Pattern Matcher

Design Specification

June 30th, 2017

Table of Contents

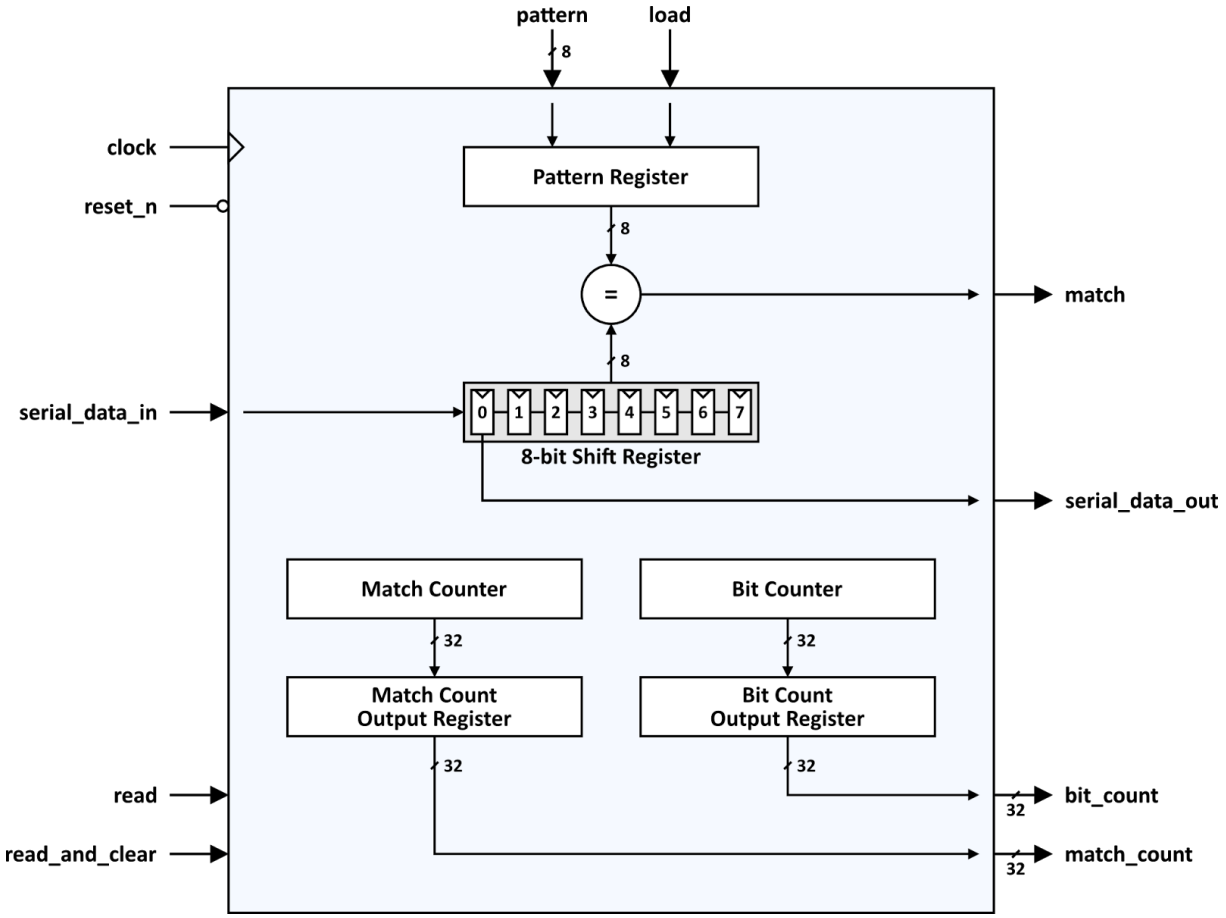
1. Introduction

The pattern matcher is a module that samples a 1-bit input signal at each clock cycle and searches for a specific pattern / sequence of bits. When the pattern is found, a one clock pulse is generated at the output.

2. Block Diagram

Below is a simplified schematic of the block diagram, presenting the main functional blocks inside the module.

Figure 2.1 - Pattern matcher block diagram



3. Operation

After a reset, all the outputs and the internal registers will be equal to 0 and the module will start in idle mode. It remains so until the first pattern is loaded. During this period, no matches are generated. By asserting the `load` signal, the value of the `pattern` input will be registered internally and used for the matching operation.

The serial data input is sampled at every clock cycle and shifted into the shift register. Bit 0 of the shift register is connected to `serial_data_out`. When the value stored in the internal pattern register matches the value of the shift register, the `match` output will be asserted on the following clock cycle and the internal match counter will be incremented.

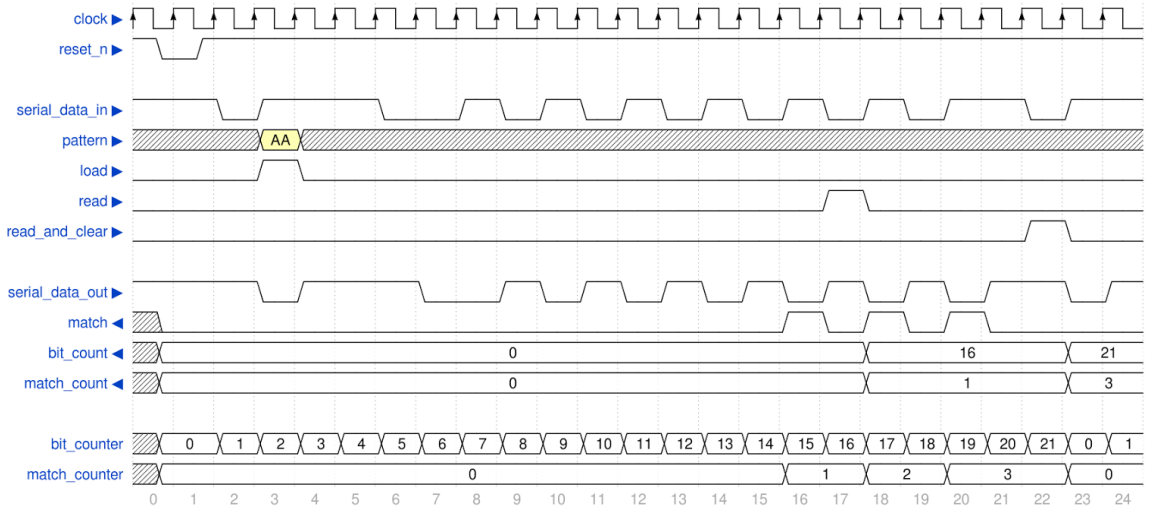
The module has an internal bit counter, which counts the number of bits shifted out of the module, starting from the first clock cycle after reset. The bit counter can't be directly accessed. Instead, by asserting `read` or `read_and_clear`, its value can be saved in the bit count register, which is directly connected to the `bit_count` output. In addition to this, if `read_and_clear` is asserted, the counter start counting from 0 beginning with the following clock cycle.

Similarly, the match counter counts the number of successful matches. Its value can be accessed by asserting `read` or `read_and_clear` and reading the value from the match_count output. Asserting `read_and_clear` will cause the match counter to start counting from 0 beginning with the following clock cycle.

4. Waveforms

In the waveforms below, the signals marked with "▸" are the module's inputs, while the signals marked with "◂" are the module's outputs.

FIgure 4.1 - Pattern matcher timing diagram



5. Port List

Table 5.1 - Port list

Name	Width	Direction	Description
clock	1	Input	System clock.
reset_n	1	Input	Active low asynchronous reset.
serial_data_in	1	Input	Serial data input.
pattern	8	Input	The pattern to look for in the serial input signal.
load	1	Input	When asserted, the pattern at the input is stored in the internal pattern register.
read	1	Input	When asserted, the module stores the current values of the bit and match counters in the output registers.
read_and_clear	1	Input	When asserted, the module stores the current values of the bit and match counters in the output registers, and clears the counters.
serial_data_out	1	Output	Serial data output.
match	1	Output	Indicates if a pattern has been found in the input signal.
bit_count	32	Output	Indicates the total number of bits shifted through the module, at the moment when read or read_and_clear was last asserted.
match_count	32	Output	Indicates the total number of matches, at the moment when read or read_and_clear was last asserted.