With regard to SHA1_Hash_DRBG (...): This was the RNG in FIPS 186-2.

1. Proposed text:

   "The original specification required that the length of the *seed* (*seedlen*) needed to be between 160 and 512 bits, where 512 bits is the input block size for SHA-1. This assumed that the seed would have full entropy (i.e., the seed would have at least 160 bits of entropy). In this Standard, the minimum entropy requirement for this DRBG is between 160 and 512 bits (i.e., $160 \leq entropy \leq 512$), requiring a *seed* of at least *entropy* bits. The length of the *seed* (*seedlen*) may be greater than the entropy requirement, depending on the entropy source, up to a maximum of 512 bits (i.e. $entropy \leq seedlen \leq 512$). Note that the use of more entropy than the minimum value will offer a security "cushion". The seed is used to determine the initial state of the DRBG. Further requirements for the seed are provided in Section 9.4."

2. The call to the NRBG is now **NRBG** (*entropy*), which is a function that acquires a string of bits from an Approved NRBG or an Approved DRBG (or chain of Approved DRBGs) that is seeded by an Approved NRBG. The parameter indicates the minimum *entropy* that is required for the string.

3. The *state* consists of:

   1. (Optional) The *purpose* of the DRBG instantiation (if the DRBG is used for multiple purposes, requiring multiple instantiations, then the *purpose* **shall** be indicated, and the implementation **shall** accommodate multiple *states* simultaneously; if the DRBG will be used for only one purpose, then the *purpose* **may** be omitted),
   2. The value (V) that is updated during each call to the DRBG,
   3. The initial value (t) for the hash function for the indicated *purpose*,
   4. The *entropy* of the *seed*, and
   5. (Optional) A transformation of the *seed* or initial *state* value using a one-way function for later comparison with a new *seed* or initial *state* when the DRBG is reseeded (this value **shall** be present if the DRBG will potentially be reseeded; it **may** be omitted if the DRBG will not be reseeded).

4. The call to the DRBG has the following inputs: integer ([*purpose*], *requested_no_of_bits*, *no_of_bits_per_block*, *requested_strength*, *UserInput_flag*).

5. In the specification, the value of b was used in the following statements, where $160 \leq b \leq 512$.

   $XVAL = (XKEY + XSEED_j) \bmod 2^b$.

   $W_t = c \| 0^{512-b}$

   $XKEY = (1 + XKEY + x_j) \bmod 2^b$.

Comment [ebb1]: FIPS 186-2 stated in Appendix 3 that "The algorithms employ a one-way function G(t,c), where t is 160 bits, c is b bits ($160 \leq b \leq 512$), and G(t,c) is 160 bits. w

Comment [ebb2]: Page: 47 If the entropy is close to 512, then the actual seed could be > 512 bits. How do we deal with that ?

Comment [ebb3]: This is used to determine the state when multiple states are used and the appropriate value for t. In DSS, a different t is used for private key generation than is used for per-message secret generation.

Comment [ebb4]: This is in the G function.

I've substituted the value of *entropy* for *b*; does this seem to be the right thing to do? Note that I've also renamed the variables, hopefully for clarity and consistency with other DRBGs in X9.82.

$data = (V + UserInput) \bmod 2^{entropy}$.

$M = data \| 0^{512\text{-}entropy}$.

$V = (1 + V + returned\_bits) \bmod 2^{entropy}$.

Comment [ebb51]: Was