

Um grupo de alunos da CDC é apaixonado por códigos binários, e estão desenvolvendo seu próprio código secreto para compartilharem mensagens entre si, para isto estão usando 5 bits para representar cada letra e número. Os códigos estão fazendo sucesso entre os amigos, e eles decidiram que precisam de uma maneira para poderem ver quais bits representam cada caractere, já que cada um dos amigos possui sua própria ordem. Para isto, resolveram pedir sua ajuda e de seu grupo para descrever um hardware usando **Verilog** ou **SystemVerilog** que irá mostrar em um display de 7 segmentos (Figura 1) os **caracteres correspondentes à ordem em bits em que eles estão alocados**. Cada caractere será associado a um código binário de 5 bits, e deverá ser mostrado no display de acordo com os valores lógicos dos 5 bits de entrada.

Os displays de sete segmentos oferecem uma maneira de baixo custo como alternativa a displays mais complexos e caros. Seus segmentos podem ser usados e combinados para criar formas simplificadas de algarismos arábicos.

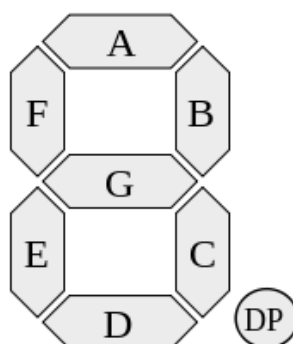


Figura 1: Display de sete segmentos.

No entanto, existe também um padrão de representação do alfabeto para tais displays, onde é possível de maneira simplificada representar cada uma das 26 letras, conforme ilustra a Figura 2.



Figura 2: Definição dos caracteres.

Para cada grupo será alocado uma ordem diferente de 20 caracteres aleatórios, incluindo números e letras, seguindo sempre a mesma ordem em bits, começando com 00000 e indo até 10011. Quando o módulo receber uma entrada inválida (>19 e < 32), deve-se apresentar todos os segmentos apagados.

Tabela 1 - Subconjunto de caracteres a ser implementado por cada grupo

Bits	000 00	00 01	00 01	000 11	00 10	001 01	001 10	00 111	01 0	010 01	01 0	01 011	011 00	01 10	01 11	011 11	10 00	10 00	100 10	10 011
1	L	V	Y	2	I	M	B	W	F	E	8	T	S	K	P	O	R	3	D	6
2	C	R	B	Q	N	4	M	3	A	5	S	7	9	1	O	2	Z	H	D	K
3	W	N	P	K	2	X	J	Z	S	R	L	A	1	Q	6	0	E	I	H	T
4	S	G	M	C	W	8	Z	B	F	P	5	X	K	9	6	V	N	Q	7	O
5	C	9	O	T	L	N	Q	1	K	W	4	Y	X	R	E	M	G	S	7	5
6	D	7	8	R	3	K	P	F	9	X	I	H	E	M	B	4	C	O	Z	6
7	3	U	4	R	T	Z	O	C	8	G	Q	2	P	N	1	A	L	6	9	F
8	2	B	K	U	3	N	I	7	9	S	D	P	Z	5	T	O	M	G	X	W
9	Y	O	M	E	V	A	H	9	N	K	W	6	0	2	P	3	5	U	X	T
10	O	W	0	N	X	8	G	I	J	E	K	M	F	H	T	U	4	P	C	2
11	0	3	H	9	V	J	P	2	E	K	6	O	M	5	U	L	N	F	4	7
12	L	6	V	P	J	R	7	2	U	B	Z	4	8	W	E	0	3	A	O	1
13	V	7	J	5	I	8	G	P	6	2	W	3	9	E	H	Y	S	K	M	L
14	7	U	0	4	9	Y	H	6	X	J	S	2	O	L	R	5	N	A	K	B

O que deve ser entregue:

- Uma documentação (em formato PDF) com uma descrição do que foi feito, incluindo tabelas verdade (se utilizadas), e partes importantes do código fonte.
- O código fonte em uma pasta compactada (.zip)
- Um vídeo de 2 a 4 minutos apresentando o trabalho funcionando para **(pelo menos)** os casos de testes apresentados.

Avaliação:

A distribuição de pontos no trabalho será feita da seguinte forma:

- 60% dos pontos: Código fonte.
- 20% dos pontos: Vídeo.
- 20% dos pontos: Documentação.

Observações:

- É permitido que grupos comuniquem entre si sobre como resolver o trabalho, mas é **PROIBIDA** a cópia de quaisquer partes de código fonte e/ou documentação. Cópias serão exemplarmente punidas para quem copiou e quem forneceu o trabalho.
- Como não temos acesso a FPGA, a saída poderá ser mostrada através de 7 saídas binárias, cada uma representando um segmento sendo: 0 para um segmento apagado ou 1 para um segmento aceso.