

| Description | Resource | Path | Location | Type |
|--|--------------|---------|---------------|---------------|
| #1376-D (MISRA-C:2004 1.1/R) Ensure strict ANSI C mode (-ps) is enabled | MyP_Types.h | | | |
| /My_LCD/libraries | | line 11 | C/C++ Problem | |
| #1379-D (MISRA-C:2004 5.2/R) Identifiers in an inner scope shall not use the same name as an identifier in an outer scope, and therefore hide that identifier ("arr") | calculator.h | /My_LCD | line 26 | C/C++ Problem |
| #1379-D (MISRA-C:2004 5.2/R) Identifiers in an inner scope shall not use the same name as an identifier in an outer scope, and therefore hide that identifier ("k") | calculator.c | /My_LCD | line 19 | C/C++ Problem |
| #1383-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call | calculator.c | /My_LCD | line 17 | C/C++ Problem |
| #1383-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call | main.c | /My_LCD | line 17 | C/C++ Problem |
| #1384-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call ("get_operationseration") | key.c | /My_LCD | line 75 | C/C++ Problem |
| #1384-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call ("init_port") | key.c | /My_LCD | line 17 | C/C++ Problem |
| #1384-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call ("make_input_UP") | key.c | /My_LCD | line 29 | C/C++ Problem |
| #1384-D (MISRA-C:2004 8.1/R) Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call ("make_output") | key.c | /My_LCD | line 19 | C/C++ Problem |
| #1386-D (MISRA-C:2004 8.6/R) Functions shall be declared at file scope (function "get_operationseration") | key.c | /My_LCD | line 75 | C/C++ Problem |
| #1386-D (MISRA-C:2004 8.6/R) Functions shall be declared at file scope (function "init_port") | key.c | /My_LCD | line 17 | C/C++ Problem |
| #1386-D (MISRA-C:2004 8.6/R) Functions shall be declared at file scope (function "make_input_UP") | key.c | /My_LCD | line 29 | C/C++ Problem |
| #1386-D (MISRA-C:2004 8.6/R) Functions shall be declared at file scope (function "make_output") | key.c | /My_LCD | line 19 | C/C++ Problem |
| #1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness | Dio.c | /My_LCD | line 21 | C/C++ Problem |

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness Dio.c /My_LCD line 70 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness Dio.c /My_LCD line 71 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness Dio.c /My_LCD line 72 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 23 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 32 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 117C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 135C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 137C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 139C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 141C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 143C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 151C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 153C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 155C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 157C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 159C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness calculator.c /My_LCD line 160C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness key.c /My_LCD line 61 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness key.c /My_LCD line 65 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness key.c /My_LCD line 67 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness key.c /My_LCD line 73 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness main.c /My_LCD line 23 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if it is not a conversion to a wider integer type of the same signedness main.c /My_LCD line 24 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is complex calculator.c /My_LCD
line 23 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is complex calculator.c /My_LCD
line 42 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is complex key.c /My_LCD line 67
C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is not constant and is a function argument
calculator.c /My_LCD line 106C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is not constant and is a function argument
key.c /My_LCD line 64 C/C++ Problem

#1393-D (MISRA-C:2004 10.1/R) The value of an expression of integer type shall not be implicitly converted to a different underlying type if the expression is not constant and is a function argument
main.c /My_LCD line 28 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 85 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 88 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 89 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 93 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 94 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 99 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 102C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 103C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 107C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 108C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 113C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 116C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 117C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 121C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 122C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 127C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 130C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 131C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 135C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 136C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 140C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 143C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 144C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 148C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 149C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 154C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 157C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 158C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 162C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 163C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 175C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 179C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 183C/C++ Problem

#1465-D (MISRA-C:2004 14.1/R) There shall be no unreachable code Dio.c /My_LCD line 140
C/C++ Problem

#1416-D (MISRA-C:2004 15.2/R) An unconditional break statement shall terminate every non-empty switch clause Dio.c /My_LCD line 67 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 175C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 163C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 179C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 162C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 183C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 158C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 187C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 157C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 191C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 154C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 195C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 149C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 208C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 148C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 212C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 144C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 219C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 143C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 223C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 140C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 230C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 136C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 234C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 135C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 241C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 131C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 245C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 130C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 252C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 127C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 256C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 122C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 263C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 121C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 267C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 117C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 116C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 113C/C++ Problem

#1500-D (MISRA-C:2004 12.10/R) The comma operator shall not be used calculator.c /My_LCD line 32 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed calculator.c /My_LCD line 34 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 108C/C++ Problem

#1484-D (MISRA-C:2004 6.1/R) The plain char type shall be used only for the storage and use of character values calculator.c /My_LCD line 42 C/C++ Problem

#179-D variable "j" was declared but never referenced calculator.c /My_LCD line 19 C/C++ Problem

#1505-D (MISRA-C:2004 13.5/R) The three expressions of a for statement shall be concerned only with loop control calculator.c /My_LCD line 32 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.c /My_LCD line 47 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed calculator.c /My_LCD line 58 C/C++ Problem

#1506-D (MISRA-C:2004 13.6/R) Numeric variables being used within a for loop for iteration counting shall not be modified in the body of the loop calculator.c /My_LCD line 65 C/C++ Problem

#1506-D (MISRA-C:2004 13.6/R) Numeric variables being used within a for loop for iteration counting shall not be modified in the body of the loop calculator.c /My_LCD line 72 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed calculator.c /My_LCD line 83 C/C++ Problem

#1506-D (MISRA-C:2004 13.6/R) Numeric variables being used within a for loop for iteration counting shall not be modified in the body of the loop calculator.c /My_LCD line 90 C/C++ Problem

#1506-D (MISRA-C:2004 13.6/R) Numeric variables being used within a for loop for iteration counting shall not be modified in the body of the loop calculator.c /My_LCD line 97 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 107C/C++ Problem

#1471-D (MISRA-C:2004 15.3/R) The final clause of a switch statement shall be the default clause
calculator.c /My_LCD line 54 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 103C/C++ Problem

#1484-D (MISRA-C:2004 6.1/R) The plain char type shall be used only for the storage and use of character values calculator.c /My_LCD line 117C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.c /My_LCD line 132C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 102C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 99 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed calculator.c /My_LCD line 137C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 94 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 93 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 89 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.c /My_LCD line 148C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 88 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 85 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 69 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 68 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 61 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 60 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 52 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.c /My_LCD line 17 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 51 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 42 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 41 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.h /My_LCD line 25 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 32 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.h /My_LCD line 29 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void calculator.h /My_LCD line 30 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 31 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void key.c /My_LCD line 15 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 22 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed Dio.c /My_LCD line 21 C/C++ Problem

#225-D function "init_port" declared implicitly key.c /My_LCD line 17 C/C++ Problem

#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions calculator.c /My_LCD line 153C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression key.c /My_LCD line 81 C/C++ Problem

#225-D function "make_output" declared implicitly key.c /My_LCD line 19 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 24 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 25 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 26 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 27 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c /My_LCD line 267C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c /My_LCD line 263C/C++ Problem

#225-D function "make_input_UP" declared implicitly key.c /My_LCD line 29 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 35 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 36 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 37 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 38 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 40 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 41 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 42 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 43 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 45 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 46 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 47 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 48 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 50 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 51 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 52 C/C++ Problem

#1485-D (MISRA-C:2004 6.2/R) signed and unsigned char type shall be used only for the storage and use of numeric values key.c /My_LCD line 53 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void key.c /My_LCD line 57 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 256 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 63 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression key.c
/My_LCD line 63 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 252C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 245C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 241C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 67 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 234C/C++ Problem

#1484-D (MISRA-C:2004 6.1/R) The plain char type shall be used only for the storage and use of character values key.c /My_LCD line 70 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 230C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 73 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 223C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 219C/C++ Problem

#225-D function "get_operationseration" declared implicitly key.c /My_LCD line 75 C/C++ Problem

#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed key.c /My_LCD line 81 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void key.h /My_LCD line 17 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void key.h /My_LCD line 18 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void lcd.h /My_LCD line 25 C/C++ Problem

#93-D identifier-list parameters may only be used in a function definition lcd.h /My_LCD
line 29 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 212C/C++ Problem

#1430-D (MISRA-C:2004 19.10/R) In the definition of a function-like macro each instance of a parameter shall be enclosed in parentheses unless it is used as the operand of # or ##. The compiler cannot determine if the parameter "Register" is used as an operand of # or ##. bitwise_operations.h
/My_LCD/libraries line 21 C/C++ Problem

#1430-D (MISRA-C:2004 19.10/R) In the definition of a function-like macro each instance of a parameter shall be enclosed in parentheses unless it is used as the operand of # or ##. The compiler cannot determine if the parameter "Register" is used as an operand of # or ##. bitwise_operations.h
/My_LCD/libraries line 22 C/C++ Problem

#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void main.c /My_LCD line 17 C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 208C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 195C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 191C/C++ Problem

#1395-D (MISRA-C:2004 10.3/R) The value of a complex expression of integer type shall only be cast to a type of the same signedness that is no wider than the underlying type of the expression Dio.c
/My_LCD line 187C/C++ Problem

#1465-D (MISRA-C:2004 14.1/R) There shall be no unreachable code main.c /My_LCD line 33
C/C++ Problem