MEMO NUMBER: GFAU_SRS_01
DATE: October 18, 2017
TO: EFC LaBerge
FROM: Sabbir Ahmed, Jeffrey Osazuwa, Howard To, Brian Weber
SUBJECT: System Requirement Specifications Draft

1 Introduction

A Galois field is a field with a finite number of elements. The nomenclature $GF(q)$ is used to indicate a Galois field with q elements. For $GF(q)$ in general, q must be a power of a prime. For each prime power, there exists exactly one finite field. The best known and most used Galois field is $GF(2)$, the binary field.

The Galois Field Arithmetic Unit handles irreducible polynomials in $GF(2^n)$, where $\{2 \leq n \leq 16\}$. The ALU generates all the terms in the field of the polynomial, and allows the user to view and apply the following binary operations:

- Addition
- Subtraction
- Multiplication
- Division
- Logarithm

1.1 Document Overview

This document serves as the System Requirements Specification for the Galois Field Arithmetic Unit. The description and requirements of the project are embodied in this document.

The Specification is divided into separate segments pertaining to individual components and requirements on different levels. Figures and tables are attached where necessary to assist in demonstrating concepts.

1.2 Mission Scenario

Cryptography has many expensive calculations that are difficult for low power and inexpensive microcontrollers to handle. Galois fields are frequently used in the field of cryptography. The GFAU will make Galois field calculations more accessible to such low powered devices.

1.3 System Overview

The GFAU system is composed of discrete modules residing in a single programmable board. Individual modules shall be programmed to solely complete an assigned task. Although modules are assigned individual tasks, they shall not have exclusive components for its functionality.

1.4 System Boundary Diagram

The system boundary diagram of the GFAU has been provided in Figure 1 below.

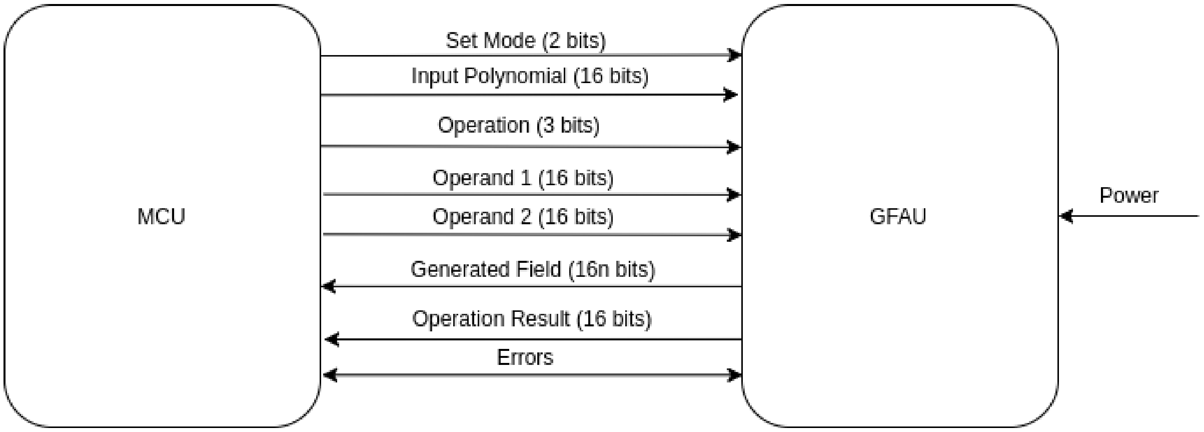

Figure 1: System Boundary Diagram of the Galois Field Arithmetic Unit , where n is the Number of Terms

The user input and output (I/O) interface are handled by the microcontroller (MCU) which are transferred via busses. The user inputs consist of the mode bit, the input generating polynomial and the binary operation(s) along with their corresponding operands. The MCU transfers the data to the unit to perform the desired operations. The GFAU will return the outputs and any errors detected back to the MCU.

1.5 Functional Flow Diagram

The functional flow diagram of the GFAU has been provided in Figure 2.

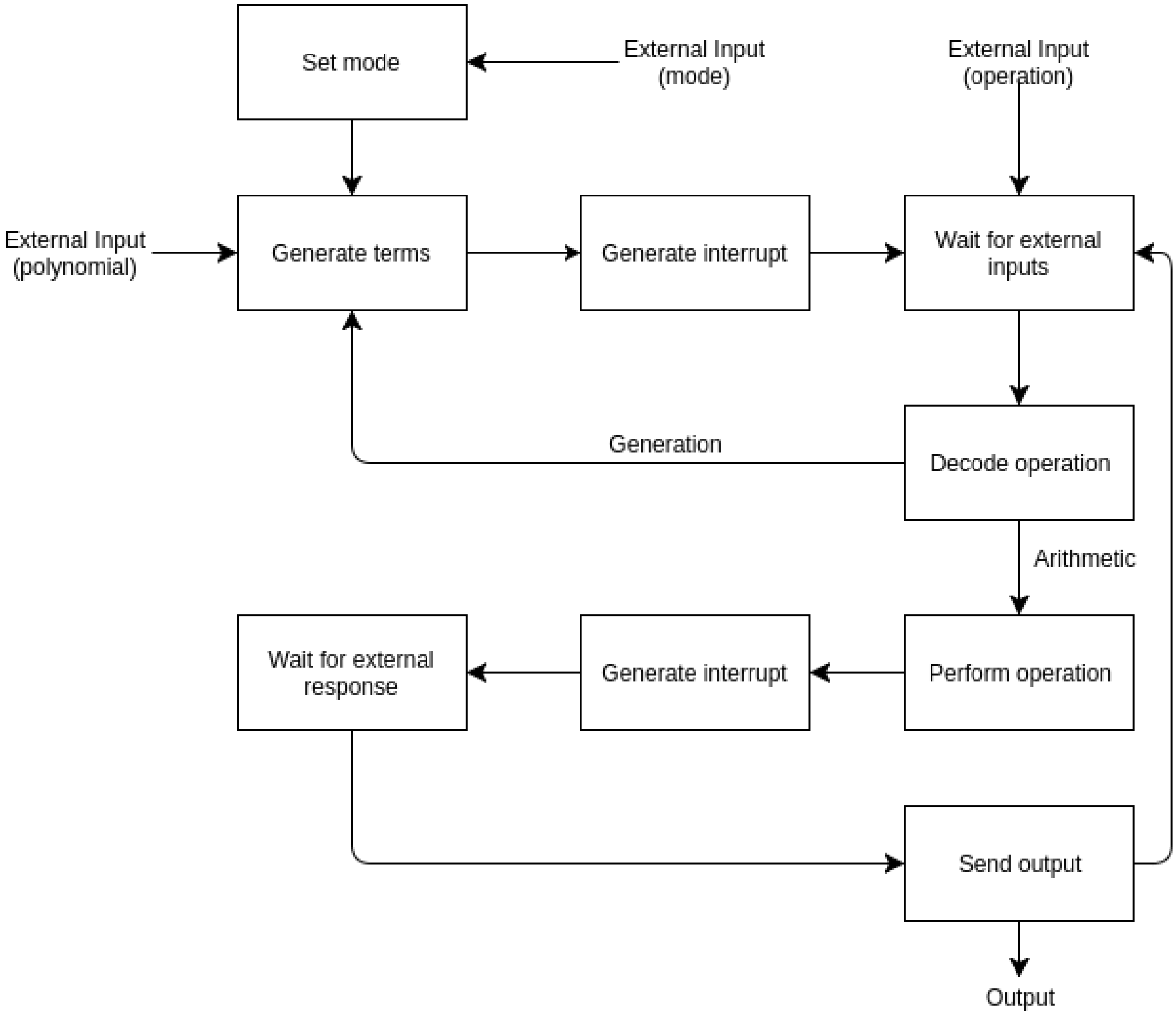

Figure 2: Functional Flow Diagram of the Galois Field Arithmetic Unit

The diagram provides a high-level overview of the sequence of processes that take place in the unit. In total, the unit waits for an input from the user in three separate instances. The order of the inputs are essential for the unit to proceed as desired. The **mode** input sets the width of the data bus in the GFAU. The **polynomial** input consist of the generating polynomial to generate its terms in the Galois field. The **operation** input consists of the two operands along with the desired binary operation.

1.6 Data Flow Diagram

The data flow diagram of the GFAU has been provided in Figure 3.

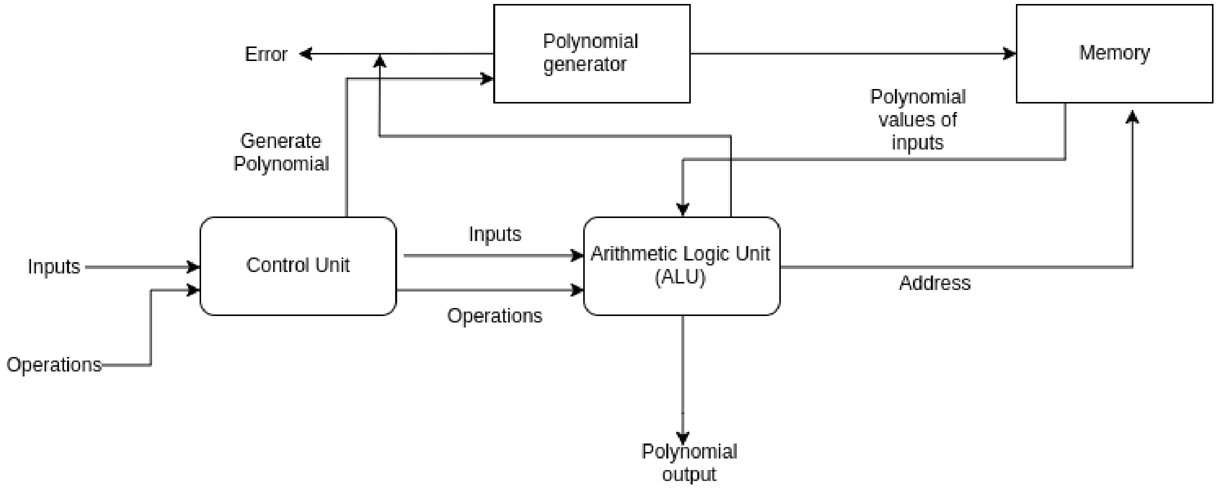

Figure 3: Data Flow Diagram of the Galois Field Arithmetic Unit

The diagram provides a lower-level view of the system emphasizing the individual components and their role in converting the input data to the desired output. The **Error** lines are used by multiple components to send interrupt signals to the user when required.

2 Requirements

2.1 System Requirements

2.2 Hardware Requirements

Hardware portability shall be prioritized on the GFAU. The hardware portability provides flexibility in the ranges of its specifications. The ranges of these specifications shall be reasonably bounded.

1. The unit shall be functional at a variety of clock speeds at a minimum range of (4 MHz-100 MHz).
2. The GFAU shall perform its computations in () instructions per clock
3. ...

2.3 Software and Testing Requirements

Extensive simulations shall be conducted during the development of the design in a hardware description language (HDL). This process makes debugging easier and minimizes the risk of unintended behavior in the prototype. This section outlines the HDL code requirements and tests that the simulations are required to pass.

1. Extensive design verification simulations shall be done before purchasing any hardware
2. All HDL code shall be synthesizable
3. Gate delay and other relevant values shall be parametrized to easily match the specifications of candidate hardware during verification

2.4 Communication Requirements

In order for the GFAU to be able to communicated with a wide variety of external devices, the GFAU shall implement communication methods which are commonly used. The details of how the GFAU shall communicate are outlined in this section.

1. The GFAU shall have an option to use either a 8, 16, (or 32) bit data-bus.
2. After any given operation is complete and the result is placed on the bus, the GFAU shall set a ready pin.
3. The external device may monitor the ready pin using polling or interrupts to know when to pull the data from the bus.
4. The GFAU shall use a common or easy to implement protocol for pushing blocks of data over a bus that is too small to send on one clock.

2.4.1 Errors

This section outlines the cases in which errors are thrown.

1. The unit shall produce an error if an operand is not in the same Galois field.
2. The unit shall produce an error if the FPGA does not receive all the inputs from the microcontroller.
3. The microcontroller shall produce an error if it does not receive all the outputs from the FPGA.