**Medindo Distâncias com Ultrassom**

Neste exemplo vamos fazer leitura de distância usando o módulo ultrassom hc-sr04. No kit ARM há somente uma entrada para este tipo de sensor, entretanto é possível ler mais de um sensor ao mesmo tempo utilizando outras portas IO do ARM.



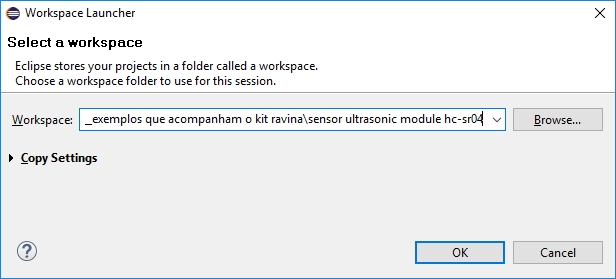
**Conexão com o Kit ARM**

No kit ARM o conector P14 é dedicado para uso deste módulo de sensor, conecte respeitando os sinais e a tensão de alimentação como descrito na seguinte tabela:

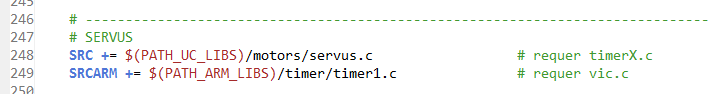
|  |  |  |
| --- | --- | --- |
| **Módulo** | **Conector** | **Pino ARM** |
| 5V | P15 pino 1 |  |
| Trigger | P15 pino 2 | 70 (P1\_14) |
| Echo | P15 pino 3 | 69 (P1\_15) |
| GND | P15 pino 4 |  |

**Programa Exemplo**

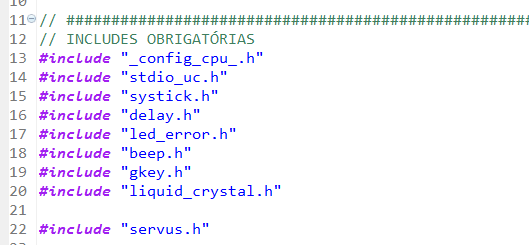
Abra o exemplo da pasta “sensor ultrasonic module hc-sr04” com o Eclipse:



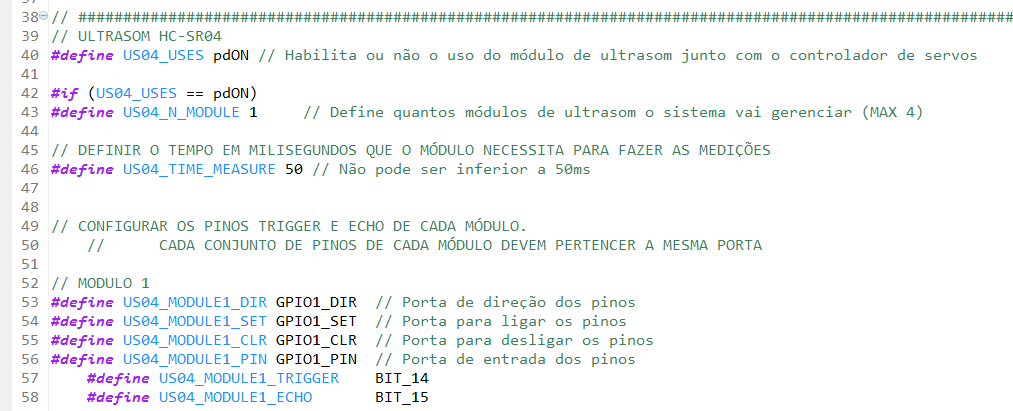
Geralmente usamos o sensor de ultrassom junto com uso de servos motores, então foi criada uma biblioteca chamada servus.c que gerencia os dois tipos de dispositivos. A biblioteca servus.c requer um timer dedicado do processador para controlar os tempos de acionamentos e leituras de distancias dos ultrassons, logo foi usado o timer1 do ARM. No makefile, na secção de inserção de fontes, foi incluindo as fontes o servus.c e timer1.c:



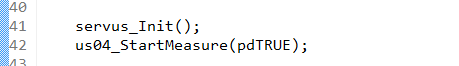
Em app.h foi adicionado a include do servus para usar as suas funções na aplicação:



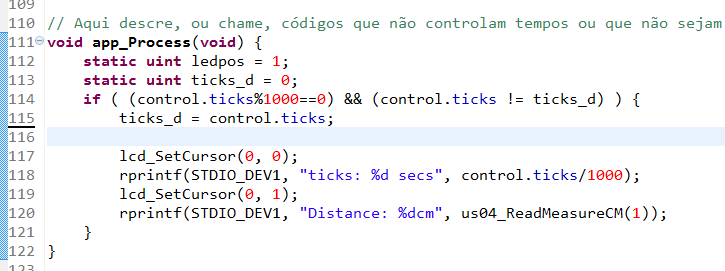
Em \_config\_cpu\_servus.h determinamos a quantidade de módulos de ultrassom que o servus vai gerenciar, e quais são os pinos IOs do ARM que vão controlar os módulos:



Em main.c dentro do int main(void), é inicializado o servus e especificamos para que as leituras dos sensores de ultrassom sejam feitas de forma continua, sem que seja necessário ficar enviado o comando de leituras:



Em app\_Process é lido o valor da distância do módulo 1 e mostrado o seu valor no display LCD:



**Compile e Execute**

Com tudo instalado e configurado, compile a aplicação, grave e execute no ARM. Após a execução os valores das distâncias serão mostrados no display LCD.