

FPGA x 摩斯电码

出题人: 颜嘉钦 易翔

【修改日志】

由于Minisys开发板与EG01的不同（Minisys并没有PS/2键盘接口），因此原文档“电码编码功能”评分标准中不合理的“外接键盘”附加分内容被修改成了“个性化设置蜂鸣器”。

此外，为了让大家更清楚地理解我们在文档中用到一些名词，我们在“电码解码功能”评分标准开头加了注释。内容如下：

“为了简化经典电报机中通过单个按键进行长按短按来编码的实现难度，我们采用【两个按键】分别进行“.”和“-”的输入（称为【短键】和【长键】），并有一个单独的【解码键】将当前已输入的摩斯电码解码成字符。”

小颜和小易是好朋友，常常在做完作业后聚在一起对答案。

但令人不愉快的是，每次他们对答案的时候，总有没写完作业的同学“隔墙有耳”，窃取劳动成果。

"我们还偏不让其他人听到。"

刚好最近隔壁的"数字逻辑"课给大家发了FPGA开发板。

于是小颜想了一个好主意：如果能用FPGA设计一个摩斯电码的编码解码器，把选择题和填空题的答案进行“加密”再和对方讨论，是不是就能够解决偷听的问题了呢？

"这样，应该就不会那么容易泄露'劳动成果'了。"

于是，小颜和小易拿出纸和笔写了一份设计文档，上面包括他们的需求、以及一些设计思路。然而，由于小颜和小易没有verilog开发环境，所以只能把接下来实现代码的工作交给正在上"数字逻辑"的你们了。

根据这个文档，你们能帮助小颜和小易实现这个基于FPGA的摩斯电码收发器吗？

Part 1 任务目标

小颜和小易需要你们实现一个摩斯电码收发器（下称“收发器”），该收发器具有两种模式，能分别支持摩斯电码的编码和解码。

收发器在**解码模式**中，能接受长短码的输入、能够在24个LED中显示输入的电码、以及能够在七段数码管中显示解码后的电码。

收发器在**编码模式**中，能接受小键盘输入，能够将输入的数字显示在七段数码管中（最多8字符）、能够将编码后的电码通过蜂鸣器按照一定规律播放出来。

Part 2 功能要求

基本功能 (20分)

- 蜂鸣器能够发出声音。
- 七段数码管能够显示内容。
- 能够切换编码/解码状态，并能看到对应状态提示。

电码解码功能 (30分+10附加分)

注：为了简化经典电报机中通过单个按键进行长按、短按来编码的实现难度，我们采用**两个按键**分别进行"."和"-"的输入（称为"**短键**"和"**长键**"），并有一个单独的**解码键**将当前已输入的摩斯电码解码成字符。

- 能够正确用长键、短键输入若干字符(例如"SOS")并成功显示在七段数码管上。
- 能够通过LED实时展示**长短码缓冲区**内容。
 - 有两种实现方式可供选择：实时显示；非实时显示，通过额外的开关刷新**长短码缓冲区**内容。
- 能够正确清空七段数码管和**长短码缓冲区**的内容。
- 对于无法编码任何字符的长短码，能够正常处理。
 - 例如：清空缓冲区，但七段数码管不新显示任何内容。
- **附加分内容**: 拥有退格(backspace)的功能，删除**长短码缓冲区**中输入最后一个"."或"-"。（多次按下可以多次删除，直到**长短码缓冲区**没有内容。）
- **附加分内容**: 拥有错误警告的功能，在出现长短码无法解码、输入字符已满(已有8位字符)等错误时，会产生**不同的**错误警告告诉我们错误类型。
 - 错误警告可以是七段数码管显示、LED灯显示、蜂鸣器特殊声音提示。

电码编码功能 (30分+10附加分)

- 能够正确使用小键盘输入若干数字(例如"2021")并显示在七段数码管上。
- 能够正确清空七段数码管的内容。
- 蜂鸣器能够正确播放出七段数码管字符所对应的电码。
 - 使用八个开关分别对应播放八个字符的电码，如果该位置上没有需要编码的字符，则蜂鸣器不输出。
- **附加分内容**: 拥有退格(backspace)的功能，删除七段数码管中最后输入的字符。（多次按下可以多次删除，直到七段数码管没有任何内容。）
- **附加分内容**: 允许使用一些开关个性化调节蜂鸣器的播放速度。例如，可以通过开关自行设置单个长码播放时间为1秒/1.5秒/2秒、单个短码时间为0.2秒/0.5秒/0.8秒、一个短间隔时间为0.5秒/1秒等。不同设置之间应易区分，即不同设置的播放速度在检查时应能被轻易的识别出来。
- **附加分内容**: 只通过一个开关控制蜂鸣器，一次性输出七段数码管中所有字符所对应的电码。

Part 3 各模块详解

为了帮助大家更好地了解**项目要求**、以及**需要做什么**，小颜和小易贴心地为大家准备了更详细的模块详解。

希望这能够为大家的分工及写代码提供一些帮助噢~

模块1 状态转换器

收发器具有两种状态，编码状态和解码状态。需要通过开关控制状态的切换。

由此，建议使用的控件以及功能如下：

状态转换开关

一个开关控制状态的切换。

状态显示灯

一个LED灯，用于显示当前状态。

通常来说，做状态转换器还兼顾着整合所有模块、让模块之间正常相互配合的任务。

模块2 解码模块

在解码状态下，解码模块会进行工作。解码模块应能接受长短码的输入、在24个LED中显示输入的电码、以及在七段数码管中显示解码后的电码。

由此，解码模块涉及的控件以及功能如下：

基本按键

包括两个按钮，分别叫做"短键"、"长键"。

按一次短键代表输入一个("·"), 按一次长键代表输入一个长码("-")。

注：通过按键输入的长短码，应被保存在一个临时存储长短码信息的地方，下文将其称为**长短码缓冲区**。

解码键

按下解码键代表长短码输入完成。此时需要将**长短码缓冲区**的信息通过摩斯电码表解码出对应的字符，并显示到七段数码管。

解码后应该清空**长短码缓冲区**。

如果输入的长短码无法解码出任何字符，则无需追加任何字符到七段数码管。（此时如果具有提示或报警信息就更好了~）

重置键

可以随时按下，清空当前的长短码输入以及七段数码管显示的字符。

LED灯

在解码状态下，实时显示**长短码缓冲区**的内容。

方案一：0表示短码，1表示长码，然后最右侧3位通过二进制数显示长短码的位数(长短码不超过7位)。

方案二：一个灯亮表示短码，两个相邻的灯亮表示长码，两个码之间空一个灯（灭）。例如，101101101表示电码 "· - - ·"。

七段数码管

显示解码出的最多8个字符。

模块3 编码模块

在编码状态下，编码模块会进行工作，由于小键盘的按键数量限制，采用小键盘编码则只要求编码数字。编码模块应能够接受小键盘输入，将输入的数字显示在七段数码显示管中、将编码后的电码通过蜂鸣器输出。

由此，编码模块涉及的控件以及功能如下：

小键盘输入

采用小键盘上的按钮，点击一个按钮就对应一个数字的输入。

键盘输入应该直接显示在七段数码管上。

编码键

按下时可以将七段数码管上显示的所有字符转换为对应摩斯电码。

转换的摩斯电码需要通过蜂鸣器播放出来。

重置键

可以随时按下，清空当前七段数码管显示的字符。

七段数码管

编码/解码状态下都能够显示最多8个字符。

蜂鸣器

在编码状态下，将字符依次编码为摩斯电码，然后通过蜂鸣器播放。

长鸣、短鸣分别代表“-”与“.”，短间隔代表一个字符尚未编码完成，而长间隔代表一个字符编码完成(准备进行下一个字符编码)。

长短鸣、长短间隔时间自己设计，能区分即可。但如果你们在时间问题上产生了纠结，那么小颜建议：设计长鸣为1秒、短鸣为0.5秒，长间隔为2秒、短间隔为0.5秒，

Part 4 一些提示

1. 建议采用的七段数码管显示字符标准：

A	b	C	d	E	F	G	H	I	J	K	L	
A	b	c	d	E	F	G	H	I	J	K	L	
M	n	o	P	q	r	S	t	U	v	W	X	
M	n	o	P	q	r	S	t	U	v	W	X	
Y	Z		1	2	3	4	5	6	7	8	9	0
Y	Z		1	2	3	4	5	6	7	8	9	0

2. 编码/解码采用的摩斯电码表：

MORSE CODE LETTERS AND NUMBERS

A · —	K — · —	U · · —	1 · — — — —
B — · · ·	L · — · ·	V · · · —	2 · · — — —
C — · — ·	M — —	W · — —	3 · · · — —
D — · ·	N — ·	X — · · —	4 · · · · —
E ·	O — — —	Y — · — —	5 · · · · ·
F · · — ·	P · — — ·	Z — — · ·	6 — · · · ·
G — — ·	Q — — · —		7 — — · · ·
H · · · ·	R · — ·		8 — — — · ·
I · ·	S · · ·		9 — — — — ·
J · — — —	T —		0 — — — — —

3. 注意：编码状态和解码状态应该完全独立，即在任何情况下都不允许混合使用小键盘和长短键解码输入字符（这样的设计容易出现multi-driver错误，在verilog中是无法进行逻辑综合的）。
4. 建议使用按键(上升沿)来作为按键的输入设备噢~