```
        """Function that converts the variable 0, normally associated with the plot X axis.
        The codification is done as follows:
               7   6   5   4     3   2   1   0
        Byte7  SGM SGE E9  E8    E7  E6  E5  E4         SGM - Signal of Mantissa: 0 - Positive 1 - Negative
        Byte6  E3  E2  E1  E0    M51 M50 M49 M48        SGE - Signal of Exponent: 0 - Positive 1 - Negative
        Byte5  M47 M46 M45 M44   M43 M42 M41 M40         E[9:0] - Exponent
        Byte4  M39 M38 M37 M36   M35 M34 M33 M32         M[51:0] - Mantissa.
        Byte3  M31 M30 M29 M28   M27 M26 M25 M24
        Byte2  M23 M22 M21 M20   M19 M18 M17 M16
        Byte1  M15 M14 M13 M12   M11 M10 M9  M8
        Byte0  M7  M6  M5  M4    M3  M2  M1  M0
        """
        self.data[n] = unpack("d", value)[0]


    def _set_steps(self, step_info):
        self.step_info = step_info

        self.step_offsets = [None for x in range(len(step_info))]

        # Now going to calculate the point offset for each step
        self.step_offsets[0] = 0
        i = 0
        k = 0
        while i < len(self.data):
            if self.data[i] == self.data[0]:
                #print(k, i, self.data[i], self.data[i+1])
                if self.data[i] == self.data[i+1]:
                    i += 1  # Needs to add one here because the data will be repeated
                self.step_offsets[k] = i
                k += 1
            i += 1

        if k != len(self.step_info):
            raise LTSPiceReadException("The file a different number of steps than expected.\n" +
```