TI-RSLK MAX

Texas Instruments Robotics System Learning Kit

TEXAS INSTRUMENTS

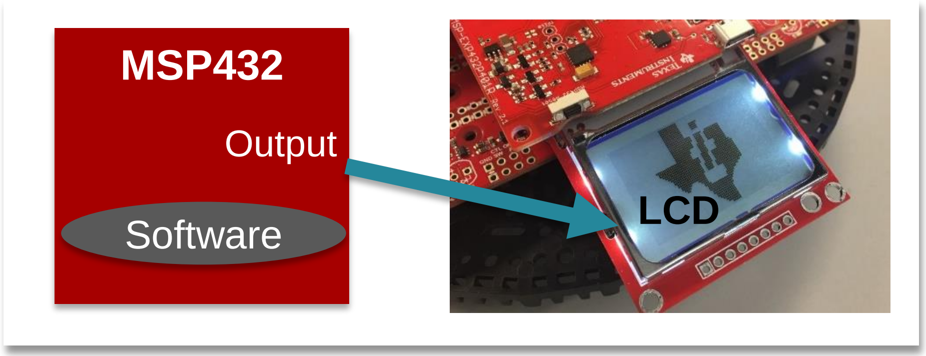
Module 11

Lecture: Liquid Crystal Display

Liquid Crystal Display

You will learn in this module

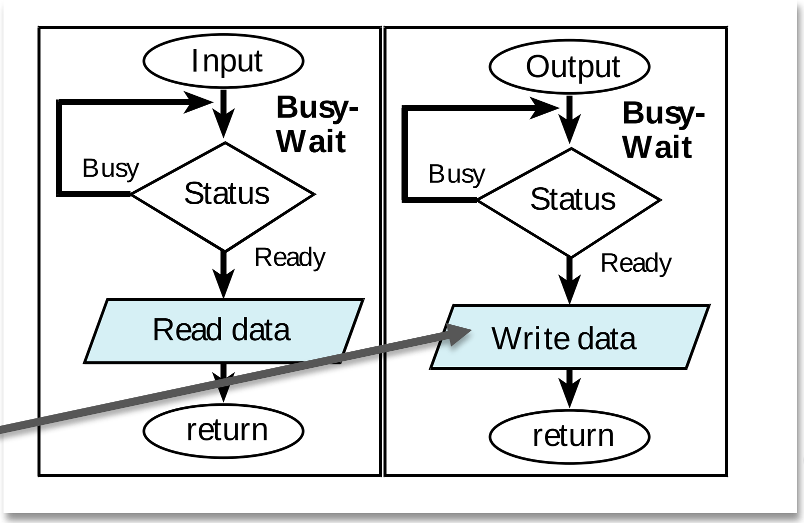
- Busy-wait hardware/software synchronization
- Fundamentals of synchronous serial communication
- How to interface an LCD to TI's Launchpad Development board
- Software driver (set of functions to create an abstract module)
- Create a minimally intrusive debugging monitor



Hardware/software synchronization

The fundamental problem

- Software executes quickly (48 MHz)
  - Instruction takes 42 ns
- Hardware operates slowly
  - Takes 2 µs to send 1 byte
  - Takes 14 µs to output a character
- Solutions
  - Blind (fixed wait time)
  - Busy-wait
  - Interrupts (Labs 10,13,14)
  - Direct memory access



Synchronous Serial Communication on the MSP432

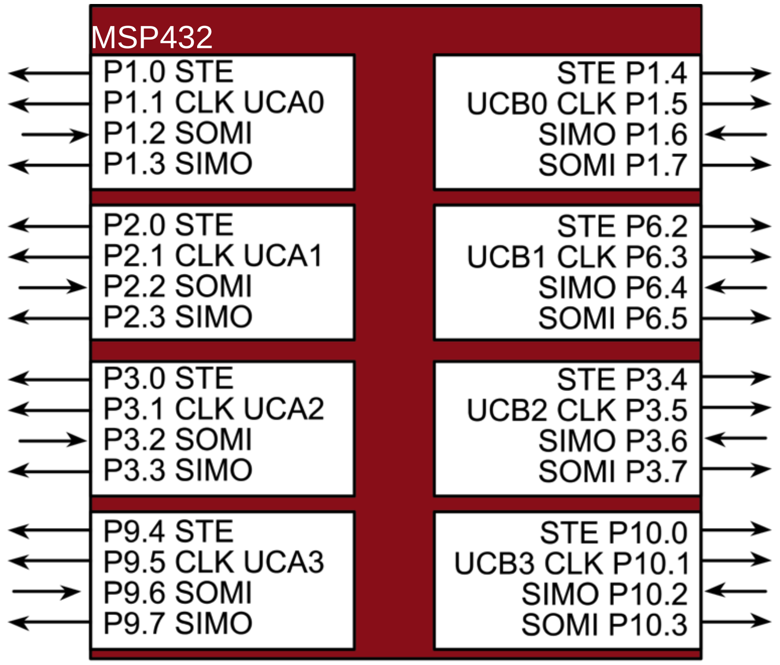
Components

- Enable
- Clock
- Data out
- Data in

MSP432 is master

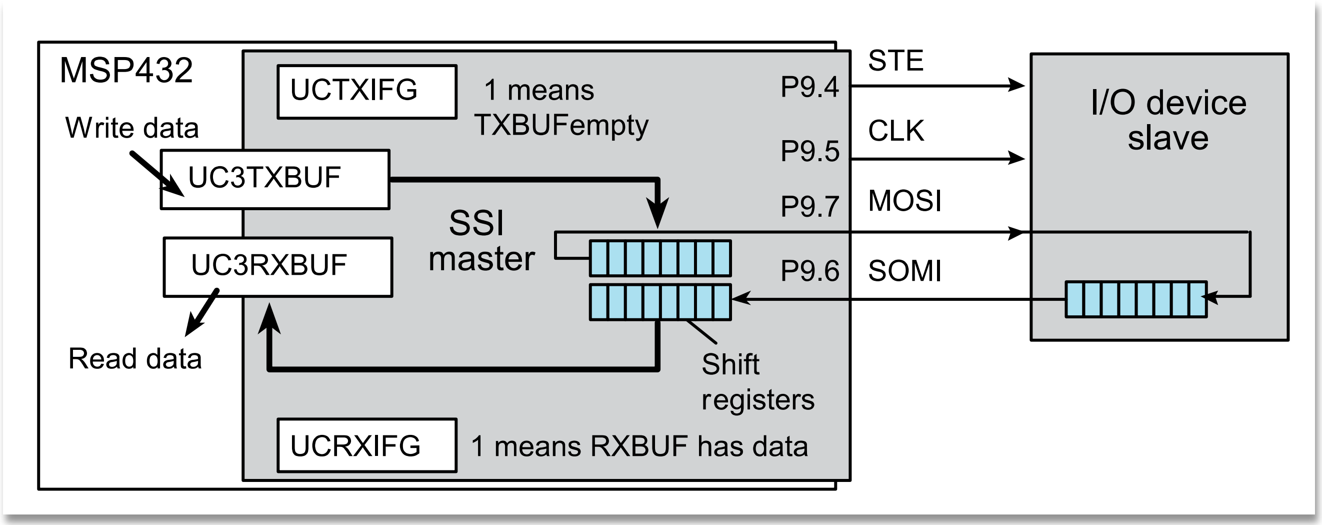
- Drives clock
- Drives enable
- Initiates transfer

LCD is slave



Synchronous Serial Communication on the MSP432

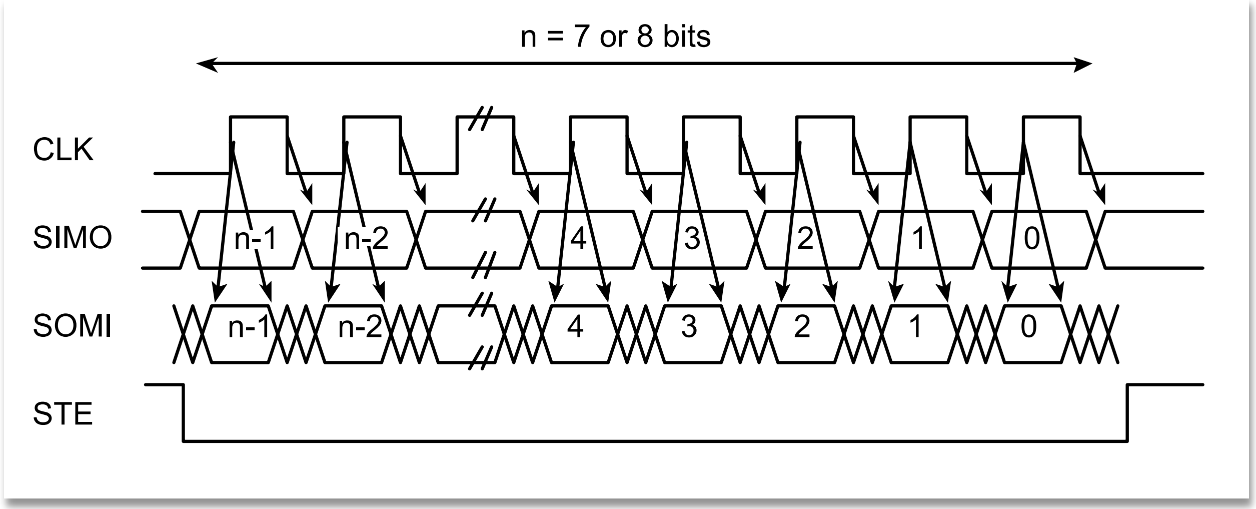
- Synchronous means send clock and data
  - Send data on one edge of clock
  - Receive data on other edge
- Serial Peripheral Interface (SPI) Protocol



Serial Peripheral Interface (SPI) Timing

Signals

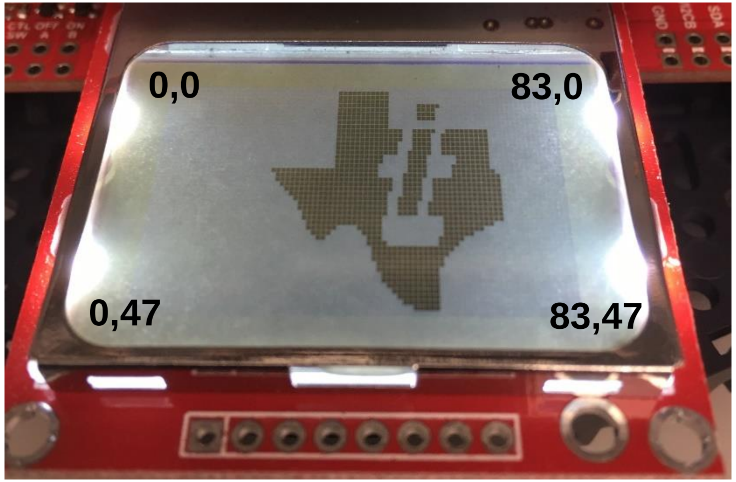
- Clock
- Data out
- Data in
- Enable



Nokia5110 LCD functionality

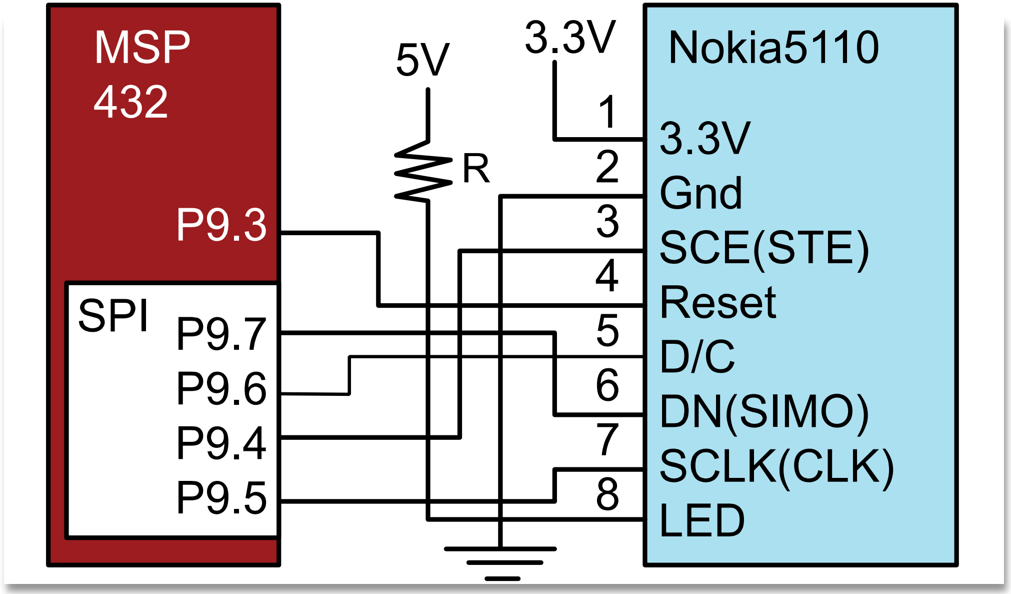
Monochrome

- Serial Peripheral Interface (SPI)
  - 5 pins
- 84 pixels wide
- 48 pixels high
- 4 MHz speed
- Low cost



LCD Interface

- SPI
  - P9.4 STE
  - P9.5 CLK
  - P9.7 SIMO
- GPIO
  - P9.3 Reset
  - P9.6 Data/command



Decimal output

Output an unsigned integer, n

- Assume n is between 1000 and 9999
- Print as 5 characters, right justified

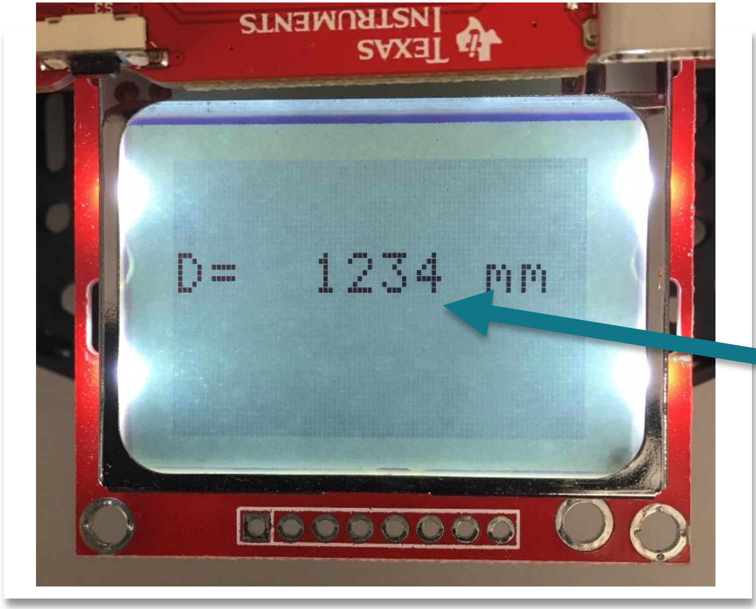
```
OutChar(0x20);           // space
OutChar(0x30+n/1000);    // thousand's digit
n = n%1000;
OutChar(0x30+n/100);     // hundred's digit
n = n%100;
OutChar(0x30+n/10);      // ten's digit
OutChar(0x30+n%10);      // one's digit
```

Application

LCD provides

1. Debugging information in real time as robot is moving (14 µs/character)
2. Graphical representation of data (optional)

Minimally intrusive



```
Nokia5110_SetCursor(0,2);
Nokia5110_OutString("D= ");
Nokia5110_OutUDec(distance);
Nokia5110_OutString(" mm");
```

4+5*14=74 µs

```
Nokia5110_SetCursor(3,2);
Nokia5110_OutUDec(distance);
```

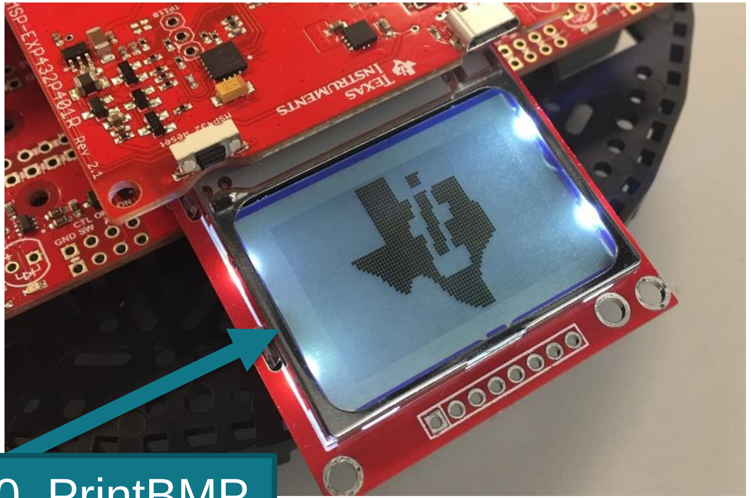
Summary

- Busy-wait synchronization
- Synchronous serial communication
- Numerical output
- Minimally intrusive debugging monitor
- Graphics: 4-bit, 16-color BMP

```
const uint8_t ti[] = {
 0x42, 0x4D, 0xF6, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x76, 0x00, 0x00, 0x00, 0x28, 0x00, 0x00, 0x00, 0x30, 0x00, 0x00, 0x00, 0x30
 0x00, 0x00, 0x80, 0x04, 0x00, 0x00, 0x23, 0x2E, 0x00, 0x00, 0x23, 0x2E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x80, 0x00, 0x00, 0x00, 0x80, 0x00, 0x80, 0x00, 0x80, 0x80, 0x00, 0x00, 0x80, 0x80, 0x80, 0x00, 0xC0
 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0x00, 0xFF, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00, 0x
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF
 [illegible] 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, [illegible]
 [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0
 0x00, [illegible] 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 0x00, 0x00, [illegible] 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x
 0x0F, 0xFF, 0xFF, [illegible] 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x
 [illegible] 0xFF, 0x00, 0x00, 0x00, 0x0
 [illegible] 0x00, 0x00, 0x00, 0x00, 0x0
 [illegible] 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F,
 [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, [illegible] 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00,
 [illegible] 0xFF, 0xFF, 0x00, 0xFF, [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 [illegible] 0xF0, 0x0[illegible] 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0x00, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xF[illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F[illegible]F0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00,
 0x0F, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F, 0xFF, 0xF0, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0x00,
 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF
};
```

BmpConvert.exe

Nokia5110_PrintBMP



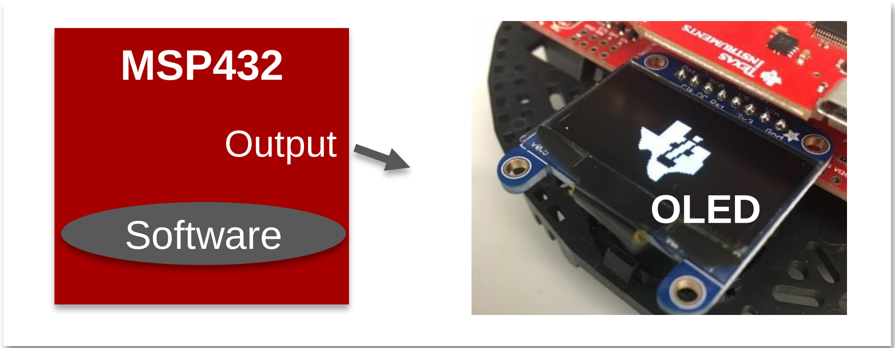
Module 11

Lecture: Organic light-emitting diode display (OLED)

OLED Display

You will learn in this module

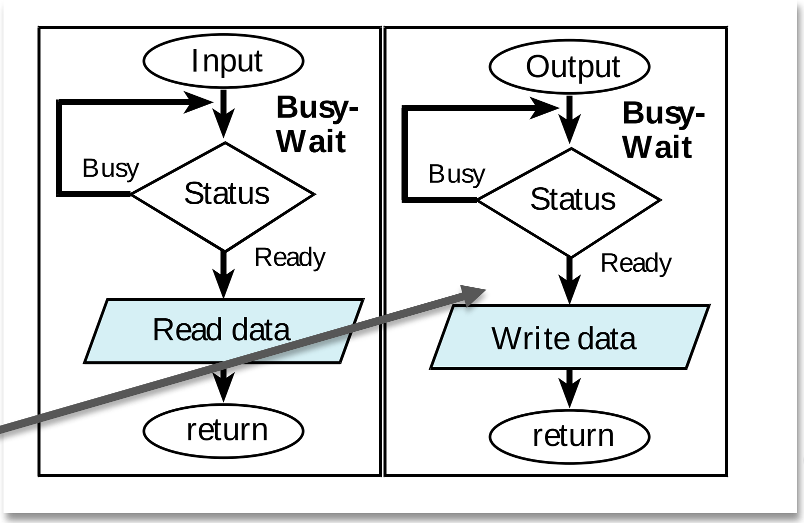
- Busy-wait hardware/software synchronization
- Fundamentals of synchronous serial communication
- How to interface an OLED to TI's Launchpad Development board
- Software driver (set of functions to create an abstract module)
- Create a minimally intrusive debugging monitor



Hardware/software synchronization

The fundamental problem

- Software executes quickly (48 MHz)
  - Instruction takes 42 ns
- Hardware operates slowly
  - Takes 2 µs to send 1 byte
  - Takes 12 µs to output a character
- Solutions
  - Blind (fixed wait time)
  - Busy-wait
  - Interrupts (Labs 10,13,14)
  - Direct memory access



Synchronous Serial Communication on the MSP432

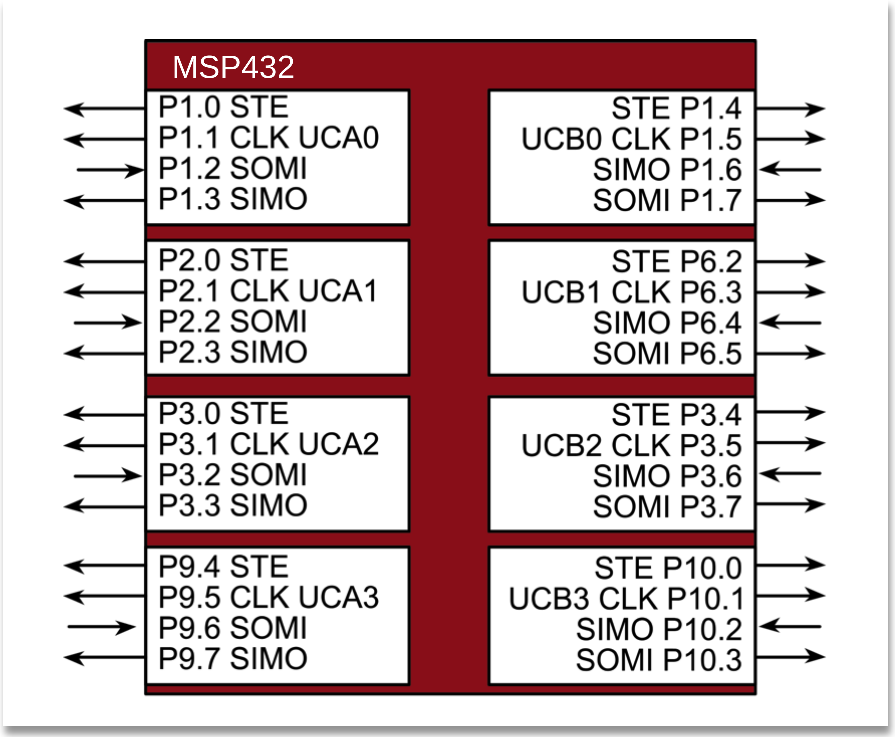
Components

- Enable
- Clock
- Data out
- Data in

MSP432 is master

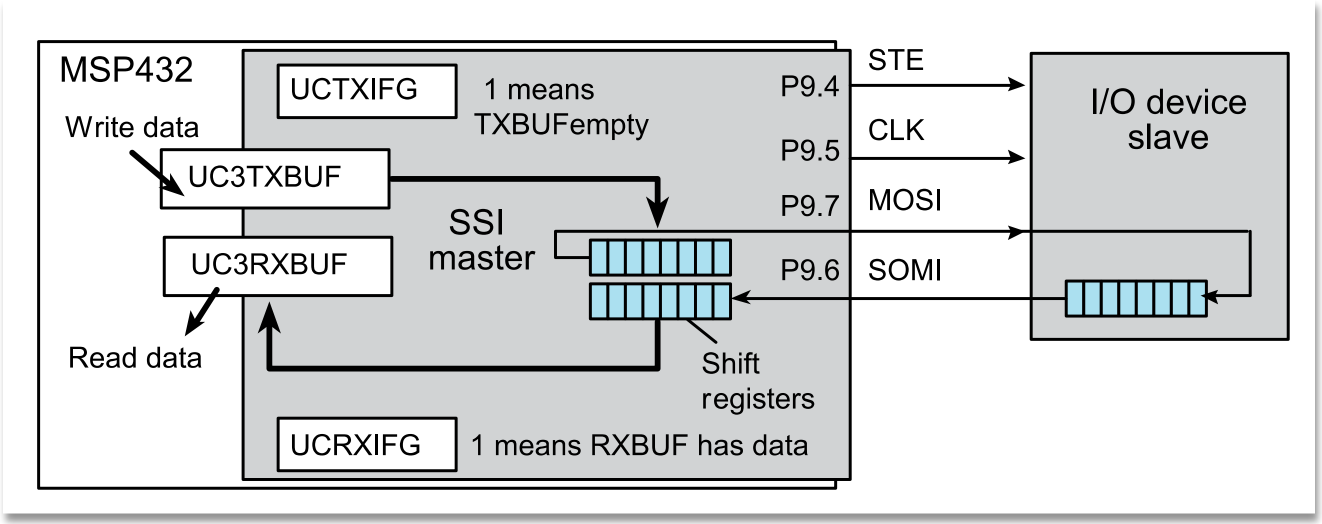
- Drives clock
- Drives enable
- Initiates transfer

OLED is slave



Synchronous Serial Communication on the MSP432

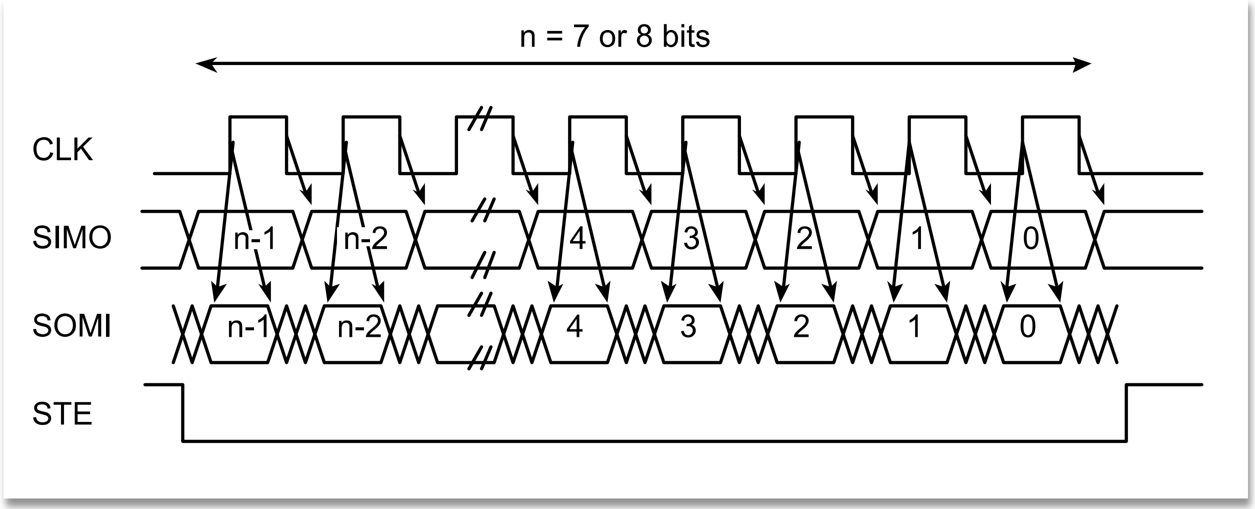
- Synchronous means send clock and data
  - Send data on one edge of clock
  - Receive data on other edge
- Serial Peripheral Interface (SPI) Protocol



Serial Peripheral Interface (SPI) Timing

Signals

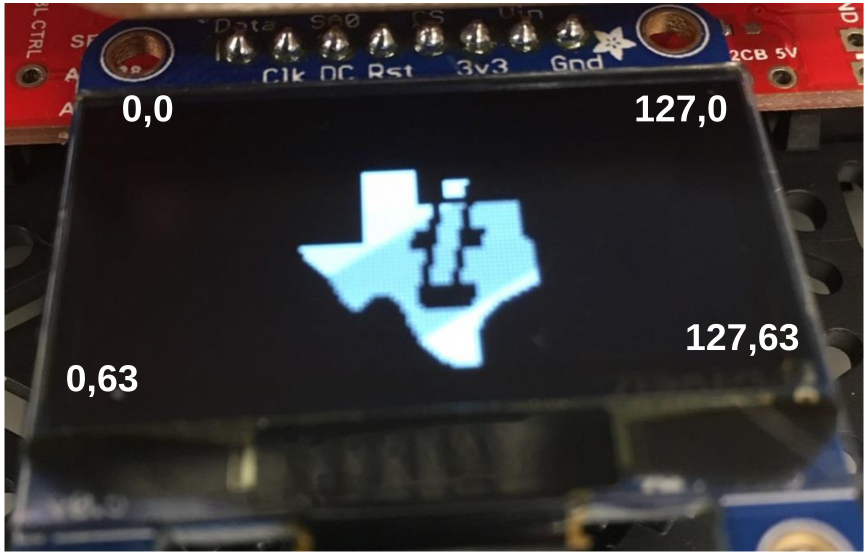
- Clock
- Data out
- Data in
- Enable



SSD1306 OLED functionality

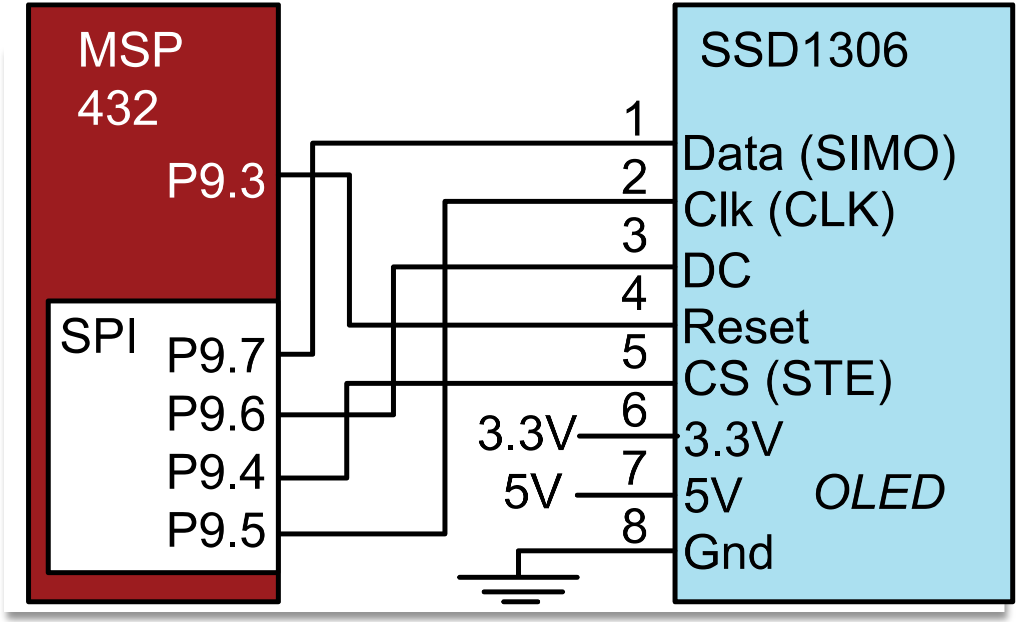
Monochrome

- Serial Peripheral Interface (SPI)
  - 5 pins
- 128 pixels wide
- 64 pixels high
- 4 MHz speed
- Low cost



OLED Interface

- SPI
  - P9.4 STE
  - P9.5 CLK
  - P9.7 SIMO
- GPIO
  - P9.3 Reset
  - P9.6 Data/command



Decimal output

Output an unsigned integer, n

- Assume n is between 1000 and 9999
- Print as 5 characters, right justified

```
OutChar(0x20);              // space
OutChar(0x30+n/1000);       // thousand's digit
n = n%1000;
OutChar(0x30+n/100);        // hundred's digit
n = n%100;
OutChar(0x30+n/10);         // ten's digit
OutChar(0x30+n%10);         // one's digit
```

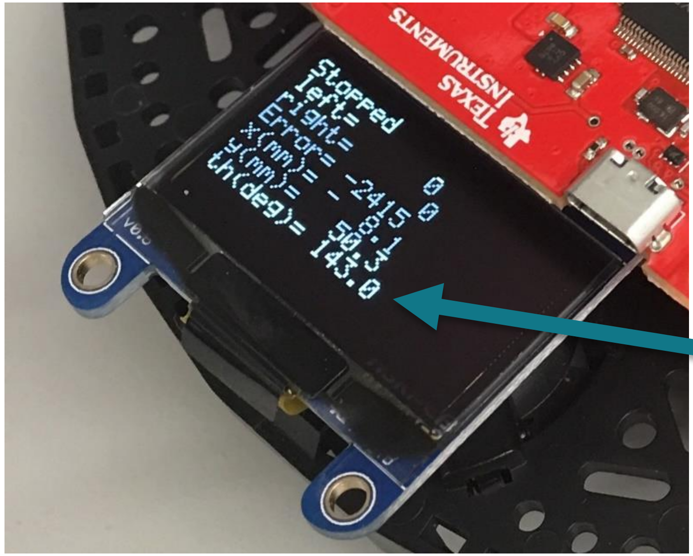
Application

OLED provides

1. Debugging information in real time as robot is moving (12µs/character)
2. Graphical representation of data (optional)

Minimally intrusive

```
SSD1306_SetCursor(0,6);
SSD1306_OutString("th(deg)  ");
SSD1306_OutSFix1(theta);
```



```
Stopped
left=          0
right=         0
Error=     -2415
x(mm)=      -8.1
y(mm)=      50.3
th(deg)=   143.0
```

12+6*12=84 µs

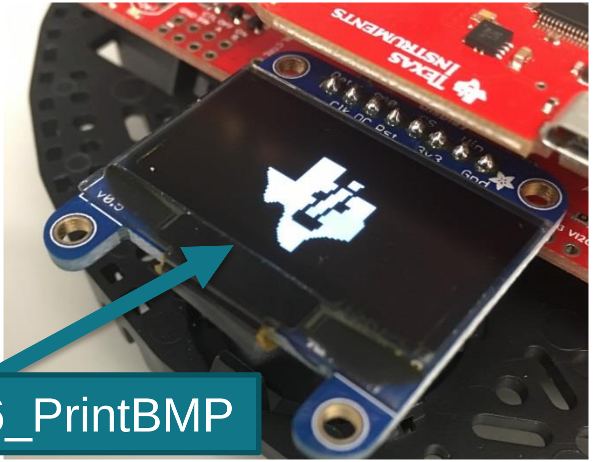
```
SSD1306_SetCursor(8,6);
SSD1306_OutSFix1(theta);
```

Summary

- Busy-wait synchronization
- Synchronous serial communication
- Numerical output
- Minimally intrusive debugging monitor
- Graphics: 4-bit, 16-color BMP

BmpConvert.exe

SSD1306_PrintBMP



```
const uint8_t ti[] = {
 0x42, 0x4D, 0xF6, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x76, 0x00, 0x00, 0x00, 0x28, 0x00, 0x00, 0x00, 0x30, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x80, 0x04, 0x00, 0x00, 0x23, 0x2E, 0x00, 0x00, 0x23, 0x2E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x80, 0x80, 0x00, 0x80, 0x00, 0x00, 0x00, 0x80, 0x00, 0x80, 0x00, 0x80, 0x80, 0x00, 0x00, 0x80, 0x80, 0x80, 0x00,
 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0x00, 0xFF, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF,
 [illegible] 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, [illegible]
 [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, [illegible]
 0x00, [illegible] 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, [illegible]
 0x00, 0x00, [illegible] 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, [illegible]
 0x0F, 0xFF, 0xFF, [illegible] 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, [illegible]
 [illegible] 0xFF, 0x00, 0x00, 0x00, [illegible]
 [illegible] 0x00, 0x00, 0x00, 0x00, [illegible]
 [illegible] 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, [illegible] 0xFF, 0xFF, 0xFF, 0xFF,
 [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, [illegible] 0xF0, 0x00, 0x00, 0x00, 0x00,
 [illegible] 0xFF, 0xFF, 0x00, 0xFF, [illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 [illegible] 0xF0, 0x0[illegible] 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0x00, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xF[illegible] 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F[illegible] 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF,
 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00,
 0x0F, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0x00, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F, 0xFF, 0xF0, 0x0F,
 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x0F, 0xFF, 0xFF, 0x00,
 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFF
};
```

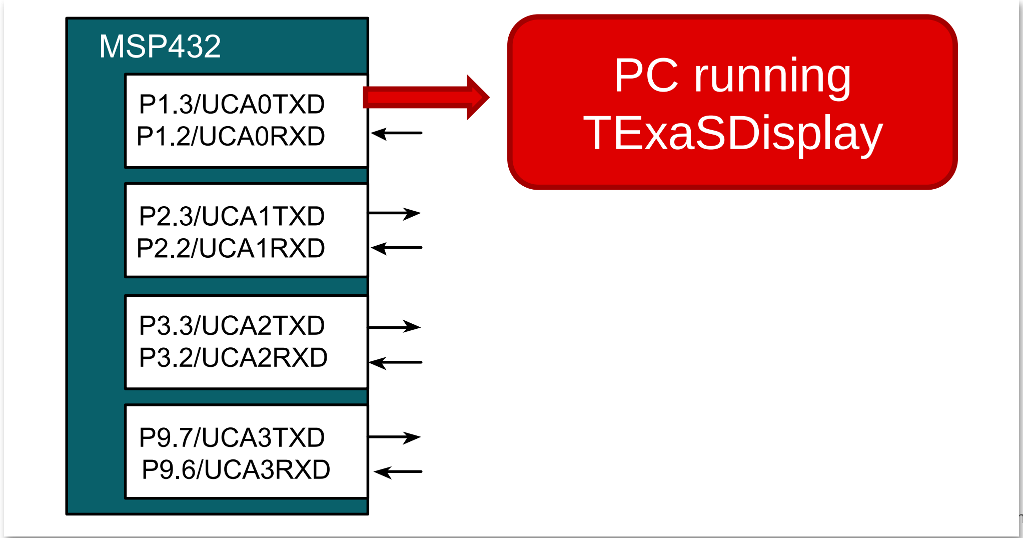
Module 11

Lecture: UART(for debugging)

Serial Communication

You will learn in this module

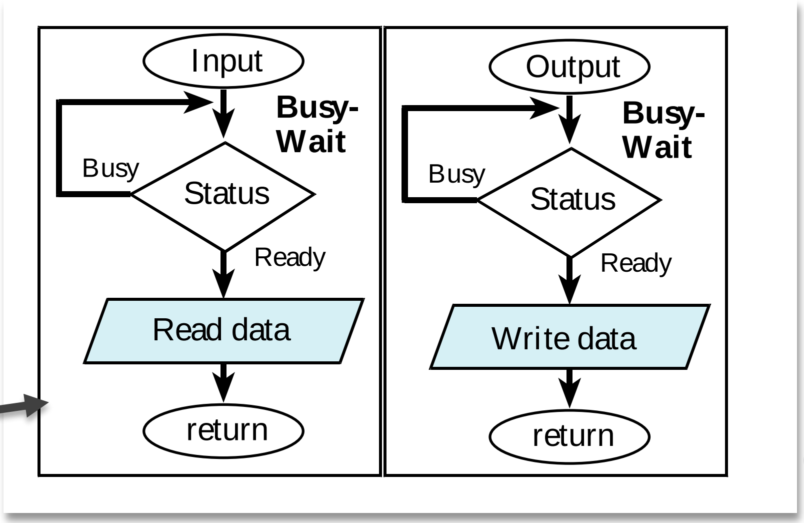
- Busy-wait hardware/software synchronization
- Fundamentals of asynchronous serial communication
- Software driver (set of functions to create an abstract module)
- Create a minimally intrusive debugging monitor



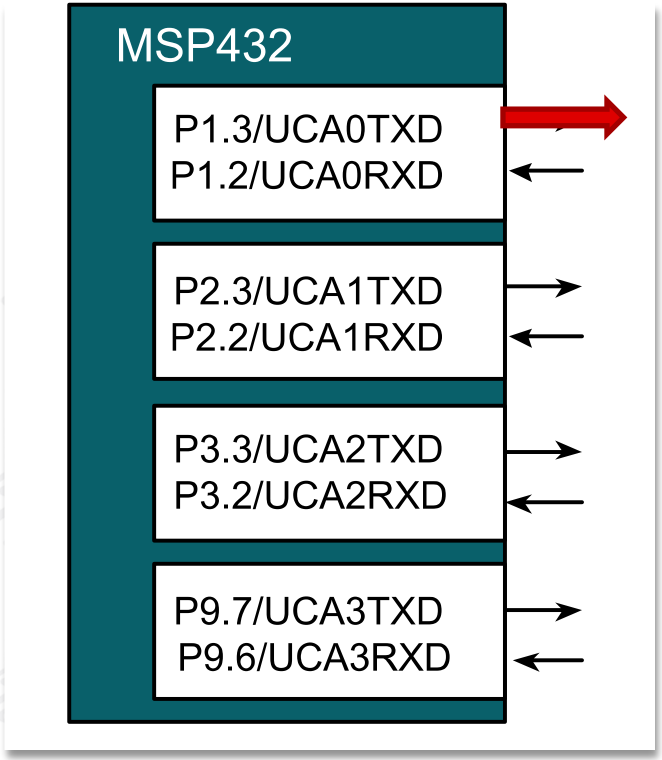
Hardware/software synchronization

The fundamental problem

- Software executes quickly (48 MHz)
  - Instruction takes 42 ns
- Hardware operates slowly
  - UART takes 87 µs to output a character
- Solutions
  - Blind (fixed wait time)
  - Busy-wait
  - Interrupts (Labs 10,13,14)
  - Direct memory access

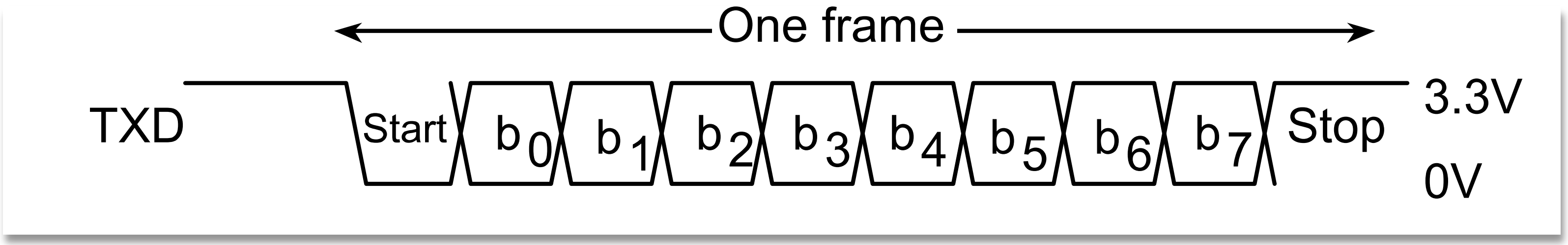


UART Port Selection



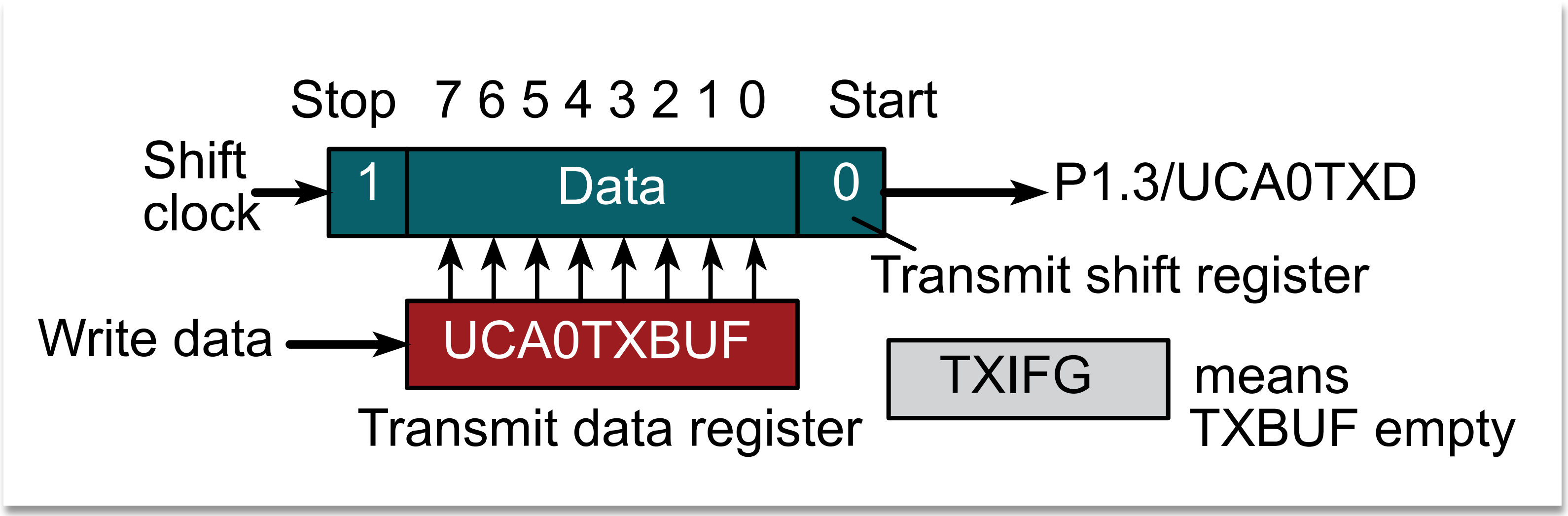
Pin	PxSEL1=0, PxSEL0=1
P1.2	UCA0RXD
P1.3	UCA0TXD
P2.2	UCA1RXD
P2.3	UCA1TXD
P3.2	UCA2RXD
P3.3	UCA2TXD
P9.6	UCA3RXD
P9.7	UCA3TXD

Universal Asynchronous Receiver/Transmitter (UART)



- Send/receive a **frame**
  - 1 start (low), 5-8 data bits , 1 stop (high)
  - Serial fashion, one bit every **bit-time**
  - No clock is sent, asynchronous, timing derived from data
- **Baud rate** is total number of bits per unit time
  - Baud rate = 1 / bit-time
  - Both transmitter and receiver agree to use the same baud rate
- **Bandwidth** is data or information per unit time
  - Bandwidth = (data-bits / frame-bits) * baud rate

UART - Transmitter



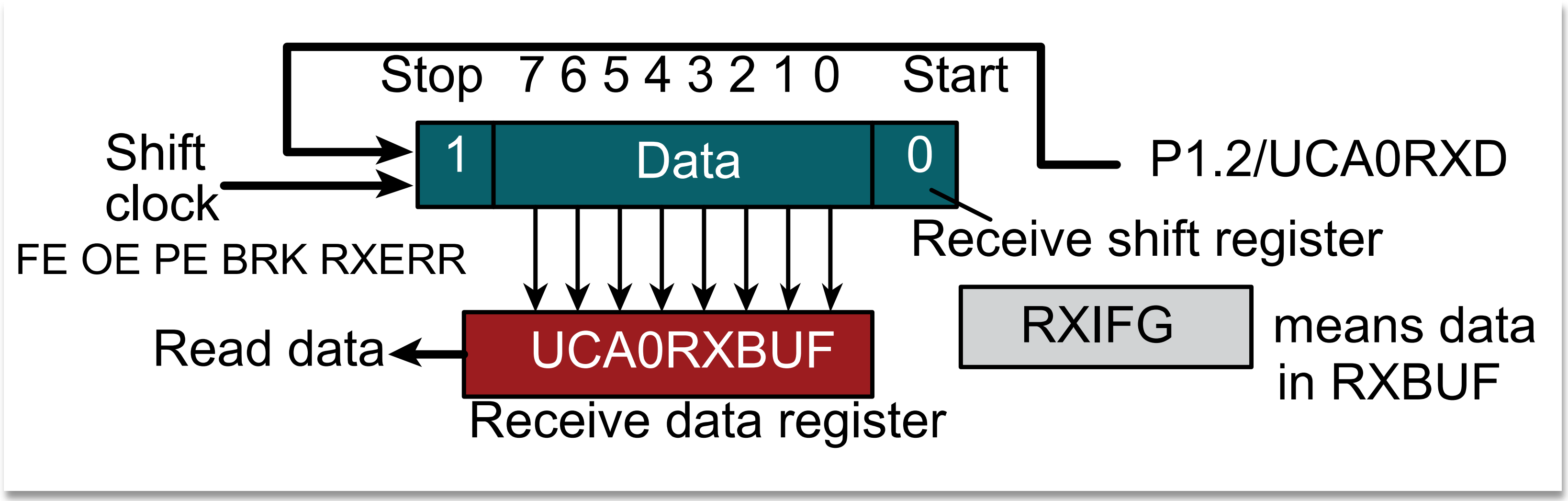
Software

- Busy-wait on TXIFG
- Write data to UCA0TXBUF

Hardware

- Add start, stop bits
- Shift out at Baud Rate clock

UART - Receiver



Hardware

- Wait for start
- Shift in Data at Baud Rate clock
- Check for errors
- Remove start, stop
- Set RXIFG

Software

- Busy-wait on RXIFG
- Read data from UCA0RXBUF

UART Registers

	15	14	13	12	11	10	9	8	
0x40001000	PEN	PAR	MSB	7BIT	SPB	MODEx		SYNC	UCAxCTLW0

	7	6	5	4	3	2	1	0	
	SSELx		RXEIE	BRKIE	DORM	TXADDR	TXBRK	SWRST	UCAxCTLW0

	15 – 0	
0x40001006	UCBRx	UCAxBRW

	7	6	5	4	3	2	1	0	
0x4000100A	LISTEN	FE	OE	PE	BRK	RXERR	IDLE	BUSY	UCAxSTATW

	15 – 8	7 – 0	
0x4000100C	▨	RXBUFx	UCAxRXBUF

	15 – 8	7 – 0	
0x4000100E	▨	TXBUFx	UCAxTXBUF

	15 – 4	3	2	1	0	
0x4000101C	▨	TXCPTIFG	STTIFG	TXIFG	RXIFG	UCAxIFG

Decimal output

Output an unsigned integer, n

- Assume n is between 1000 and 9999
- Print as 5 characters, right justified

```
OutChar(0x20);               // space
OutChar(0x30+n/1000);        // thousand's digit
n = n%1000;
OutChar(0x30+n/100);         // hundred's digit
n = n%100;
OutChar(0x30+n/10);          // ten's digit
OutChar(0x30+n%10);          // one's digit
```

Application

UART serial output provides

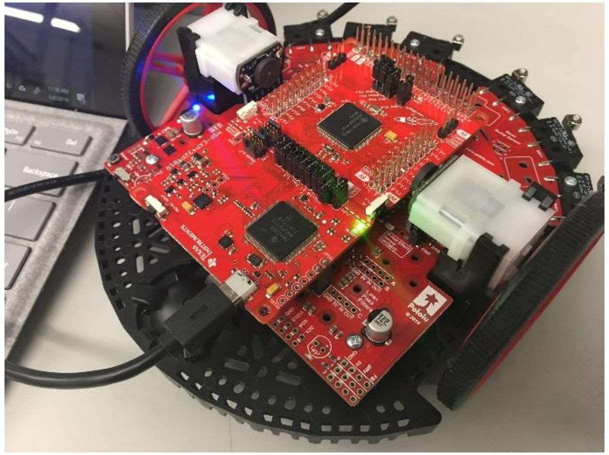
1. Debugging information in real time as robot is moving (87 µs/character)
2. Numerical and character information

Moderately intrusive

```
UART_OutString("D= in mm\n");
```

4char*87µs/char = 348 µs

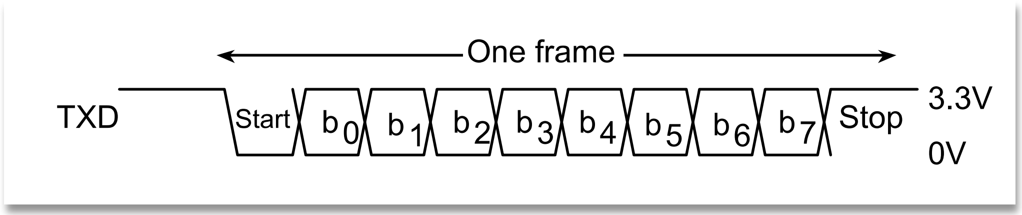
```
UART_OutUDec(distance);
UART_OutChar('\n');
```



Assume called every 100ms;
Intrusiveness = 348µs/100ms = 0.35%

Summary

- Busy-wait synchronization
- Asynchronous serial communication
- Numerical output
- Moderately intrusive debugging monitor



ti.com/rslk