API List PSDK
for the SG100 TRNG Security Generator Linux/Windows source code drivers

Overview - This Document

This document describe API functions of the SG100 Linux/Windows.

API List

```
uint32 Test_Datatypes_OK( void);

void GB_Format_to_str( char *Tempstr, uint32 Bytes_GB, uint32 Bytes);

void Setup_TRNG_System( char *Port_Specification);

uint32 SG100_Random( uint32 Select_Range);

void Get_Noise( uchar *Output_Buffer, uint32 Output_Buffer_Size);

uint32 Noise_Buffer_Size( void);

void Memory_Buffered_Bytes( uint32 *Buffer_GB, uint32 *Buffer_Bytes);

void All_Buffered_Bytes( uint32 *Buffer_GB, uint32 *Buffer_Bytes);

void Normalize( uint32 *Number_GB, uint32 *Number);
```

Datatypes

The used datatypes are declared in **m_types.h**:

!!MAKE SHURE!!

that the datatypes are set up properly for your target system!

The file machine.h includes Win32->UNIX translations for process synchronisation and similar functions. This file enables WIN32 compatibility for other projects!

API function uint32 Test_Datatypes_OK(void);

This compile-time constant function merely check if sizeof(uint32) is four bytes. When you are certain that the datatypes are set up properly, you may comment-out this API call.

```
if ( ! Test_Datatypes_OK())
{
   /* There is a problem with an integer constant,
      and you must change  */
   /* a datatype in m_types.h   */
   exit( -1);
}
```

API function void Setup_TRNG_System(char *Port_Specification);

This function take the port specification and fork a thread for a driver thread. The Linux driver support only a single such driver thread.

```
#ifdef WIN32
   Setup_TRNG_System( "COM3");
#endif


#ifdef LINUX
   Setup_TRNG_System( "/dev/ttyS0");
#endif
```

API function void Get_Noise(uchar *Output_Buffer,
uint32 Output_Buffer_Size)

The "Get Noise" is identical to the binary Win32 driver. It output random bytes to a memory buffer. Output_Buffer_Size may be zero. Don't set Output_Buffer to NULL. The API is multi-thread safe.

API function uint32 SG100_Random(uint32 Select_Range)

This function is identical to the SG100_Random function in the WIN32 driver. The function return a number in range (inclusive) [0..(Select_Range-1)]. Example: an ordinary dice roll 1,2,3,4,5,or 6 is obtained by Dice = 1 + SG100_Random(6);

If Select_Range == 1 the function always returns zero.

Select_Range may not be zero. The maximum Select_Range is 0xFFFFFFFF (32 bits).

This function is multi-thread safe.

API function uint32 Noise_Buffer_Size(void)

The function return current allocated memory buffer size.

Unit is in 32 bit uint32:s. Multiply by four to convert to bytes.

This function is multi-thread safe.

API function void Memory_Buffered_Bytes(uint32 *Buffer_GB, uint32 *Buffer_Bytes)

This function return the number of memory-buffed bytes in the system.

The unit is in bytes for Buffer_Bytes, and in (1024)^3 bytes for Buffer_GB.

The returned size is not normalised.

This function is multi-thread safe.

API function void All_Buffered_Bytes(uint32 *Buffer_GB, uint32 * Buffer_Bytes)

This function return the total number of buffered bytes in the system, The returned values are normalised. **This function is multi-thread safe.**

API function void Normalize(uint32 *Number_GB, uint32 *Number)

This function adjust the "Number_GB" value making the "Number" value less than one GB. "Number" is unsigned and cannot be negative. This function is multi-thread safe.

API function void GB_Format_to_str(char *Tempstr, uint32 Bytes_GB, uint32 Bytes);

This function convert a "size" into string form in char *Tempstr. char pointer *Tempstr should point to a target string of sufficient length. This function use "sprintf", so it is maybe not multi-thread safe on all systems.