MISRA-2004 REPORT

n	Description	Resource	Location
1	#1385-D (MISRA-C:2004 8.2/R) Whenever an object or function is declared or defined, its type shall be explicitly stated	ADC.c	line 112
2	#1386-D (MISRA-C:2004 8.6/R) Functions shall be declared at file scope (function "sel")	ADC.c	line 47
3	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	ADC.c	line 45
4	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	ADC.c	line 59
5	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 54
6	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 60
7	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 72
8	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 78
9	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 84
10	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 91
11	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 100
12	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 103
13	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	ADC.c	line 109
14	#1421-D (MISRA-C:2004 16.8/R) All exit paths from a function with non-void return type shall have an explicit return statement with an expression (function "ISR")	ADC.c	line 114
15	#225-D function "sel" declared implicitly	ADC.c	line 47
16	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 37
17	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 35
18	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 32
19	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 29
20	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 26
21	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 23
22	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 20
23	#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 "REG" is used as an operand of # or ##.	common_macros.h	line 17
24	#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 "num" is used as an operand of # or ##.	common_macros.h	line 29
25	#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 "num" is used as an operand of # or ##.	common_macros.h	line 26
26	#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 "BIT" is used as an operand of # or ##.	common_macros.h	line 37
27	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 38
28	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 48
29	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 58
30	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 68
31	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 102
32	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 112
33	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 122
34	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	GPIO.c	line 132
35	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 34
36	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 38
37	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 44
38	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 48
39	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 54
40	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 58
41	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 64
42	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 68
43	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 98
44	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 102
45	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 108
46	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 112
47	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 118
48	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 122
49	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 128
50	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 132
51	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 160
52	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 162
53	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 164
54	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 166
55	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 177
56	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 179
57	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 181
58	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	GPIO.c	line 183
59	#1-D last line of file ends without a newline	I2C.c	line 192
60	#1397-D (MISRA-C:2004 10.5/R) If the bitwise operators ~ and << are applied to an operand of underlying type unsigned char or unsigned short, the result shall be immediately cast to the underlying type of the operand	I2C.c	line 133
61	#1404-D (MISRA-C:2004 12.4/R) The right-hand operand of a logical && or || operator shall not contain side effects	I2C.c	line 138
62	#1404-D (MISRA-C:2004 12.4/R) The right-hand operand of a logical && or || operator shall not contain side effects	I2C.c	line 142
63	#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions	I2C.c	line 138
64	#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions	I2C.c	line 142
65	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 11
66	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 15
67	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 19
68	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 23
69	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 27
70	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 35
71	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 37
72	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 42
73	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 44
74	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 52
75	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 62
76	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 64
77	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 69
78	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 71
79	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 80
80	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 113
81	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 124
82	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 138
83	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 142
84	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 156
85	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 160
86	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 174
87	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 178
88	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	I2C.c	line 191
89	#1-D last line of file ends without a newline	I2C.h	line 97
90	#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions	lcd.c	line 167
91	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 49
92	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 50
93	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 51
94	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 52
95	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 60
96	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 61
97	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 62
98	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 63
99	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 80
100	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 81
101	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 82
102	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 83
103	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 91
104	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 92
105	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 93
106	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	lcd.c	line 94
107	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	lcd.c	line 104
108	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	lcd.c	line 106
109	#1484-D (MISRA-C:2004 6.1/R) The plain char type shall be used only for the storage and use of character values	lcd.c	line 132
110	#1505-D (MISRA-C:2004 13.5/R) The three expressions of a for statement shall be concerned only with loop control	lcd.c	line 124
111	#1468-D (MISRA-C:2004 14.7/R) A function shall have a single point of exit at the end of the function	lcd.c	line 117
112	#188-D pointless comparison of unsigned integer with zero	lcd.c	line 151
113	#188-D pointless comparison of unsigned integer with zero	lcd.c	line 159
114	#188-D pointless comparison of unsigned integer with zero	lcd.c	line 167
115	#188-D pointless comparison of unsigned integer with zero	lcd.c	line 175
116	#1390-D (MISRA-C:2004 9.1/R) All automatic variables shall have been assigned a value before being used (variable "ADC_Data")	Potentiometer.c	line 37
117	#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void	Potentiometer.c	line 13
118	#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void	Potentiometer.c	line 33
119	#1435-D (MISRA-C:2004 20.1/R) Reserved identifiers, macros and functions in the standard library, shall not be defined, redefined or undefined ("NULL")	Potentiometer.c	line 11
120	#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void	Potentiometer.h	line 15
121	#1420-D (MISRA-C:2004 16.5/R) Functions with no parameters shall be declared and defined with the parameter list void	Potentiometer.h	line 16
122	#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 ("speed_limit")	SpeedLimit.c	line 158
123	#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions	SpeedLimit.c	line 108
124	#1405-D (MISRA-C:2004 12.5/R) The operands of a logical && or || shall be primary-expressions	SpeedLimit.c	line 113
125	#70-D integer conversion resulted in truncation	SpeedLimit.c	line 267
126	#70-D integer conversion resulted in truncation	SpeedLimit.c	line 264
127	#1468-D (MISRA-C:2004 14.7/R) A function shall have a single point of exit at the end of the function	SpeedLimit.c	line 78
128	#1467-D (MISRA-C:2004 14.6/R) For any iteration statement there shall be at most one break statement used for loop termination	SpeedLimit.c	line 67
129	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	spi.c	line 46
130	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	spi.c	line 49
131	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	spi.c	line 77
132	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	spi.c	line 80
133	#1406-D (MISRA-C:2004 12.7/R) Bitwise operators shall not be applied to operands whose underlying type is signed	spi.c	line 94
134	#1455-D (MISRA-C:2004 2.2/R) Source code shall only use /* ... */ style comments	spi.c	line 12
135	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 114
136	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 120
137	#552-D variable "received_data" was set but never used	spi.c	line 111
138	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 134
139	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 137
140	#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	spi.c	line 137
141	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 140
142	#1476-D (MISRA-C:2004 17.4/R) Array indexing shall be the only allowed form of pointer arithmetic	spi.c	line 144
143	#1376-D (MISRA-C:2004 1.1/R) Ensure strict ANSI C mode (-ps) is enabled	std_types.h	line 17