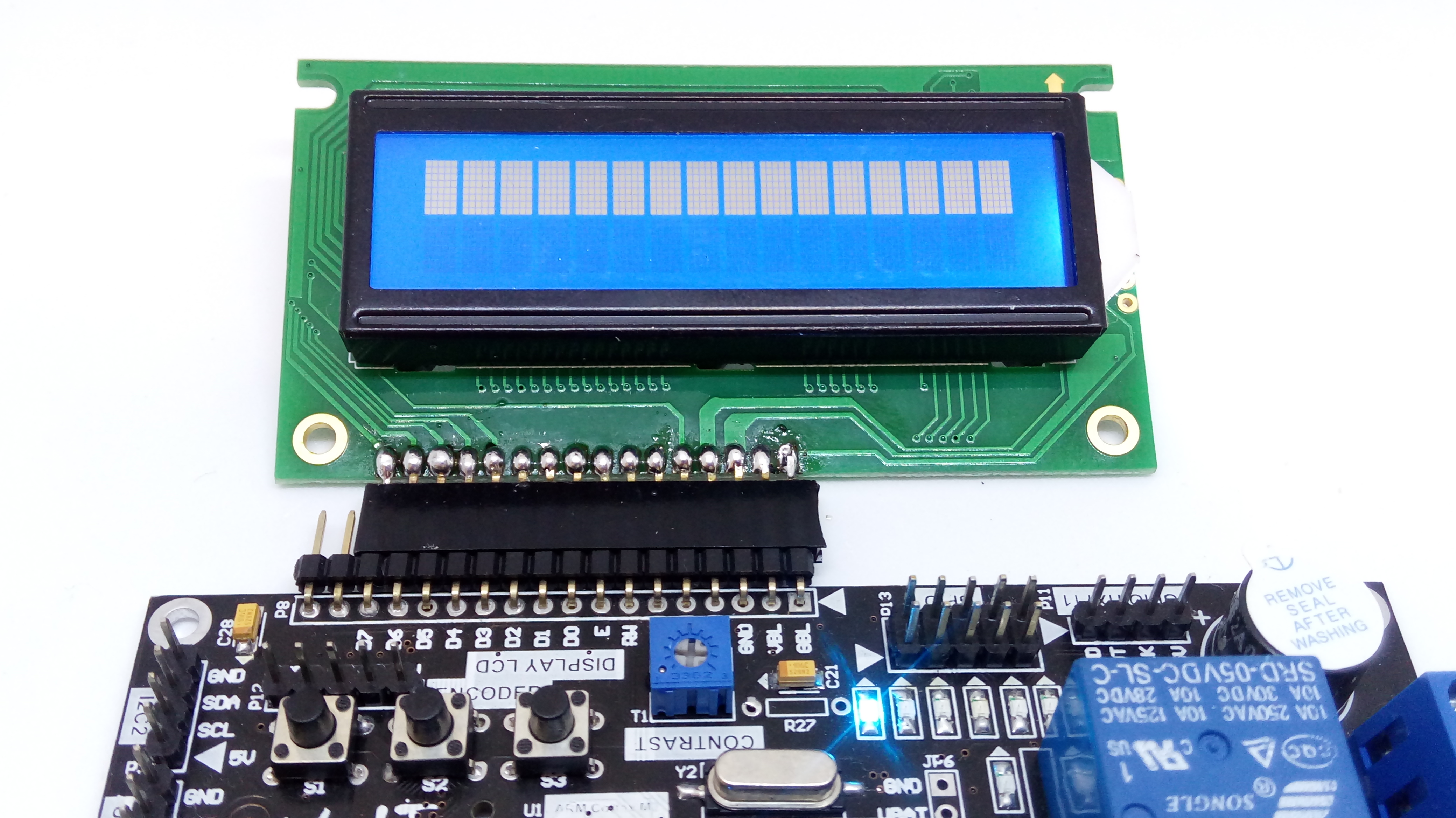
**Usando Display LCD Texto**

Neste exemplo utilizamos um display de LCD 16x2 conectado ao kit ARM via porta IO e com 8 bits para barramentos de dados. Porém, podemos usar outros tamanhos de display como 16x1, 16x2, 16x4, 20x4, e outros tipos de portas de comunicações como IO de 4ibts de dados ou porta I2C com o pcf8574.

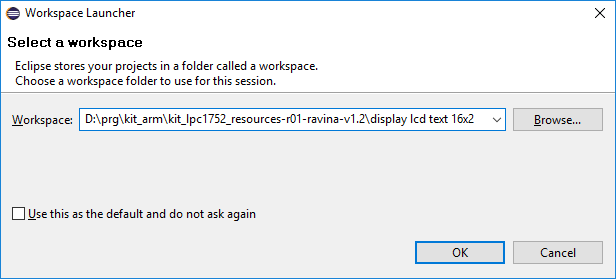
**Conexão com o Kit ARM**

Somente encaixe o display no conector P8. **Atenção**: Alguns displays permitem jumpear o tensão do LED backlight, veja que o mesmo está conforme os padrões:



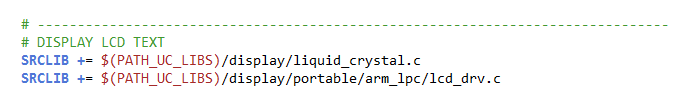
**Programa Exemplo**

Abra o exemplo da pasta “display lcd text 16x2” com o Eclipse:

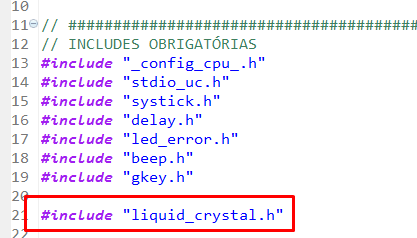


No makefile, na secção de inserção de fontes, foi incluindo as seguintes fontes:

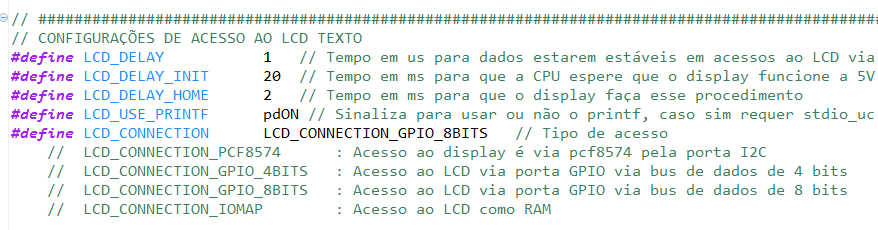
* liquid\_crystal.c: Controlador do display texto;
* lcd\_drv.c: Driver ARM de acesso ao display. Cada processador tem seu próprio driver.



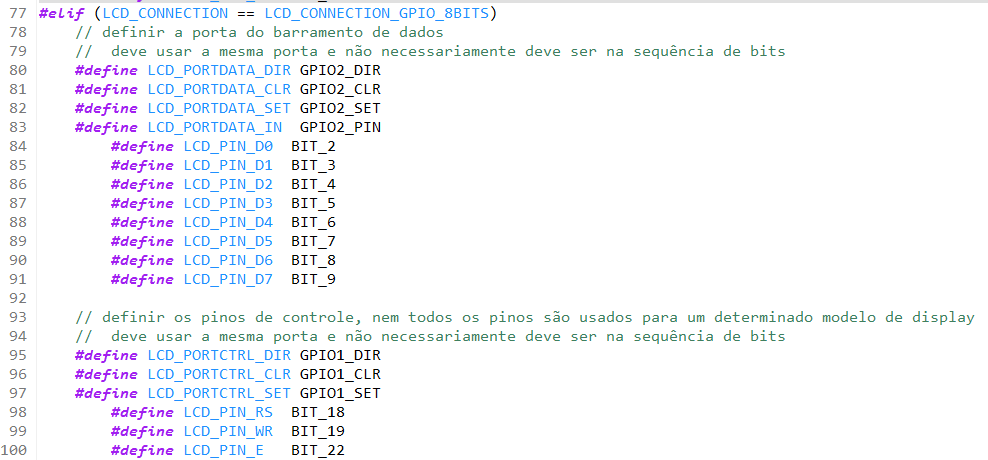
Em app.h foi adicionado o include do controlador do display:



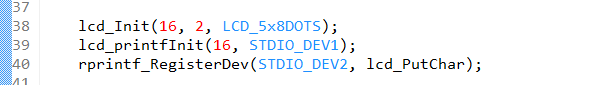
Em \_config\_cpu\_display.h é feito a configuração do tipo de acesso ao display, neste caso foi usado a porta IO com barramento de 8bits de dados. Contudo, outros modos de acesso podem ser configurados:



Caro queria trocar os pinos de acessos do display, suas definições estão descritas mais embaixo do arquivo \_config\_cpu\_display.h:



Em main.c dentro do int main(void), é inicializado a porta o controlador do display a trabalhar por 16 colunas e 2 linhas, porém, podemos usar outras quantidades de colunas e linhas. Também, inicializamos os módulos responsáveis para impressão do printf no LCD:



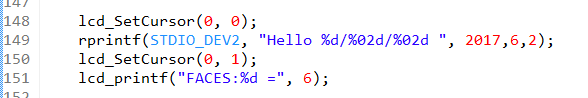
Neste exemplo estamos usando dois modos de uso do printf, um modo usando a função rprintf da stdio\_uc, e outro modo usando a função nativa do display lcd\_printf. Então, estamos inicializando os dois modos.

**Compile e Execute**

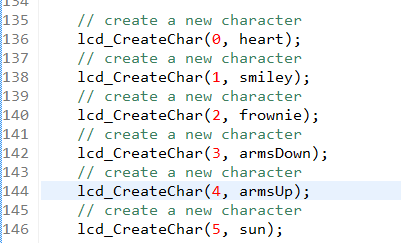
Com tudo instalado e configurado, compile a aplicação, grave e execute no ARM. Ajuste o contraste do display via trimpot T1:



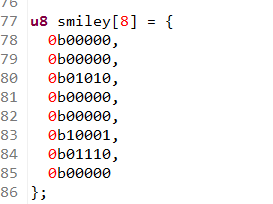
Neste exemplo estamos usando os dois modos de uso do printf e abaixo é exibido os dois modos de impressão.



Os caracteres especiais foram criados no código abaixo, e neste tipo de display são usados no máximo 6 deles:



Os formatos dos caracteres são descritos em vetores, como este a seguir:



E depois imprimimos nos locais corretos usando a função de posicionamento lcd\_SerCursor e a função de impressão de caracteres lcd\_PutChar:

