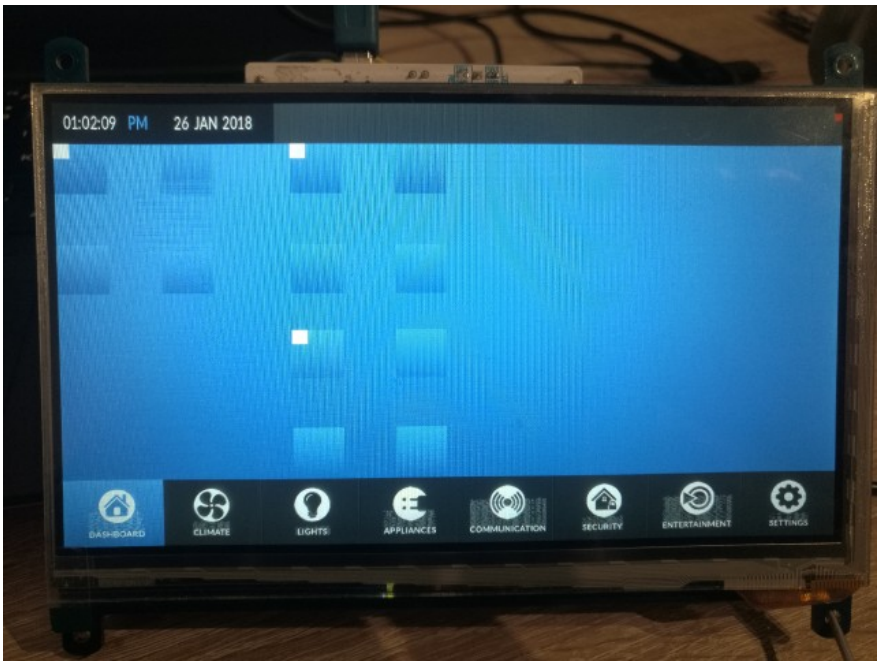
Hi, I've been working on a UI for past couple weeks,

I've successfully created functionality for drawing bitmaps (with or without alpha), anti-aliased fonts and solid rectangles / lines using Chrom-ART / DMA2D

I am however currently unsure how to do rectangle fills with transparency. I assumed that DMA2D_R2M would accept a ARGB color and layer configurations, however that was not the case.

Can someone explain in rough lines how I can fill a rectangle, that contains an alpha channel, over an existing drawn area?

Thanks



My current bitmap/font character drawing function:

```
void LCD::DMA_DrawBitmap(void *pSrc, void *pDst, uint32_t xSize, uint32_t ySize, uint32_t ColorMode, uint32_t inputAlpha, uint32_t inputOffset) {
    hDma2dHandler.Init.Mode = DMA2D_M2M_BLEND;
    hDma2dHandler.Init.ColorMode = hLtdcHandler.LayerCfg[0].PixelFormat;
    hDma2dHandler.Init.OutputOffset = SCREEN_WIDTH - xSize;


    hDma2dHandler.LayerCfg[0].AlphaMode = DMA2D_NO_MODIF_ALPHA;
    hDma2dHandler.LayerCfg[0].InputColorMode = hLtdcHandler.LayerCfg[0].PixelFormat;
    hDma2dHandler.LayerCfg[0].InputAlpha = 0;
    hDma2dHandler.LayerCfg[0].InputOffset = SCREEN_WIDTH - xSize;

    /* Foreground Configuration */
    hDma2dHandler.LayerCfg[1].AlphaMode = DMA2D_NO_MODIF_ALPHA;
    hDma2dHandler.LayerCfg[1].InputAlpha = inputAlpha;
    hDma2dHandler.LayerCfg[1].InputColorMode = ColorMode;
    hDma2dHandler.LayerCfg[1].InputOffset = inputOffset;
    hDma2dHandler.Instance = DMA2D;

    /* DMA2D Initialization */
    if (HAL_DMA2D_Init(&hDma2dHandler) == HAL_OK) {
        if (HAL_DMA2D_ConfigLayer(&hDma2dHandler, 0) == HAL_OK && HAL_DMA2D_ConfigLayer(&hDma2dHandler, 1) == HAL_OK) {
            if (HAL_DMA2D_BlendingStart(&hDma2dHandler, (uint32_t) pSrc, (uint32_t) pDst, (uint32_t) pDst, xSize, ySize) == HAL_OK) {
                HAL_DMA2D_PollForTransfer(&hDma2dHandler, 500);
            }
        }
    }
}
```

And rectangle fill function:

```
void LCD::DMA_FillRect(void *pSrc, void *pDst, uint32_t xSize, uint32_t ySize, uint32_t ColorMode) {
    /* Configure the DMA2D Mode, Color Mode and output offset */
    hDma2dHandler.Init.Mode = DMA2D_M2M_PFC;
    hDma2dHandler.Init.ColorMode = hLtdcHandler.LayerCfg[0].PixelFormat;
    hDma2dHandler.Init.OutputOffset = SCREEN_WIDTH - xSize;

    /* Foreground Configuration */
    hDma2dHandler.LayerCfg[1].AlphaMode = DMA2D_NO_MODIF_ALPHA;
    hDma2dHandler.LayerCfg[1].InputAlpha = 0xFF;
    hDma2dHandler.LayerCfg[1].InputColorMode = ColorMode;
    hDma2dHandler.LayerCfg[1].InputOffset = 0;
    hDma2dHandler.Instance = DMA2D;

    /* DMA2D Initialization */
    if (HAL_DMA2D_Init(&hDma2dHandler) == HAL_OK) {
        if (HAL_DMA2D_ConfigLayer(&hDma2dHandler, 1) == HAL_OK) {
            if (HAL_DMA2D_Start(&hDma2dHandler, (uint32_t) pSrc, (uint32_t) pDst, xSize, ySize) == HAL_OK) {
                HAL_DMA2D_PollForTransfer(&hDma2dHandler, 500);
            }
        }
    }
}
```