LPC1114 display driver

LPC1114 display driver

Example:
LPC1114 driving a 4 digit LED display

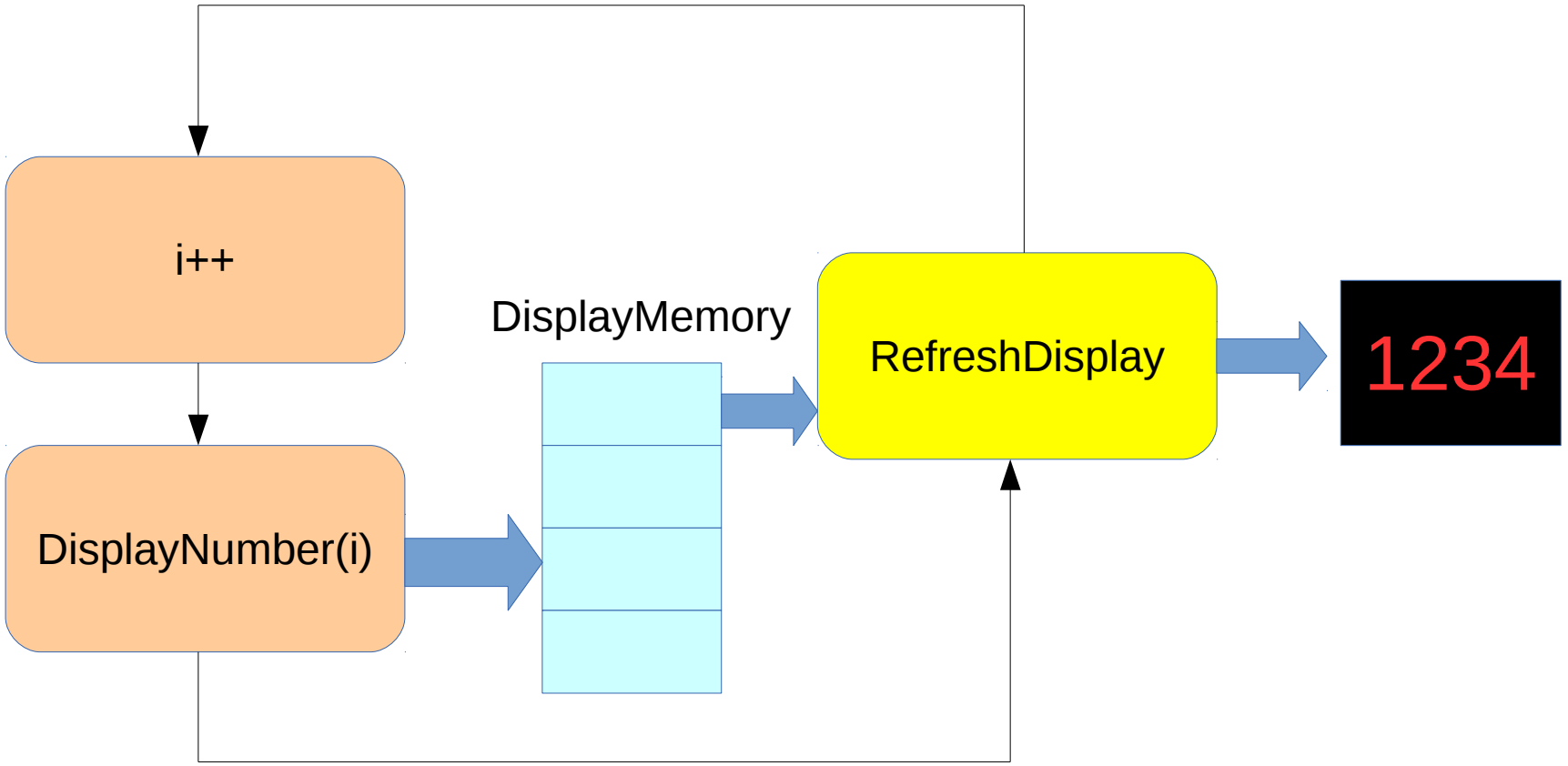
LPC1114 display driver

The main function

1) Perform initial IO configuration
2) Increment a number (reset to 0 after 9999)
3) Call on DisplayNumber
4) Call on RefreshDisplay
5) Repeat 2-4

LPC1114 display driver

Main loop



LPC1114 display driver

```
int main()
{
   ConfigPins();
   int i=0;

   while(1)
   {
      DisplayNumber(i++);
      RefreshDisplay();
      if (i > 9999)
         i=0;
   }
}
```

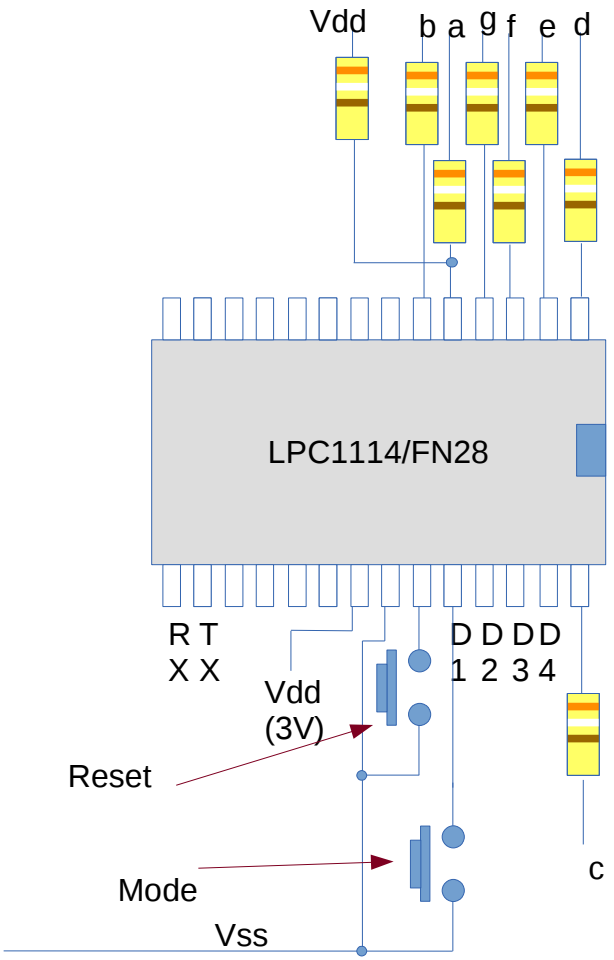
LPC1114 display driver

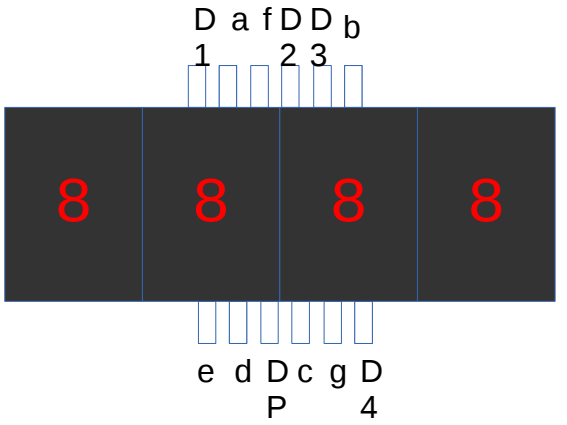
ConfigPins function

1) Enable IO configuration function
2) Enable GPIO ports
3) Set port bit function
4) Set port bit direction

LPC1114 display driver

Wiring





LPC1114 display driver

```
// Association of bits to segments due to wiring
// Display is on Port 0
#define SEG_A BIT5
#define SEG_B BIT6
#define SEG_C BIT7
#define SEG_D BIT8
#define SEG_E BIT9
#define SEG_F BIT10
#define SEG_G BIT11
#define DIG_1 BIT1
#define DIG_2 BIT2
#define DIG_3 BIT3
#define DIG_4 BIT4
```

LPC1114 display driver

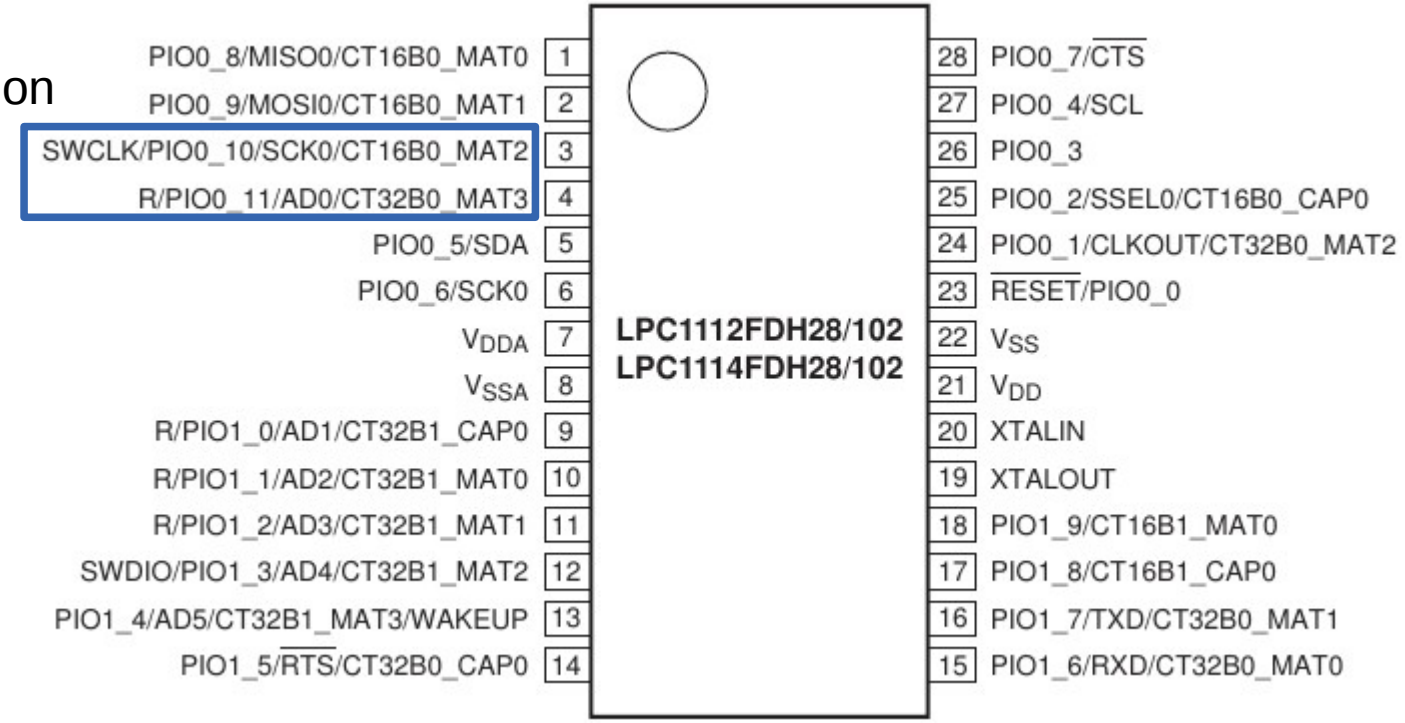
```
const short digits[]=
{ \
 SEG_A | SEG_B | SEG_C | SEG_D | SEG_E | SEG_F         ,\
         SEG_B | SEG_C                                 ,\
 SEG_A | SEG_B |         SEG_D | SEG_E |         SEG_G,\
 SEG_A | SEG_B | SEG_C | SEG_D |                 SEG_G,\
         SEG_B | SEG_C |                 SEG_F | SEG_G,\
 SEG_A |         SEG_C | SEG_D |         SEG_F | SEG_G,\
 SEG_A |         SEG_C | SEG_D | SEG_E | SEG_F | SEG_G,\
 SEG_A | SEG_B | SEG_C                         ,\
 SEG_A | SEG_B | SEG_C | SEG_D | SEG_E | SEG_F | SEG_G,\
 SEG_A | SEG_B | SEG_C | SEG_D |         SEG_F | SEG_G \
 };
```

LPC1114 display driver

```
void ConfigPins()
{
   SYSAHBCLKCTRL |= BIT6 + BIT16; // Turn on clock for
                                  // GPIO and IOCON
   // Make all of the segment and digit bits outputs
   GPIO0DIR = SEG_A | SEG_B | SEG_C | SEG_D | SEG_E \ |
      SEG_F | SEG_G | DIG_1 | DIG_2 | DIG_3 | DIG_4;
   // Turn off (make high) all display digits
   GPIO0DATA = DIG_1 | DIG_2 | DIG_3  | DIG_4;
   // Make Port 0 bit 5 behave as a generic
   // output port (open drain)
   IOCON_PIO0_5 |= BIT8;
   // Make Port 0 bit 10 behave as a generic I/O port
   IOCON_SWCLK_PIO0_10  = 1;
   // Make Port 0 bit 11 behave as a generic I/O port
   IOCON_R_PIO0_11  = 1;
}
```

LPC1114 display driver

Default function
Not GPIO



LPC1114 display driver

(Refer to tables: 69,82,85)

LPC1114 display driver

```
void RefreshDisplay(void)
{
   // Turn on (make low) the desired
   // digit and blank all segments
   GPIO0DATA = DIG_1 | DIG_2 | DIG_3;
   // Set the relevant segment bits
   GPIO0DATA |= DisplayMemory[0];
   // Wait for display to light up
   delay(1000);

   // repeat for next digit
   GPIO0DATA = DIG_1 | DIG_2 | DIG_4;
   GPIO0DATA |= DisplayMemory[1];
   delay(1000);
```

LPC1114 display driver

```
    // repeat for next digit
    GPIO0DATA = DIG_1 | DIG_3 | DIG_4;
    GPIO0DATA |= DisplayMemory[2];
    delay(1000);


    // repeat for next digit
    GPIO0DATA = DIG_2 | DIG_3 | DIG_4;
    GPIO0DATA |= DisplayMemory[3];
    delay(1000);
}
```

LPC1114 display driver

```
void DisplayNumber(int Number)
{
   DisplayMemory[0]=digits[Number % 10];
   Number = Number / 10;
   DisplayMemory[1]=digits[Number % 10];
   Number = Number / 10;
   DisplayMemory[2]=digits[Number % 10];
   Number = Number / 10;
   DisplayMemory[3]=digits[Number % 10];
}
```

LPC1114 display driver

- Demonstration of display
- Vary timings
- Task: recode for display in Hex
  - New segment codes
  - Create DisplayHex (variation of Display Number)
  - Test