CSW 41 - Pedro Romano Splendore
Lab 3

Especificação da solução:

Nessa etapa foi pensado em como seria o fluxo e a implementação do projeto. A partir das duas imagens de teste, especificadas no arquivo main.cpp, com seus valores de altura e largura, conseguimos chamar a rotina em assembly, passando também o vetor que contaria os bits para criar o histograma.

Essa função em assembly tem o objetivo de, primeiro, realizar a checagem de se a imagem é maior ou menor do que 65.536 pixels. Caso a imagem seja maior, o programa retorna zero e finaliza, caso contrário, continua a execução. Após isso, por garantia, é chamada uma função que inicializa o vetor de contador de bits para o histograma, preenchendo-o com zero em todas as posições.

Após essas etapas podemos começar a preencher o vetor de contador de bits da imagem propriamente dito. Para isso, foi utilizado uma solução próxima à sugerida nos slides. Ou seja, realizamos a leitura do primeiro pixel e utilizamos esse valor para chegar na posição desejada do vetor e somar 1, ou seja, o valor do pixel como índice do vetor de contagem. Continuamos lendo todos os pixels da imagem, somente 1 leitura por pixel, até que o loop de iterações chegue no tamanho total da imagem. Com isso, retorna-se o número de pixels lidos e é exibido no terminal do IAR o vetor de contagem de bits de histograma.

Projeto (design) da solução:

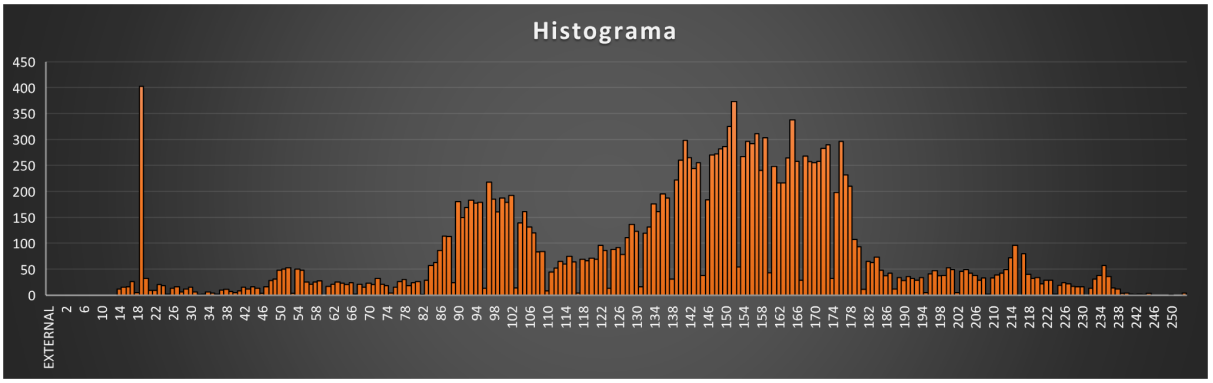
Foram utilizadas as estruturas de dados de uint16_t, sendo uma delas um vetor com o contador de bits e outra somente com 1 posição que recebe o valor total de pixels lidos.

O único valor passado como ponteiro para a rotina em assembly foi a imagem, visto que ela não seria modificada pela rotina em assembly, porém precisaria ser lida completamente. Os valores de altura e largura foram passados em forma de valor absoluto e o vetor de contagem de bits do histograma, se tratando de um vetor, o enviado para a rotina de assembly foi a posição inicial desse vetor.

A alocação dos registradores em assembly pode ser vista abaixo:

R0 - Width
R1 - Height
R2 - Ponteiro para a imagem
R3 - Contador de bits
R4 - Tamanho total da imagem calculado em assembly
R5 - ponteiro para guardar o início do contador de bits
R8 - valor atual das posições da imagem
R10 - registrador auxiliar para contagem
R11 - registrador auxiliar para contagem

O resultado obtido pela execução do programa pode ser visualizado abaixo:


Histograma
450
400
350
300
250
200
150
100
50
0
EXTERNAL
2
6
10
14
18
22
26
30
34
38
42
46
50
54
58
62
66
70
74
78
82
86
90
94
98
102
106
110
114
118
122
126
130
134
138
142
146
150
154
158
162
166
170
174
178
182
186
190
194
198
202
206
210
214
218
222
226
230
234
238
242
246
250