

Subject: MCR-10047, Fix vtocx\_to\_record to handle 3381 drives

Author: Eric Swenson

Date: January 17, 2018

## Introduction

vtocx\_to\_record is a command that converts an octal VTOCE index to a Multics record number and sector offset. It computes the incorrect values when used on a 3381 drive.

## Problem

If you invoke this program, without specifying a device type, or when specifying a d451 or 3381 device type, you get the same results:

```
vtocx_to_record (10 11)
    vtocx 10 = Rec 11, rs 11; abs sect 231 (d451)
    vtocx 11 = Rec 11, rs 14; abs sect 234 (d451)
r 16:24 0.090 0 level 2

vtocx_to_record (10 11) d451
    vtocx 10 = Rec 11, rs 11; abs sect 231 (d451)
    vtocx 11 = Rec 11, rs 14; abs sect 234 (d451)
r 16:24 0.055 0 level 2

vtocx_to_record (10 11) 3381
    vtocx 10 = Rec 11, rs 11; abs sect 231 (3381)
    vtocx 11 = Rec 11, rs 14; abs sect 234 (3381)
r 16:24 0.175 0 level 2
```

This is incorrect. The 3381 case should provide different values for the absolute sector and rs values.

This command includes the two internal subroutines, below:

```
mulphy: proc (rec) returns (fixed bin);
    dcl (sect, rec) fixed bin;

    sect = rec * 16;
    sect = sect + divide (sect, usable, 17, 0) * unusable;
    return (sect);
end mulphy;

phymul: proc (sect) returns (fixed bin);
    dcl (r, sect) fixed bin;

    r = divide (sect, sect_per_cyl (dvt), 17, 0) * unusable;
    return (divide (sect - r, 16, 17, 0));
end phymul;
```

In both of those procedures, the constant 16 is used. This value represents the number of sectors in a record. This value is 16 for most disk devices, such as the d500, d451, d400, d191, d181, and d501. However, for 3380 and 3381 disks, this value should be 2.

## Proposed Changes

There is a constant array defined in fs\_dev\_types\_sector.incl.pl1, called sect\_per\_rec, which provides the correct value for “sectors per record” based on the provided device type index.

The proposed fix is to change the above two procedures to:

```
mulphy: proc (rec) returns (fixed bin);
  dcl (sect, rec) fixed bin;

  sect = rec * sect_per_rec (dvt);
  sect = sect + divide (sect, usable, 17, 0) * unusable;
  return (sect);
end mulphy;

phymul: proc (sect) returns (fixed bin);
  dcl (r, sect) fixed bin;

  r = divide (sect, sect_per_cyl (dvt), 17, 0) * unusable;
  return (divide (sect - r, sect_per_rec (dvt), 17, 0));
end phymul;
```

The proposed fix is courtesy of Charles Anthony.

## Testing of the Change

Run the `vtoctx_to_record` command on various vtoctx indices and verify that the correct value is displayed for various device types.

For example:

```
vtoctx_to_record (10 11)
  vtoctx 10 = Rec 11, rs 11; abs sect 231 (d451)
  vtoctx 11 = Rec 11, rs 14; abs sect 234 (d451)
r 16:24 0.090 0 level 2

vtoctx_to_record (10 11) d451
  vtoctx 10 = Rec 11, rs 11; abs sect 231 (d451)
  vtoctx 11 = Rec 11, rs 14; abs sect 234 (d451)
r 16:24 0.055 0 level 2

vtoctx_to_record (10 11) 3381
  vtoctx 10 = Rec 14, rs 0; abs sect 300 (3381)
  vtoctx 11 = Rec 14, rs 1; abs sect 301 (3381)
r 16:24 0.175 0 level 2
```

## Bug Reference

- Reference URL of Multics Change Ticket: <http://multics-trac.swenson.org/ticket/95>.

## Documentation

The change is the documentation. No further document needed.

## Version History

Date	Revision	Author	Comment
2018-01-17	1.0	Eric Swenson	Initial version of MCR.