



选题名称：基于 verilog 的自动售货机

院（系）：	吴健雄学院
小组成员：	李勃璘（61522529）
	冯光宇（61522527）
指导教师：	翟建锋
评定成绩：	

## 1 课题背景

自动售货机是一种无需人工服务的设备，消费者可以通过它自动购买各种商品。通常，自动售货机内部装有商品（如饮料、零食、日常用品等），通过投币、纸币、刷卡、移动支付等方式支付后，机器会自动将选择的商品提供给消费者。

## 2 FPGA 与 Vivado 简介

### 2.1 FPGA 简介

现场可编程逻辑器件 (FPGA, Field Programmable Gate Array) 是一种集成电路可编程逻辑器件，它具有高灵活性、高可靠性、低功耗、高速度等特点，适合作为数字系统硬件设计的核心部件。利用 FPGA，可以实现复杂的功能，如图像处理、视频处理、音频处理、机器人控制等，已经成为电子信息领域仿真、测试、验证的重要工具。

本设计中，我们将采用 NEXY4 4 DDR 型号的主板作为主控芯片。它基于 Artix-7 FPGA 系列，具有多种接口，如 VGA、USB、Ethernet 接口，同时允许 16 位拨片输入信号，5 个按钮，以及两个复位键，支持用户自定义，能够满足自动售货机的设计需求。

### 2.2 Vivado 简介

Vivado 是一款基于 Xilinx FPGA 的集成开发环境，它是一款功能强大、界面简洁、操作方便的集成开发环境。本实验将使用 Vivado 2017.4 版本进行设计，并作为仿真及测试软件进行自动售货机的设计。

## 3 自动售货机功能介绍

### 3.1 功能概述

该项目的目标是设计一款基于 FPGA 的自动售货机，该自动售货机具有以下功能：

1. 商品选择：用户可以通过按下按钮选择商品，并将其放入“购物车”。允许用户每次最多选择两件商品（数量不限）。
2. 结账：用户可以在购买的过程中随时结算“购物车”中的商品，此时系统进入结账环节。支持用户使用 1 元、5 元、10 元、20 元、50 元进行支付，并支持溢出找零。

3. 找零：用户在结账过程中，若购物金额超过支付金额，系统将自动给用户找零。规定用户每次找零不超过 1 元，直至找零完毕。
4. 退出：用户可以在任意时刻退出系统，系统将自动清空“购物车”中的商品并退回至空闲状态。

### 3.2 系统流程图

为了让自动售货机的工作流程更加清晰，便于用户理解该系统的工作原理，以及理清各个状态之间的关系，我们设计了下面的系统流程图，便于后续参考：

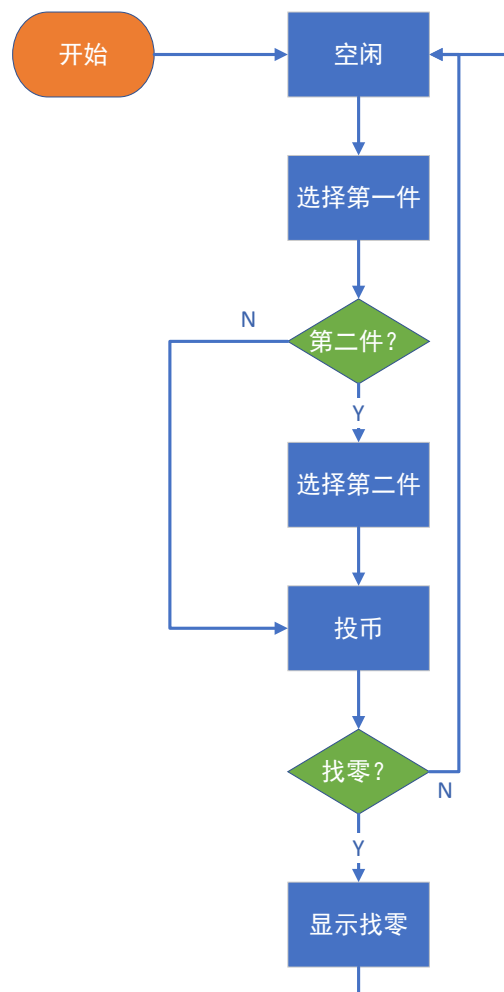


图 1: 自动售货机系统架构

## 4 系统设计

### 4.1 代码文件列表

- `state_transition.v`: 状态机及状态机之间的转换模块;
- `display_design.v`: 显示模块;
- `top_layer.v`: 顶层模块;
- `key_filter.v`: 按键消抖部分;
- 其他文件: 测试文件、测试脚本等。

### 4.2 状态机设计

为了实现自动售货机的功能,我们使用了两层状态机进行实现。在 Verilog 设计中,可以通过 `case-when` 语句或者 `if-else` 语句实现状态机的功能。两层状态机的各层功能如下:

1. **第一层状态机**: 完成六个状态的切换,包括空闲、商品选择、结账、找零、退出、空闲状态。
2. **第二层状态机**: 分多个 `always` 块,分别完成商品选择、结账、找零、退出状态的具体功能(空闲状态的功能由第一层状态机完成)。

下面将具体描述该状态机实现的功能和转换条件。

#### 4.2.1 状态机功能描述

##### 空闲状态 (IDLE):

初始状态。允许用户按下“确认”键进入商品选择状态。投入钱币数被初始化为 0。

##### 商品选择状态 (GOODS\_one):

用户选择第一件商品,输入购买的商品编号和数量,系统将基于此计算总钱数。用户可以继续按下“选择”键选择第二件商品,或按下“确认”键进入结账状态。

空闲状态下,系统处于待机状态,等待用户的操作。在该状态下,用户一旦按下“确认”键,系统将进入商品选择状态。

##### 商品选择状态 (GOODS\_two):

用户选择第二件商品,输入购买的商品编号和数量,系统将基于此计算总钱数。用户可以继续按下“取消”键退回第一件商品选择,或按下“确认”键进入结账状态。

### 结账状态 (PAYMENT):

用户确认购买的商品，系统将根据用户的支付金额，结算购物车中的商品。用户通过按键输入钱数，系统将计算总钱数（并显示），按下“确认”键将进入找零，按下取消键则进入“待定”状态，决定继续选择还是找零退出系统。

### 找零状态 (CHANGE):

用户支付的金额超过购买的金额，系统将自动给用户找零。规定用户每次找零不超过 1 元（按 CHANGE 键），直至找零完毕，系统退回到初始状态 (IDLE)。

### 待定状态 (TEMP):

该状态用于决定是否继续选择商品，或是进行找零，退出所有在结账时输入的钱数后，系统回到初始状态。按下“找零”键将进入找零状态，按下“确认”键将回到商品 1 选择。

## 4.2.2 状态机功能验证

使用 vscode 中的 TerosHDL 插件内置的状态机分析模块，我们可以很方便地生成 Verilog 代码对应状态机的状态转移图。经过比较，和理论设计的状态转移图进行了对比，发现两者转移图的状态数量、状态转移方向、状态转移条件等都一致。

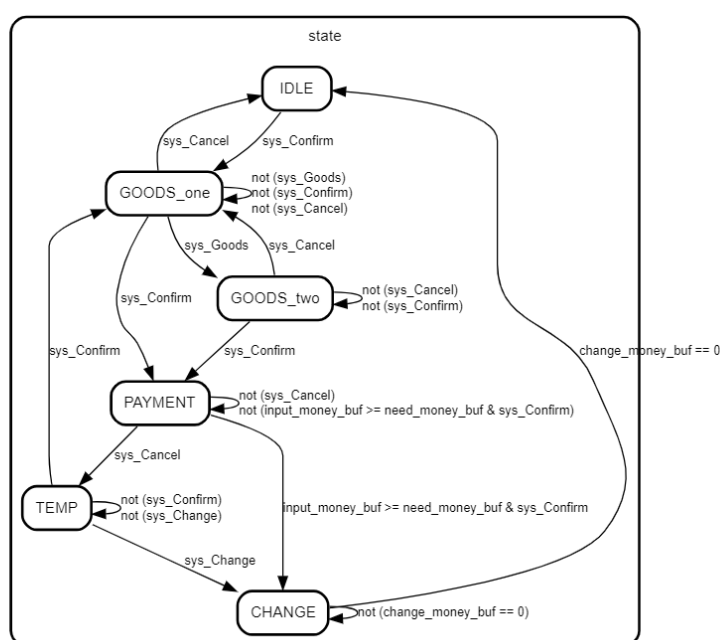


图 2: 状态转移图

### 4.3 按键消抖设计

### 4.4 显示模块设计

### 4.5 I/O 接口设计

为了便于自动售货机的硬件实现，需要设计可以映射到 FPGA 主板上的 I/O 接口。我们使用了如下的接口：

- 五个按键：分别为“确认”、“取消”、“复位”、“找零”、“商品选择”；
- 五个拨片输入：分别为“1 元”、“5 元”、“10 元”、“20 元”、“50 元”；
- 八个拨片输入：对应商品编号的六位二进制数，以及购买数量。

## 5 仿真与实物验证

### 5.1 仿真

我们使用 Vivado 的仿真功能对自动售货机进行了仿真，并通过测试验证了其功能。

值得一提的是，在仿真时，我们尝试使用 Vscode 中的 TerosHDL 插件进行仿真。TerosHDL 支持对项目进行仿真，并可以生成波形图。但由于其需要安装的附加组件较多，我们最终放弃了这个想法。

### 5.2 实物验证

将程序烧录至 FPGA 主板上，并连接上电源，测试其功能。

## 6 总结

## 附录 A 项目发布

该项目的工程文件及本报告已经上传到 Github 上，项目地址为：<https://github.com/LiPtP0000/Micro-Vending-Machine>，包含了本项目的所有代码以及仿真测试文件。

## 附录 B 项目代码列表