Figure 1: round key algorithm

Figure 2: Algorithm Rounds

1 Introduction

The following paper seeks to investigate Cryptography issues in within the domain of embedded systems. In specific this report will conduct three individual performance investigations of the AES-128 encryption environment on a ATmega16 micro controller. The dimensions of this analysis will be, RAM consumption, Code Size, and finally Execution Time, all of which will be implemented using the low level programming language C, and analysed using AVR Studio 6.
Knowledge about AES and Cryptography in general is assumed, as well as basic programming skills in C. However, this article will provide a short introduction towards AES-128 in terms of algorithmic specifications.

1.1 AES-128 - Algorithm Summary

As mentioned the AES cryptography system that this article will focun on is the AES-128 environment with 10 rounds. The operations of AES and their order is given in table 1.

In short terms, the `AddRoundKey` operation simply XOR's the message with the key from the key schedule.
The Substitution Bytes operation simply substitutes the message value with the value of the SBox table. That is the value of the message byte becomes the index.
Shiftrows operates by shifting every row x positions to the left, where x is the row number. Eg: row 0 stays the same, row 1 is shifted 1 position to the left, and so on.

AES Encryption over n rounds	
AddRoundKey	Pre-whitening
SubBytes ShiftRows MixColumns AddRoundKey	For n-1 rounds
SubBytes ShiftRows AddRoundKey	The Final Round

Table 1: The general AES algorithm structure

1.2 Analysis parameters explained