**Nome: André Carnevale RA: 13.03313-0**

**Provinha**

1. **O que é um microcontrolador?**

Circuito integrado que funciona como um mini computador, porém integrado em um único chip. Possui um núcleo que processa seus dados, memórias voláteis e não voláteis; e periféricos de entrada e saída (PIO’s), como I/O’s, geradores de sinais PWM, Timers, Conversores A/D, etc.

1. **Qual a família de microcontroladores utilizada no curso?**

A família de microcontroladores utilizada no curso é a família ARM Cortex M-4.

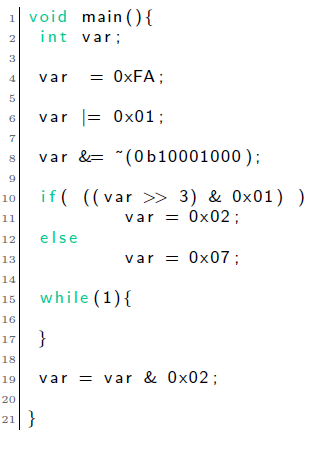
1. **Das propriedades básicas do microcontrolador, explique duas de sua escolha:**
2. **Harvard / Von Neumann:** dois tipos de arquiteturas de computadores, com a diferença de que a arquitetura Harvard baseia-se na utilização de memórias diferentes para controle e dados, enquanto a arquitetura Von Neumann utilizada a mesma memória para armazenar ambas.
3. **RISC/ CISC:** dois tipos de arquitetura de linguagem de processadores, que se diferenciam por o RISC (Reduced Instruction Set Computer) possuir uma linguagem de instruções mais reduzidas, porém mais rápidas de serem executadas, enquanto o CISC (Complex Instruction Set Computer) possui uma linguagem de instruções mais variadas e complexas, mas que demoram mais para serem executadas.
4. **Explique as seguintes definições do C: volatile/const/static.**

* Volatile: uma variável declarada como *volatile* pode ter seus dados alterados durante a execução do programa, porém sem o reconhecimento do compilador, impedindo-o de fazer otimizações no programa onde a variável encontra-se presente.
* Const: uma variável declarada como *const* pode apenas ter seus dados lidos, não podendo ser alterada. A variável *const* serve de proteção para que o valor armazenado na mesma não seja perdido no executar do programa.
* Static: uma variável declarada como s*tatic* pode ser global ou local. Variáveis globais *static* funcionam como variáveis globais comuns, com a diferença de que só funcionam no programa em que foram declaradas, já variáveis locais *static* mantém o seu valor em duas chamadas distintas da mesma função.

1. **Explique o uso do Watchdog Timer.**

O Watchdog Timer é um dispositivo de clock independente que, com a ausência de reset até um tempo programado, entende que houve uma condição de erro no programa principal, reiniciando o sistema.

1. **Dado um microcontrolador hipotético de 8bits, preencha o valor das variáveis conforme a evolução do código (sem considerar nenhuma otimização por nível do compilador). O subscrito no nome da variável indica a linha no código correspondente. Se o seu valor for indefinido, deixar o espaço em branco.**



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Var4 = | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| Var6 = | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| Var8 = | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| Var11 = | - | - | - | - | - | - | - | - |
| Var13 = | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Var19 = | - | - | - | - | - | - | - | - |