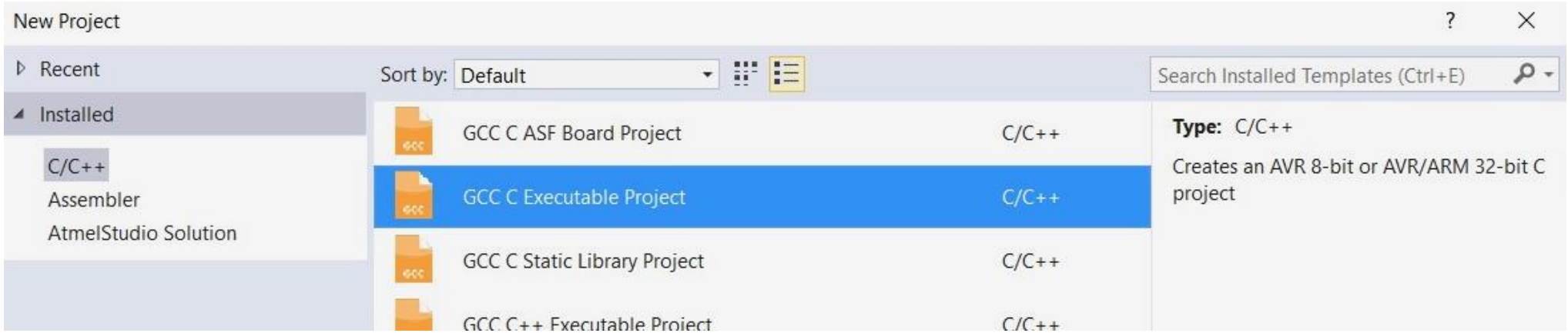
C Project

Ensure you select the ‘GCC C Executable Project’ type

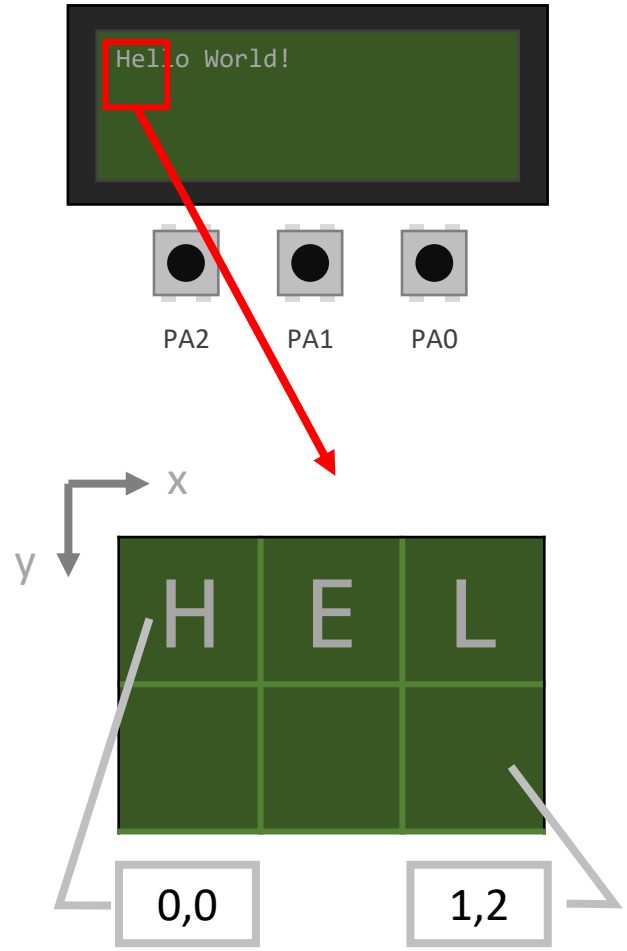


ADC

- You will need to use ADC in this lab
- There are three registers that need to be set for each task
  - ADCSRA
  - ADCSRB
  - ADMUX – this is set in the `readADC()` function, as we have multiple ADC channels to read
- Refer to Lab 4 PowerPoint and content for information

LCD Display

- This week will require using the 20x4 Alphanumeric display
  - Review the contents of your lab manual to see how this is setup
  - The example on Page 24 is a good example to refer to
- Call these functions in your `setup()`
  - `SLCDInit()` – This will initiate the TWI serial protocol to allow your µC to talk to the LCD
  - `SLCDDisplayOn()` – This will turn on the display to show characters
  - `SLCDClearScreen()` – This will clear the screen of any existing content (such as the last students code)
- The LCD is split into a 20 x 4 grid
  - The top-left cell is position 0,0
  - Use the `SLCDSetCursorPosition(y,x)` function to set the cursor position
  - Use the `SLCDWriteString(charArrayToWrite)` function to place text on the screen
  - The text needs to be in the form of a `char` array (use the `sprintf()` function)



sprintf Function

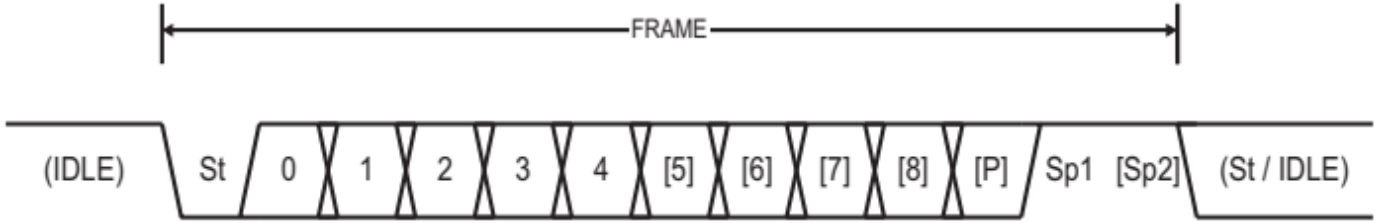
```
// Example of using the sprintf function

char line0[20];
sprintf(line0, "temperature - %3d",tempSensorValue);
```

- ‘line0’ is the char array in which the string will be stored in
  - This should be defined prior (as per the example above), and we set this to a blank array of 20 char values (since each line of the display can only show 20 characters)
- ‘temperature - %3d’ is the string we want to convert to show on the display
  - %3d indicates that we intend to reserve three (‘3’) spaces for a decimal number (‘d’) to be displayed
- ‘tempSensorValue’ is the number we plan to substitute in this location
  - The space we reserved earlier will be replaced by a 3-digit decimal number
- Multiple variables can be used for the sprintf function
  - See your lab manual for more information

USART – Data Packet

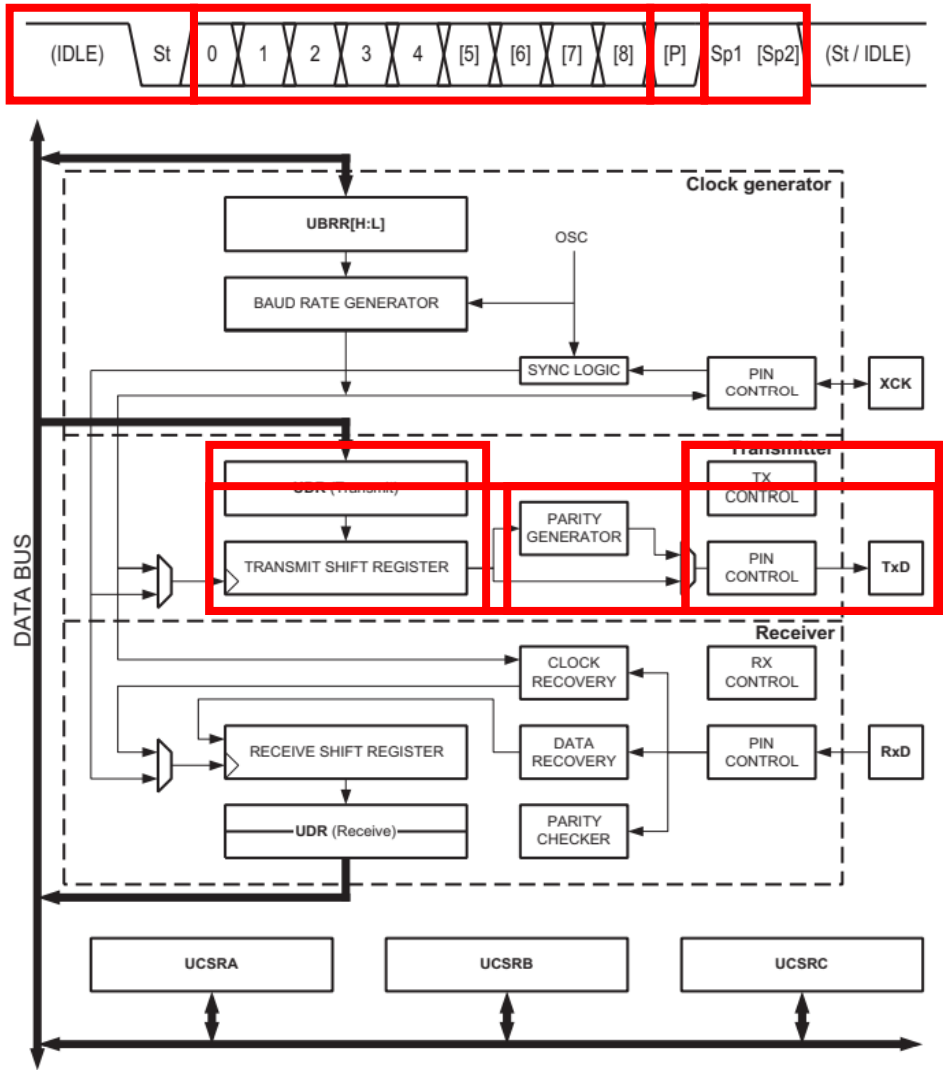
Figure 19-4. Frame formats.



St	Start bit, always low
(n)	Data bits (0 to 8)
P	Parity bit. Can be odd or even
Sp	Stop bit, always high
IDLE	No transfers on the communication line (RxDn or TxDn). An IDLE line must be high

USART – Send Process

- Data is placed into the UDRn register
  - This automatically starts the transmit process
  - This data is shifted into the Transmit Shift Register
  - Once the data is placed into the Shift Register, the UDRE flag on UCSRnA is set to a 1
  - The USART Tx Control is activated
- ‘Start Bit’ is sent by the USART
  - This is simply pulling down the line (0)
- ‘Data Bits’ are sent
  - Each Data bit is sent out of the Transmit Shift Register
  - The data bit is sent out over the TxD pin
  - The data bit is also sent to the parity generator to calculate the parity bit
- ‘Parity Bit’ is sent
  - The parity bit calculated during the transmission of the data bit is sent out over the TxD pin
- ‘Stop Bit(s)’ are sent
  - The TXCn flag in the UCSRnA register is set to a 1

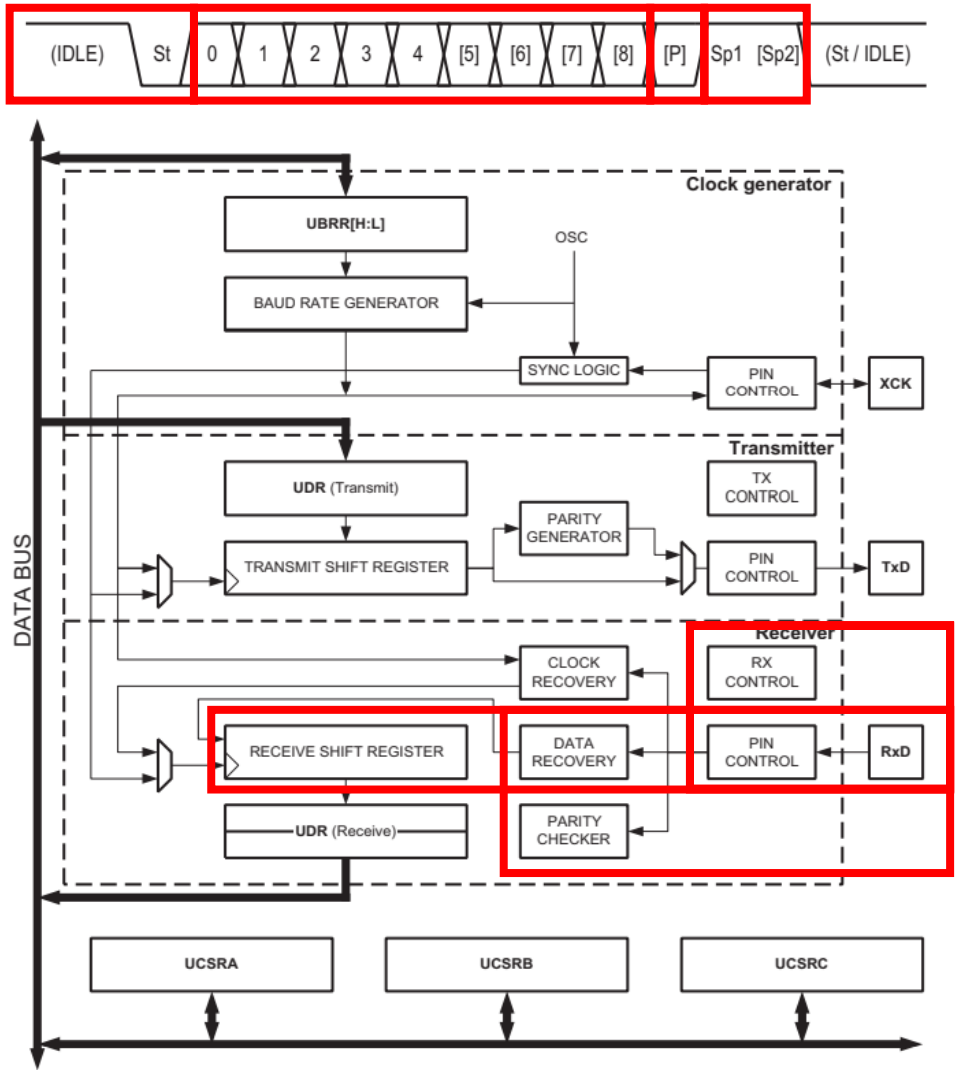


USART – Send Process

- Data is placed into the UDRn register
- Once the data is placed into the Shift Register, the UDRE flag on UCSRnA is set to a 1, indicating new data can be placed into the UDRn register
- The TXC flag in the UCSRnA register is set to a 1 to indicate the data transmission has been complete

USART – Receive Process

- ‘Start Bit’ received by the USART
  - This is simply pulling down the line (0)
  - The USART Rx Control is activated and starts listening
- ‘Data Bits’ are received
  - Each Data bit is transferred into the Receive Shift Register
  - This is a holding spot until all the data bits are received by the USART
- ‘Parity Bit’ is received
  - The USART automatically performs the calculation to check the integrity of the data
  - If there is an issue with the data, the UPE flag from the UCSRnA register is set to a 1
- ‘Stop Bit’ is received
  - At this point, the data will shift into the UDRn register
  - The RXC flag in the UCSRnA register is set to a 1



USART – Receive Process

- The RXC flag in the UCSRnA register is set to a 1 to indicate the data has been received
- If there are any issues with the integrity of the data (based on the calculation of the parity checker), the UPE flag from the UCSRnA register is set to a 1
- Read the data from the UDRn Register

Setting USART Registers

- When setting the UBRR register, simply define it as equal to the number stated on the Baud Rate Register table
  - There is no need to convert and enter in the number as a decimal or hexadecimal value!
  - The µC uses a 8MHz clock and we will not be using the double transmission speed for USART (U2X bit on UCSRnA)

```
// Set the UBBR1 register using the
   number

UBRR1 = 51;
```

Setting up USART on the Lab Board

- Ensure that the selector switches are set to the correct setting for each problem
  - Locate these under the seven-segment displays
  - For Problem 2 (Loopback), set the switch to LOOP
  - For Problem 4 (PC Communication), set the switches to UART & USB