PWM Module Requirements

1. Introduction

The purpose of this document is to define the requirements for implementing a Pulse Width Modulation (PWM) module on the Atmega32 target. The module should contain the following options as pre-compiler configuration:

- PWM modulation mode (Fast PWM or Contract Phase Mode)
- PWM polarity (inverting or non-inverting)

In addition, the module shall contain the following APIs:

- void PWM_Init();
- void PWM_Generate_CHANNELA(uint8 Copy_u8DutyCycle,uint32 Copy_u32freq);
- void PWM_Generate_CHANNELB(uint8 Copy_u8DutyCycle,uint32 Copy_u32freq);

2. PWM Configuration Options

The following options should be configurable using pre-compiler directives:

2.1 PWM Modulation Mode

The PWM module should support two modes of operation:

Fast PWM mode: In this mode, the PWM frequency is fixed and the duty cycle can be varied.

Contrast Phase Mode: In this mode, the PWM frequency and duty cycle can be varied independently.

2.2 PWM Polarity

The PWM module should support two polarities:

- Inverting: In this mode, the PWM output is inverted.
- Non-inverting: In this mode, the PWM output is not inverted.

3. PWM APIs

The following APIs should be provided by the PWM module:

3.1 void PWM_Init()

This API should initialise the PWM module based on the #define for the PWM Mode and Polarity for the Mode.

3.2 void PWM_Generate_CHANNELA(uint8 Copy_u8DutyCycle,uint32 Copy_u32freq)

This API should generate a PWM signal on Channel A of the Atmega32 target with the following parameters:

- Duty cycle: Copy_u8DutyCycle (0 to 255)
- Frequency: Copy_u32freq in Hz

3.3 void PWM_Generate_CHANNELB(uint8 Copy_u8DutyCycle,uint32 Copy_u32freq)

This API should generate a PWM signal on Channel B of the Atmega32 target with the following parameters:

- Duty cycle: Copy_u8DutyCycle (0 to 255)
- Frequency: Copy_u32freq in Hz

4. Implementation

The implementation of the PWM module should use conditional directives like #ifdef #if #endif to optimise the code and close the sections of the code that will not be used based on the pre-compiler configuration for the modes that will be used.

5. Conclusion

This document has defined the requirements for implementing a PWM module on the Atmega32 target. The module should support two modes of operation, two polarities, and provide three APIs. The implementation should use conditional directives to optimize the code.