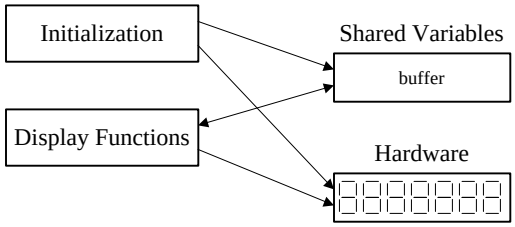
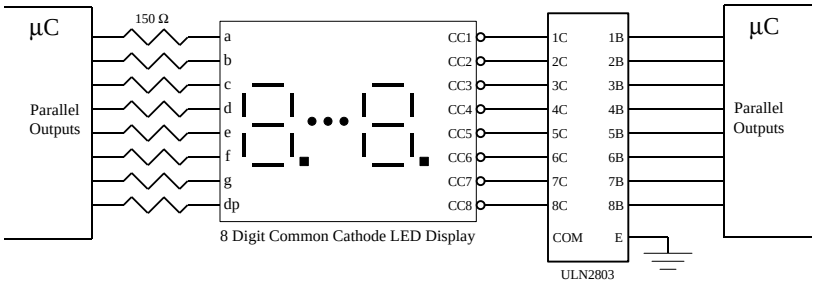
Display Software

Non-Multiplexed or Hardware Multiplexing



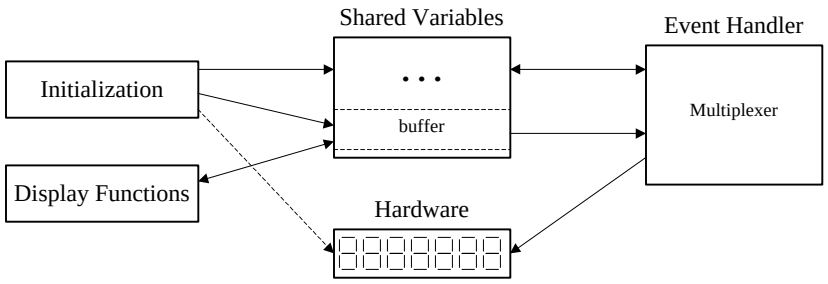
Multiplexed 7-Segment LED Interfacing

Software Multiplexing



Display Software

Software Multiplexing



Display Software

Software Multiplexing Pseudo Code

```
save registers

    display[current_column] = buffer[current_column]
    current_column = (current_column + 1) mod NUM_COLUMNS

restore registers
return
```

Display Software

Software Multiplexing Pseudo Code

display[current_column] = buffer[current_column]

~~write row pattern to 1 or more ports~~
~~write column pattern to 1 or more ports~~
~~for a brief period of time the wrong row pattern is on a column~~

turn off columns
write row pattern to 1 or more ports
write column pattern to 1 or more ports

- or -

turn off rows
write column pattern to 1 or more ports
write row pattern to 1 or more ports

Display Software

Software Multiplexing Pseudo Code with Blinking/Dimming

```
save registers

if (blink_dim_cnt < on_time) then
   display[current_ column] = buffer[current_ column]
   current_ column = (current_ column + 1) mod NUM_COLUMNS
else
   display = off
endif
blink_dim_cnt = (blink_dim_cnt + 1) mod (on_time + off_time)

restore registers
return
```

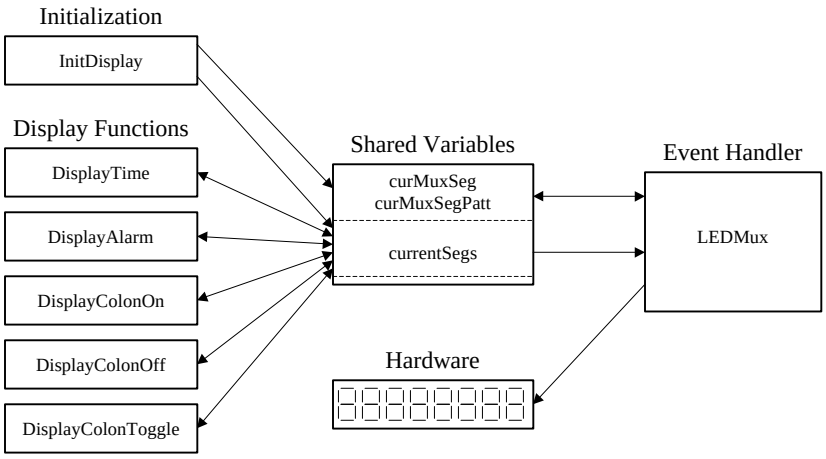
Display Software

Software Multiplexing Pseudo Code with Scrolling

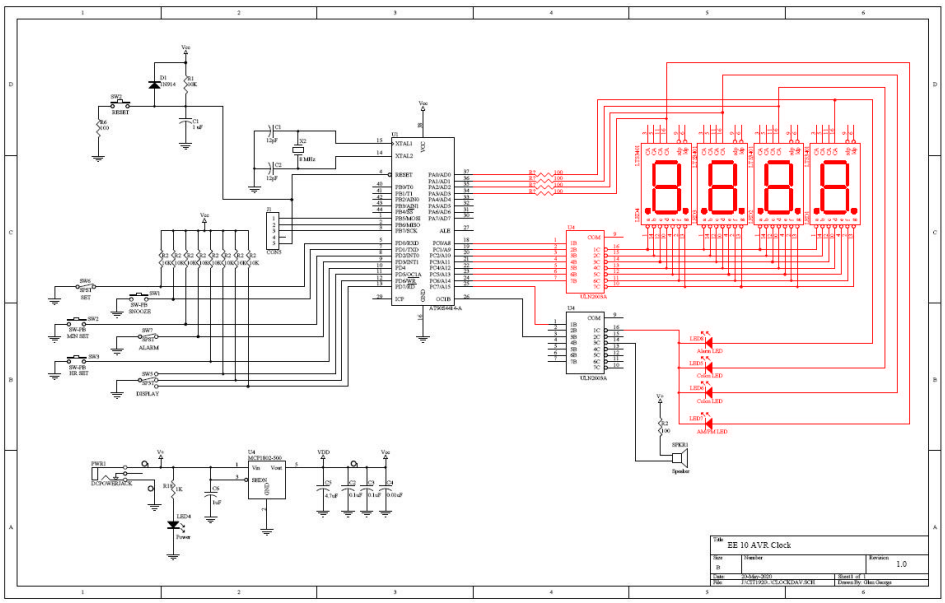
```
save registers

if (blink_dim_cnt < on_time) then
   display[current_ column] = buffer[current_ column + scroll_pos]
   current_ column = (current_ column + 1) mod NUM_COLUMNS
else
   display = off
endif
blink_dim_cnt = (blink_dim_cnt + 1) mod (on_time + off_time)

restore registers
return
```

Display Software Multiplexing Example

Alarm Clock Example



AVR Alarm Clock Schematic



Digit drivers/sources (rows) on PA_3 to PA_0 (Digit3 to Digit0)
Segment drivers/sinks (columns) on PC_7 to PC_0 (LEDs, g to a)

Display Multiplexing Example - Shared Variables

buffer (currentSegs)

h_{10a}	h_{1a}	m_{10a}	m_{1a}
h_{10b}	h_{1b}	m_{10b}	m_{1b}
h_{10c}	h_{1c}	m_{10c}	m_{1c}
h_{10d}	h_{1d}	m_{10d}	m_{1d}
h_{10e}	h_{1e}	m_{10e}	m_{1e}
h_{10f}	h_{1f}	m_{10f}	m_{1f}
h_{10g}	h_{1g}	m_{10g}	m_{1g}
AM/PM	colon	colon	alarm

muxing order:
a, b, …, g, LEDs

⇩

00000001
00000010
…
01000000
10000000

```
;the data segment

.dseg


currentSegs:    .BYTE  NUM_SEGS  ;buffer holding currently displayed segment patterns

curMuxSeg:      .BYTE  1         ;current segment number being multiplexed
curMuxSegPatt:  .BYTE  1         ;current segment output pattern
```