

```

// Module-level Variables
// static uint8_t LocalRegisterImage = 0;

// void SR_Init(void)
    // set up port B by enabling the peripheral clock, waiting for the
    // peripheral to be ready and setting the direction
    // of PB0, PB1 & PB2 to output

    // Initialize Port B
    // Wait for clock to get ready
    // Set data direction and assign digital port on Port B
    // Start with the data & SCLK lines low and RCLK line high
// end

// uint8_t SR_GetCurrentRegister(void)
    // return LocalRegisterImage
// end

// void SR_Write(uint16_t NewValue)
    // Lower the register clock
    // Shift out the data while pulsing the serial clock
        // Isolate the MSB of NewValue
        // Bit 0 high if Bit 15 is high
        // Set bit 0 low if Bit 15 is low
        // Lower SCLK
        // Raise SCLK
        // Left shift NewValue by 1
    // Finish looping through bits in NewValue
    // Raise the register clock to latch the new data
// end

```