MCUXpresso IDE helps users to create easy to use Eclipse based environment. It includes Arm Cortex-M cores, LPC and kinrtis microcontroller. It offers advanced compiling, debugging and editing features. It also provides code trace, debugging views, multicore debugging and integrated configuration tools.

**Features of MCUXpresso**

* It provides free toolchain to developers without any restriction on code or the size of debugging.
* It provides powerful interface with power measurement, profiling on supported boards, multicore capable debugger and many more.
* It provides complete C/C++ integrated development environment

Advance project wizard:

* Trace functionality
* Link Server Power Measurement
* MCUXpresso Configuration Tools

Supported Debug Probes:

There is built in support for three debug solutstions.

* Native LinkServer
* P&E Micro
* Segger J-Link

Program:

|  |
| --- |
| #if defined (\_\_USE\_LPCOPEN) |
|  | #if defined(NO\_BOARD\_LIB) |
|  | #include "chip.h" |
|  | #else |
|  | #include "board.h" |
|  | #endif |
|  | #endif |
|  |  |
|  | #include <cr\_section\_macros.h> |
|  |  |
|  | // TODO: insert other include files here |
|  |  |
|  | // TODO: insert other definitions and declarations here |
|  |  |
|  | int main(void) { |
|  |  |
|  | #if defined (\_\_USE\_LPCOPEN) |
|  | // Read clock settings and update SystemCoreClock variable |
|  | SystemCoreClockUpdate(); |
|  | #if !defined(NO\_BOARD\_LIB) |
|  | // Set up and initialize all required blocks and |
|  | // functions related to the board hardware |
|  | Board\_Init(); |
|  | // Set the LED to the state of "On" |
|  | Board\_LED\_Set(0, true); |
|  | //Board\_LED\_Set(1, true); |
|  | // Board\_LED\_Set(2, true); |
|  | #endif |
|  | #endif |
|  |  |
|  | // TODO: insert code here |
|  |  |
|  | // Force the counter to be placed into memory |
|  | volatile static int i = 0 ; |
|  | // Enter an infinite loop, just incrementing a counter |
|  | while(1) { |
|  | int k,j; |
|  | Board\_LED\_Set(0,false); |
|  | Board\_LED\_Set(1,true); |
|  | Board\_LED\_Set(2,true); |
|  | for(k=0;k<5000;k++) |
|  | { |
|  | for(j=0;j<2000;j++) |
|  | {} |
|  | } |
|  |  |
|  | Board\_LED\_Set(0,true); |
|  | Board\_LED\_Set(1,false); |
|  | Board\_LED\_Set(2,true); |
|  | for(k=0;k<5000;k++) |
|  | { |
|  | for(j=0;j<2000;j++) |
|  | {} |
|  | } |
|  | Board\_LED\_Set(0,true); |
|  | Board\_LED\_Set(1,true); |
|  | Board\_LED\_Set(2,false); |
|  | for(k=0;k<5000;k++) |
|  | { |
|  | for(j=0;j<2000;j++) |
|  | {} |
|  | } |
|  |  |
|  |  |
|  |  |
|  |  |
|  | i++ ; |
|  | } |
|  | return 0 ; |
|  | } |