

Purpose:

This module provides a set of interface functions to perform a simplified initialization of the A/D converter system on the Freescale MC9S12C32 microcontroller. The initialization function should be called before any use is made of Port AD. After the initialization is complete, reads of the analog input pins should be made using the `ADS12_ReadADPin()` function. Reads of digital input pins on the AD port should be made by reading from `PTIAD`. Writes to outputs defined on port AD should be performed by making assignments to `PTAD`.

Header:

The header file `ADS12.h` should be included in any module wishing to use the functions provided by this module.

Revision History:

December 18, 2008 added to note to description for `ADS12_ReadADPin()` to include the need for a delay
November 13, 2004 First release.

Initialization**Function:**

```
ADS12ReturnTyp ADS12_Init(char * modeString)
```

Parameters:

`char [9] modeString`
A null terminated string of 8 ASCII characters to describe the mode of each of the pins on Port AD. The legal values are: `I` for digital input, `O` for digital output, `A` for analog input. The string positions, reading left to right, correspond to the pins MSB to LSB (`modeString[0]=MSB`, `modeString[7]=LSB`)

Returns:

`ADS12_Err` if the input string is malformed
`ADS12_OK` if the mode string was OK

Description:

Initializes Port AD data direction register & `ATDDIEN` for digital I/O and the A/D converter to multi-channel, continuous conversion.

Notes:

Assumes a 24MHz bus clock, but simply sets the default values for now. Enforces a single block of A/D channels even though it is possible to make a contiguous sequence that occupies a non-contiguous block. For example, channels 6,7,0,& 1 are sequence contiguous but occupy 2 blocks so would be rejected by this code.

Usage:

```
if (ADS12_Init("OOIIAAAA") == ADS12_OK)
```

would initialize the A/D converter and port AD with bits 6 & 7 as outputs, 4 & 5 as inputs and bits 0-3 as analog inputs and test to see that the initialization was successful.

Analog Pin Read**Function:**

```
short ADS12_ReadADPin( unsigned char pinNumber)
```

Parameters:

`unsigned char PinNumber`
the Port AD pin number to read the analog value from

Returns:

the A/D conversion result for the selected pin (10 bits right justified) if the pin number is legal, -1 otherwise

Description:

Reads and returns A/D conversion result from the appropriate A/D converter register.

Notes:

With a 24MHz bus clock, a delay of at least ($7\mu\text{S} * \text{number of channels in the sequence}$) must elapse between the call to `ADS12_Init()` and the first call to `ADS12_ReadADPin()`.

Usage:

`ADS12_ReadADPin(0)`

Would read the analog value on bit 0 of port AD.