Simple GPIO Specification

Introduction

The Simple General Purpose IO (GPIO) core is a basic 8-bit GPIO module designed for interfacing with external devices through general-purpose input/output pins. It adheres to a simple WISHBONE bus interface for communication, allowing for easy integration into larger systems. The module supports both input and output modes for each GPIO pin, configurable through a control register. This design is suitable for applications requiring basic GPIO functionality with minimal complexity.

Architecture

The Simple GPIO module consists of a single top-level module named `simple_gpio`. The architecture is centered around two primary registers: the Control Register and the Line Register. The Control Register determines the mode (input or output) of each GPIO pin, while the Line Register is used to read the status of input pins or set the output level of output pins.

Key Operations

- **Control Register (0x00):** Configures each GPIO pin as input (‘0’) or output (‘1’).
- **Line Register (0x01):**
  - In input mode, it reads the current level of the GPIO pins.
  - In output mode, it sets the output level of the GPIO pins.

Implementation Details

- The module supports up to 8 GPIO pins, configurable via the `io` parameter.
- The design includes mechanisms to reduce metastability by double-latching the GPIO inputs.
- The WISHBONE interface is used for communication, with signals for clock, reset, cycle, strobe, address, write enable, data input, and data output.

Interface

Signal	Width	In/Out	Description
clk_i	1	In	Clock input signal.
rst_i	1	In	Asynchronous active-low reset signal.
cyc_i	1	In	WISHBONE cycle signal.
stb_i	1	In	WISHBONE strobe signal.
adr_i	1	In	Address input signal (1 bit for selecting between Control and Line Register).
we_i	1	In	Write enable signal for WISHBONE interface.
dat_i	8	In	Data input for WISHBONE interface.
dat_o	8	Out	Data output for WISHBONE interface.
ack_o	1	Out	Acknowledge signal for WISHBONE interface.
gpio	io	In/Out	General-purpose input/output pins.

Timing

- **Latency:** The module operates synchronously with the clock signal `clk_i`. The output

data `dat_o` and acknowledge signal `ack_o` are updated on the rising edge of the clock.
- **Signal Behavior:**
  - The GPIO inputs are latched twice to mitigate metastability issues.
  - The acknowledge signal `ack_o` is asserted one clock cycle after a valid WISHBONE access is detected.

Usage

To use the Simple GPIO module: 1. **Configure GPIO Pins:** - Set the desired mode (input or output) for each GPIO pin by writing to the Control Register (address 0x00). - For input mode, set the corresponding bit to '0'. - For output mode, set the corresponding bit to '1'.

2. **Read/Write GPIO Pins:**
   - In input mode, read the current level of the GPIO pins from the Line Register (address 0x01).
   - In output mode, write the desired output level to the Line Register.
3. **Integration with WISHBONE:**
   - Ensure proper WISHBONE cycle (`cyc_i`) and strobe (`stb_i`) signals are asserted for valid transactions.
   - Use the `ack_o` signal to confirm the completion of a transaction.

This module provides a straightforward interface for controlling GPIO pins, suitable for various digital interfacing tasks in embedded systems.

Functional Description (Generated by funcgen)

Verilog Design Modules Functional Description

Module: simple_gpio (File: simple_gpio.v)

Purpose

The `simple_gpio` module is a simple 8-bit General Purpose Input/Output (GPIO) core that interfaces with a WISHBONE bus. The module allows each GPIO pin to be configured as either an input or an output. GPIO pins can read input signals from external devices or drive output signals based on the configuration.

Parameters

- **io**: Default = 8. This parameter specifies the number of GPIO pins supported by the module. The module supports up to a maximum of 8 GPIO pins.

Ports

- **clk_i** (input, 1-bit): The clock signal driving the sequential logic of the module.
- **rst_i** (input, 1-bit): Asynchronous active-low reset signal that initializes the module's state.
- **cyc_i** (input, 1-bit): Indicates a valid bus cycle in conjunction with `stb_i`.
- **stb_i** (input, 1-bit): Strobe signal indicating a valid data transfer cycle.
- **adr_i** (input, 1-bit): Address line to select between the Control Register (`adr_i = 0`) and the Line Register (`adr_i = 1`).
- **we_i** (input, 1-bit): Write enable signal for the WISHBONE interface.
- **dat_i** (input, 8-bit): Data bus used for writing data into the module.
- **dat_o** (output, 8-bit): Data bus used for reading data from the module.

- **ack_o** (output, 1-bit): Acknowledgment signal indicating the completion of a bus transaction.
- **gpio** (inout, io-bit): Bidirectional GPIO pins interface for external signal interaction.

Internal Signals

- **ctrl** (reg, io-bit): Control register that determines the mode (input/output) of each GPIO pin.
- **line** (reg, io-bit): Line register utilized to read current pin states or set output levels.
- **lgpio** (reg, io-bit): Latches the states of `gpio` input pins on the clock edge.
- **llgpio** (reg, io-bit): Further latches `lgpio` to reduce the risk of metastability.
- **igpio** (reg, io-bit): Temporary internal signal used to assign GPIO outputs.
- **dat_o** (reg, 8-bit): Register holding the current output data on the `dat_o` port.
- **ack_o** (reg, 1-bit): Register managing the acknowledgment signal logic.

Functionality

Sequential Logic

1. **Reset Logic**: On a low edge of `rst_i`, the `ctrl` and `line` registers are reset to zero. Similarly, `ack_o` is reset to zero.
2. **Control and Line Register Update**: If a valid write (`wb_wr`) is detected:
   - If `adr_i` is high, the `line` register is updated with `dat_i` to set output levels.
   - If `adr_i` is low, the `ctrl` register is updated with `dat_i` to configure pin modes.
3. **GPIO Latching**: The `lgpio` and `llgpio` registers are sequentially updated with the current input levels on the `gpio` pins to manage metastability.

Combinational Logic

- **Data Output Muxing**: The `dat_o` output data is multiplexed between the `ctrl` register and the `llgpio` level according to the `adr_i` address signal.
- **GPIO Output Assignment**: The GPIO's output behavior is controlled by checking `ctrl` bits. If the bit is set (`'1'`), the corresponding `line` bit is driven on the `gpio`; otherwise, the pin is left in high impedance (`'z'`).

Control Logic

- **Acknowledgment Signal**: The `ack_o` signal is used to confirm the termination of bus transactions. It remains high for one cycle after detecting a WISHBONE access.

Instantiations

There are no sub-module instantiations in the `simple_gpio` module.

Inter-Module Connections

Since `simple_gpio` is a standalone module in this design, there are no hierarchical interconnections to describe. The module interfaces directly with a WISHBONE bus and external GPIO signals through its ports.

This document provides a detailed examination of the `simple_gpio` Verilog module, offering insights into its design and functionality, which can guide hardware developers in further implementing or verifying similar GPIO interfaces.