

Select Project Name and Folder

Project Name:

Project Location:

Project Folder:

Overwrite existing project.
 Also delete sources.
 Set as main project
 Use project location as the project folder

Encoding:

< Back Next > Finish Cancel Help

High Endurance Flash – Invoke MCC

MPLAB X IDE v6.00 - cecHEF_Day1: default

File Edit View Navigate Source Refactor Production Debug Team Tools Window Help

default [Icons] PC: 0x0 z dc c : W:0x0 : bank 0 How do I? Keyword(s)

Projects | **Files** | **Services**

cecCuriosityDay1
 cecHEF_Day1
 Header Files
 Important Files
 Linker Files
 Source Files
 Libraries
 Loadables

Content Manager

MCC Content Manager Wizard

1. Content Type | 2. Required Device Content

Select a Content Type

MCC Melody ?

Supports the MCC Builder
 Supports content versioning at driver level
 An iteration of MCC Generated Code
 Works both on- and off-line

Select MCC Melody

[Release notes and supported devices](#)

MCC Classic ?

Development process you are accustomed to
 All components and libraries that you have used before

Select MCC Classic

[Release notes and supported devices](#)

MPLAB® Harmony ?

Embedded Software Development Framework for 32-bit Microcontrollers and Microprocessors

Select MPLAB Harmony

[Release notes and supported devices](#)

Library support may be a key factor in your choice of MCC flavor:

> MCC Melody and MCC Classic - Library Summary

> MPLAB Harmony - Library Summary

Still unsure which content type is right for your project?

[See More Details](#)

Output - MPLAB® Code Configurator

```

13:19:42.625 INFO: Fetching list of available libraries.
13:19:43.329 INFO: Download Complete: C:\Users\fred\.mcc\mcc_libraries.xml
13:19:43.898 INFO: Start MCC
13:19:43.909 INFO: Core v5.4.11 loaded.
  
```

cecHEF_Day1 - Dashboard | **EUSART_Write(uint8_t txData) - Navigator**

cecHEF_Day1
 Project Type: Application - Configuration: default
 Device
 PIC16F1619
 Checksum: Blank, no code loaded
 CRC32: Hex file unavailable
 Packs
 PIC12-16F1xxx_DFP (1.3.90)
 Compiler Toolchain
 XC8 (v2.36) [C:\Program Files\Microchip\xc8\v2.36\bin]
 Production Image: Optimization: Disabled
 Device support information: PIC12-16F1xxx_DFP (1.3.90)
 Memory
 Usage Symbols disabled. Click to enable Load Symbols.
 Data 1,024 (0x400) bytes
 Program 8,192 (0x2000) words
 Debug Tool
 Starter Kits (PKOB): BUR153472455
 Debug Resources
 Program BP Used: 0 Free: 1
 Data BP: No Support
 Data Capture BP: No Support
 Unlimited BP (S/W): No Support

High Endurance Flash – Configure I/O

MPLAB X IDE v6.00 - cecHEF_Day1 : default

File Edit View Navigate Source Refactor Production Debug Team Tools Window Help

default [V] [T] [W] [R] [C] [D] [E] [F] [G] [H] [I] [J] [K] [L] [M] [N] [O] [P] [Q] [R] [S] [T] [U] [V] [W] [X] [Y] [Z] [0] [1] [2] [3] [4] [5] [6] [7] [8] [9] [PC: 0x0] [z dc c : W:0x0 : bank 0] How do I? [Keyword(s)]

Projects | **Files** | **Services** | **Resource Management [MCC] x**

Project Resources | Generate | Import... | Export

- System
 - Interrupt Module
 - Pin Module
 - System Module

Device Resources | Content Manager

- Documents
 - PIC16F1619 Product Page
- Libraries
 - Foundation Services
 - MQTT
- Peripherals
 - ADC
 - AT

cecHEF_Day1 - Dashboard | EUSART_Write(uint8_t txData) - N... | **Core Versions [MCC] x**

MCC Core Versions

- MPLAB® Code Configurator (Plugin) v5.1.17

Libraries may be updated in the Content Manager.

Packs x | Pin Module x | Interrupt Module x | System Module x

Easy Setup | Registers

Selected Package: PDIP20

Pin Name	Module	Function	Custom Name	Start High	Analog	Output	WPU	OD	IOC
RA5	Pin Module	GPIO	LED_D6	<input type="checkbox"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	none

Pin Manager: Package View x

Notifications | **Output - MPLAB® Code Configurator** | **Pin Manager: Grid View x**

Package:	Pin No:	19	18	17	4	3	2	13	12	11	10	16	15	14	7	6	5	8	9
		Port A ▼				Port B ▼				Port C ▼									
Module	Function	Direction	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	
OSC	CLKOUT	output																	
Pin Module ▼	GPIO	input																	
	GPIO	output																	
RESET	MCLR	input																	

High Endurance Flash – Include Flash Routines

MPLAB X IDE v6.00 - cecHEF_Day1: default

File Edit View Navigate Source Refactor Production Debug Team Tools Window Help

default [MCC] [MEMORY] PC: 0x0 z dc c : W:0x0 : bank 0

Projects Files Services **Resource Management [MCC]** Pins x Pin Module x Interrupt Module x System Module x MEMORY x

Project Resources Generate Import... Export

- Peripherals
 - MEMORY
- System
 - Interrupt Module
 - Pin Module
 - System Module

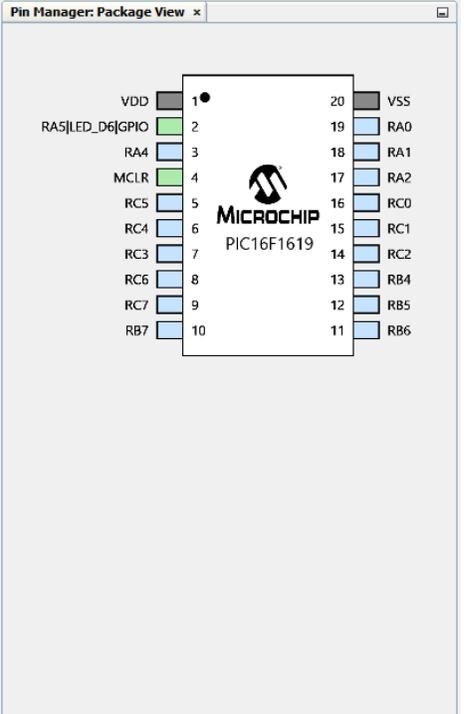
Device Resources Content Manager

- Ext_Interrupt
- FVR
- Math Accelerator
- MSSP
- PWM
- SMT
- Timer
- ZCD

Easy Setup

Flash Routines are included

Pin Manager: Package View



cecHEF_Day1 - Dashboard EUSART_Write(uint8_t txData) - N... Core Versions [MCC]

MCC Core Versions

- MPLAB® Code Configurator (Plugin) v5.1.17
- Libraries may be updated in the Content Manager.

Notifications Output - MPLAB® Code Configurator Pin Manager: Grid View x

Package:	PIDP20	Pin No:	19	18	17	4	3	2	13	12	11	10	16	15	14	7	6	5	8	9
			Port A ▼				Port B ▼				Port C ▼									
Module	Function	Direction	0	1	2	3	4	5	4	5	6	7	0	1	2	3	4	5	6	7
OSC	CLKOUT	output																		
Pin Module ▼	GPIO	input																		
	GPIO	output																		
RESET	MCLR	input																		

High Endurance Flash – Configure the EUSART

MPLAB X IDE v6.00 - cecHEF_Day1: default

File Edit View Navigate Source Refactor Production Debug Team Tools Window Help

default [Icons] PC: 0x0 z dc c : W:0x0 : bank 0 How do I? [Keyword(s)]

Projects Files Services **Resource Management [MCC]** x

Project Resources Generate Import... Export

- Peripherals
 - EUSART
 - MEMORY
- System
 - Interrupt Module
 - Pin Module
 - System Module

Device Resources Content Manager

- Comparator
- CRC
- CWG
- DAC
- Ext_Interrupt
- FVR
- Math Accelerator
- MSSP

cecHEF_Day1 - Dashboard EUSART_Write(uint8_t txData) - N... Core Versions [MCC] x

MCC Core Versions

- MPLAB® Code Configurator (Plugin) v5.1.17
Libraries may be updated in the Content Manager.

Easy Setup Registers

Hardware Settings

Mode asynchronous

Enable EUSART Baud Rate: 9600 Error: 0.040 %

Enable Transmit Transmission Bits: 8-bit

Enable Wake-up Reception Bits: 8-bit

Auto-Baud Detection Data Polarity: Non-Inverted

Enable Address Detect Enable Receive

Enable EUSART Interrupts

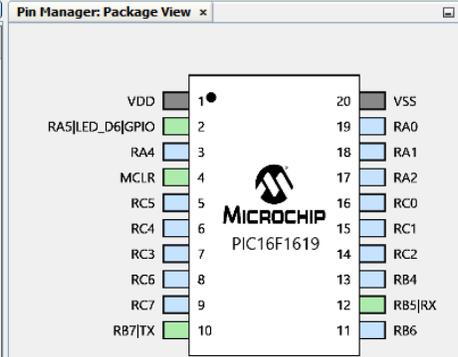
Software Settings

Redirect STDIO to USART

Software Transmit Buffer Size: 8

Software Receive Buffer Size: 8

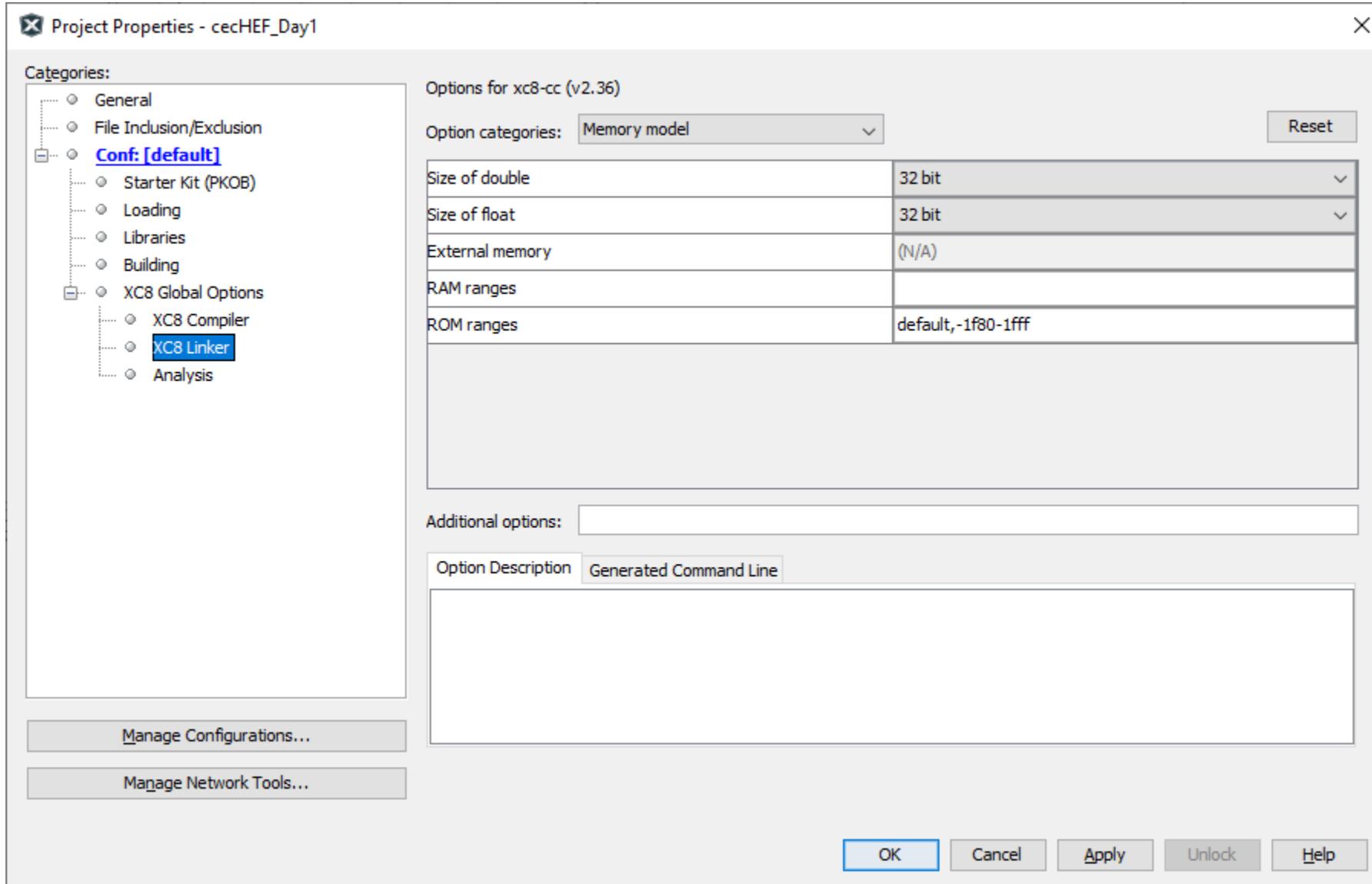
Pin Manager: Package View



Notifications Output - MPLAB® Code Configurator Pin Manager: Grid View x

Package:	PDI20	Pin No:	19	18	17	4	3	2	13	12	11	10	16	15	14	7	6	5	8	9
			Port A				Port B				Port C									
Module	Function	Direction	0	1	2	3	4	5	4	5	6	7	0	1	2	3	4	5	6	7
EUSART	RX	input	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	TX	output	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
OSC	CLKOUT	output																		
Pin Module	GPIO	input	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	GPIO	output	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
RESET	MCLR	input				🔒														

High Endurance Flash – Reserve HEF Memory Area



Project Properties - cecHEF_Day1

Categories:

- General
- File Inclusion/Exclusion
- Conf: [default]**
 - Starter Kit (PKOB)
 - Loading
 - Libraries
 - Building
 - XC8 Global Options
 - XC8 Compiler
 - XC8 Linker**
 - Analysis

Options for xc8-cc (v2.36)

Option categories: Memory model Reset

Size of double	32 bit
Size of float	32 bit
External memory	(N/A)
RAM ranges	
ROM ranges	default,-1f80-1fff

Additional options:

Option Description Generated Command Line

Manage Configurations...

Manage Network Tools...

OK Cancel Apply Unlock Help

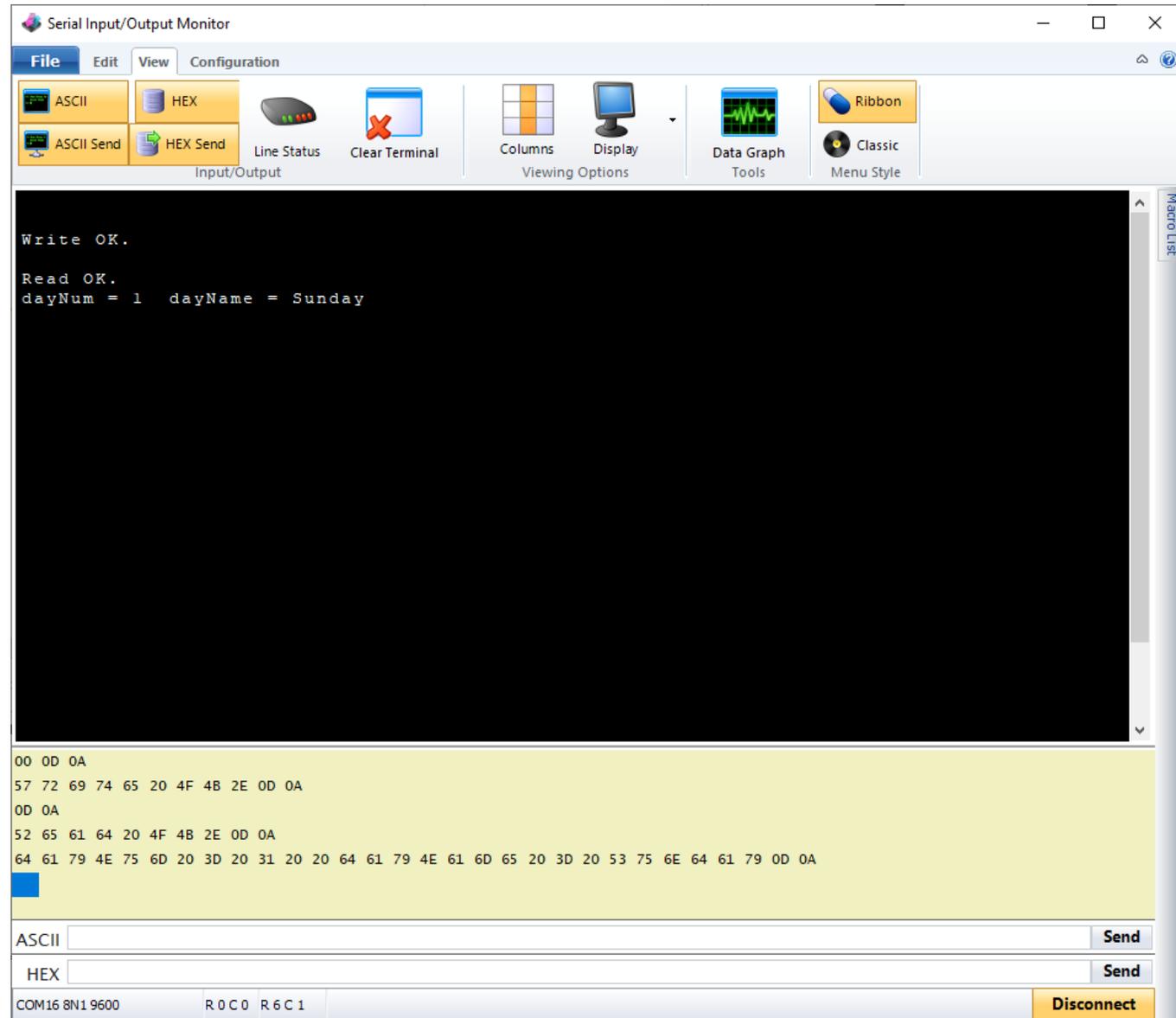
High Endurance Flash – Code and Execute

```

int8_t rc;
char  buffer[ HEF_BLOCKSIZE];
typedef struct
{
    uint16_t dayNum;
    char  dayName[20];
} DOW;
DOW day; //spawn day structure
void main(void)
{
    SYSTEM_Initialize();

    LED_D6_SetLow(); //LED off
    day.dayNum = 1; //Load dayNum
    sprintf(day.dayName, "Sunday"); //Load dayName
    rc = HEF_ByteArrayWrite( 1, (void *) &day, sizeof( day)); //write struct day info to HEF
    if(rc == 0)
    {
        printf("\r\nWrite OK.\r\n");
    }
    else
    {
        printf("\r\nWrite FAILED!\r\n");
        while(1)
        {
            LED_D6_Toggle();
            __delay_ms(100);
        }
    }
    memset(&day,0x00,sizeof(day)); //clear day structure to 0x00
    rc = HEF_ByteArrayRead( (void *) &day, 1, sizeof( day)); //read HEF
    if(rc == 0)
    {
        printf("\r\nRead OK.\r\n");
        printf("dayNum = %u  dayName = %s\r\n",day.dayNum,day.dayName);
    }
    else
    {
        printf("\r\nRead FAILED!\r\n");
        while(1)
        {
            LED_D6_Toggle();
            __delay_ms(100);
        }
    }
}

```



Serial Input/Output Monitor

File Edit View Configuration

ASCII HEX Line Status Clear Terminal Columns Display Data Graph Ribbon Classic Menu Style

Input/Output Viewing Options Tools

```

Write OK.

Read OK.
dayNum = 1  dayName = Sunday

```

00 0D 0A
57 72 69 74 65 20 4F 4B 2E 0D 0A
0D 0A
52 65 61 64 20 4F 4B 2E 0D 0A
64 61 79 4E 75 6D 20 3D 20 31 20 20 64 61 79 4E 61 6D 65 20 3D 20 53 75 6E 64 61 79 0D 0A

ASCII Send

HEX Send

COM16 8N19600 R 0 C 0 R 6 C 1 Disconnect

MORE TO COME..

Thank you for attending!!!

Please consider the resources below:

- ccsinfo.com
- **AN1673**
- microchip.com





Thank You

Sponsored by

