GET /* for (;clack;) return buf;* HI OK we are making a new kind of clock here, y for, special purposes.
 * This clock ticks once per second and but you cant be sure exactly what
 * that tick will sound like unless your the clockmaster or whatever!
 * The value is stored in the lower 26 bits of a 4 byte ping packet data.
 * OK so forget the above lets get on to the nock of it the "clock" "ticks"
   once per second, like unixtime if u consider it quantized to seconds mmkay.
 * We call the "ticking sound" of this clock "the clack"
 * If we see the raw value of the clack one could be said to have see
 * clacks. The clock signal can be considered as 26 bit at 1hz or 1 bit at 26hz
 * So in order to eat ram burn cpu and make things more complex, there is
 * a special format for clock transport and storage that is 4 bytes per
 * second, and the cannonical storage format and wire protocol and the raw
 * presentation style are thus: raw presentation display format 1a is 4 hex bytes in ascii like:
 * aa bb c dd or 0xaabbccdd to store the clacking into a file. he most based display format is type b clack display format
 * it is the display in binary form of the clack raw format or value
 * so its 32bytes long for 32 bits or 26 bytes long for 26 bits uuuuuuuus time to get on with how the format works
 * the raw clack is stored in the middle 26 of 2^5 aka the first 3 and the
 * last 3 bits of the 32bit raw storage format are the clackalope or
 * the raw dod clacket packet. There is 4 different clacket envolope modes, and the mode is entirely
 * determined by the value of the 26 bit raw clack
 * so the clack must be greater than > 2^5 or 32 of course, and the most
 * 26 bits can be in decimal * is 67108863 so from 33 to 67108863 is valid clack range.
 * 00000000000000000000100001 = CLACK_MIN
 * 11111111111111111111111111 = CLACK_MAX
 * so for the 4 six bit clackolope modes
 * here they are:
 * 001 100 if clackvalue == 2601 or (clackvalue == (65347568 - 1)
 * 010 010 if clackvalue % ((2601 - 8) || (2601 + 8))
 * 100 001 if clackvalue odd
 * 110 011 if clackvalue even
 * 0 == ((65347568/26/26/26/26) - (11 × 13));
 * 86399-65535 = 20864
 * so how to make a clack? need to collect all of its 26 bits!
 * we need 1 random bit * we need the current unixtime * we need access to the dogma 2^12 page
 * we need our tracking buffer (64*8)
 * 2^14 = 16384 page_SZ * 2 == 8192
R = Random Bit 1 bitD = DOGMA PAGE BITLOOP 5 bits 32768 bitsU = USERS LETTER 5 bits
H = The hour of the shift, as 3 bits 0-7
M = The current minute of the hour, as 6 bits 2^6 0-59
S = The current second of the minute, as 6 bits 2^6 0-59
S = 6 bit , 2^6, 0-59 "second of the minute"
M = 6 bit , 2^6, 0-59 "minute of the hour"
H = 3 bit , 2^3, 0-7 "hour of the shift"
DRDDDD UUUUU HHH MMMMMM SSSSSS
DRDDDDUUUUUHHHMMMMMMSSSSSS
DRDDDD MUMHMUMUMUHMUHM SSSSSS
DRDDDDMUMHMUMUMUHMUHMSSSSSS
    5   4
    6   2
RDDDDD UUUUU HHH MMMMMM SSSSSS