

2

① Todas as tarefas tempo-real têm restrições temporais mas estas podem não ser do tipo deadline. Ex. Janela, sinusóide, etc.

② Entre outros outros:

- código condicional
- `do while`
- DMA, PIPELINE, ...

③ O tick deverá ter um valor tão grande quanto possível para minimizar a carga de CPU. Todavia deve permitir expressar com rigor o período:  $\text{msec}(25,30,50) = 5$

④ Na condição indicada

$$U_{\text{LWS}}^{\text{na}} = n(2^{1/n} - 1) \leq 1 \quad \left. \begin{array}{l} \\ \\ \end{array} \right\} U_{\text{LWS}}^{\text{na}} \leq U_{\text{LWS}}^{\text{EOF}}$$

$$U_{\text{LWS}}^{\text{EOF}} = 1$$

Logo é verdade!

<p>⑤ <code>for (i=0; i&lt;12; i++) {</code>  <code>sum = sum + v[i];</code>  <code>}</code></p>	<p><code>for (i=0; i&lt;12; i+=4) {</code>  <code>sum = sum + v[i]</code>  <code>" = " v[i+1]</code>  <code>" = " v[i+2]</code>  <code>" = " v[i+3]</code>  <code>}</code></p>
---	--

ANTES      DEPOIS

↑

A técnica consiste em replicar as instruções no corpo de um ciclo `for` para a redução do overhead de controle. Usa mais memória, o q pode causar problemas / cache misses.