**Wireless AVR Chip Programmer**

**Final Project Report**

Abhijeet Dutt Srivastava

Atharva Nandanwar

ECEN 5613 Embedded System Design

December 12, 2019

1. Introduction
   1. System Overview
2. Technical Description
   1. Board Design Considerations
      1. Power Supply
      2. Oscillator Circuit
      3. Reset Circuit
      4. General Things for convenience
      5. MSP432P401R Development Board
   2. Firmware Elements
      1. UART Driver
      2. Bluetooth Setup Driver
      3. CRC Driver
      4. SPI Driver
      5. ISP Command Library
      6. Target Board Software
   3. Software Elements
      1. Hex Parser
      2. State Machine Design
3. Results and Error Analysis
   1. SPI Topologies
   2. Setting up Bluetooth Modules
   3. Error condition after second reception
4. Conclusion
5. Future Development Ideas
6. Acknowledgements
7. References
8. Appendices
9. Introduction

This year we had an opportunity to learn Embedded System Design, and I must say that we have learnt quite enough. This project is demonstration of what we have learnt, and how we went about applying this learning. This project is Wireless AVR Chip Programmer. Why a wireless device? It comes from Abhijeet’s love for wireless devices. Why a programmer? It comes from Atharva’s fondness for exploring the field of programmers/debuggers.

This project incorporates software that we wrote ourselves from scratch. It involves design considerations while designing a generic Embedded System. It involves our dedication to make a system that is stand-alone and can manage errors.

* 1. System Overview

A close up of text on a white background

Description automatically generated

Figure .1 System Overview

Figure 1.1 shows overall system overview. We have used MSP432P401R Development Board as a programmer for our AVR Embedded System. AVR Embedded System comprises ATmega328p based system, along with elements required to help it function – power circuitry, oscillator circuit, reset circuit. We are using power circuit on our AVR Embedded System to power the Programmer Board. Transmitter will be powered from the USB connection on the Computer, and will receive data from Computer over UART Protocol. The HC-05s communicate over Bluetooth, and transfer program data and control data.

* In general, we talk about basic architecture of our system.
* Talk about AVR Emb system
* Talk about MSP432 board – Programmer
* Talk about HC-05s
* Talk about transmitter
* Describe how each element has been added in the system

1. Technical Description
   1. Board Design Considerations

Put a diagram.

Talk about what board we are talking about – Target Board or AVR Embedded System Board

Give overview of Connectors, what considerations we put when we added the connectors

Give detailed analysis regarding the flowing current and how decoupling capacitors will interact with the flow of current

* + 1. Power Supply

We are using 9V Power Supply unit to get the power. This provides 9V DC. We have a bridge rectifier for reverse polarity protection. Output of the bridge rectifier is 7.3V, because of voltage drop across the diodes. This voltage is brought down to 5V by 7805. We have added adequate capacitors on Input and Output rails to ensure noise suppression. This 5V rail is available for other components to use. (Talk about current consumption through this 5V regulator, give details of thermal calculations.) One of the consumers of this 5V is MSP432P401R Development Board which acts as programmer. Other consumer is MCP1857 which is used to generate 3.3V Power Rail. 3.3V Power rail is used to power the AVR Embedded System.

* + Talk about why we choose MCP1857, talk about why we choose 7805, choices of decouping capacitors roughly
  + Add graph for power on, power rail voltages for 5V and 3.3V rails.
    1. Oscillator Circuit

We have used 8 MHz oscillator for the circuit.

Attach the screenshot of application note talking about the values of capacitor to be used.

Put a photo of actual oscillator on the board, and different considerations we had to take care of, and the possible errors it produces

Put a oscilloscope screenshot of oscillator rise up time. Talk about the fuse setting we changed to get the clock from oscillator, and the delay timing – which is perfectly adequate.

Talk about why 8MHz crystal and not 16 MHz, show the curve about stable frequency operation – check if it’s alright first

* + 1. Reset Circuit
    2. General Things for convenience
    3. MSP432P401R Development Board
  1. Firmware Elements
     1. UART Driver
     2. Bluetooth Setup Driver
     3. CRC Driver
     4. SPI Driver
     5. ISP Command Library
     6. Target Board Software
  2. Software Elements
     1. Hex Parser
     2. State Machine Design

1. Results and Error Analysis