



# Defensive Reactions

By Randy Moss, December 22, 2005 in Hacking Documentation

[Follow](#) 0[Start new topic](#)[Reply to this topic](#)

## Randy Moss

Starter



Members

+ 1

242 posts

Posted December 22, 2005

[Report post](#)

I've played around a little with the defensive reaction bytes at 1DC10 ... No really good info to report, but I tried switching around the bytes from different plays to see what happened.

It seemed like the defense would play pass coverage based on those bytes as if the original passing play was running.

For example, if I switched the shotgun bomb pass defense bytes with the bytes for a passing play with mostly short routes, then called the short passing play, the defense almost always covers the deep routes as first priority and leave the short routes open. So it was like the defense was playing a shotgun bomb scheme, when I copied those defensive bytes onto another play.

I'm thinking that the defensive bytes are really just pointers to another string of code that actually contains the defensive logic.

Has anyone found another set of defensive code like that?

One idea based just on this info, is you could set up the defensive reactions to plays based on deep, medium, or short coverage. For example you could change the bytes for all the passing plays so that if the defense calls a 4th slot play, they always get the shotgun bomb, aka deep coverage, etc. There could be a strategy in calling passing plays to take away deep route or short routes based on the slot that you call.

[+ Quote](#)

## bruddog

Down with button mashing



Posted December 22, 2005

[Report post](#)

I thought I've said all along that they are just pointers to the defensive code. I think I stumbled on to part of where the pointers pointed to a while back but thought it wasn't very



Moderators

+ 3,074

11,466 posts

Location: Ca

+ Quote



worthwhile to mess with stuff unless you are going to create entirely new plays like soby and jstout are doing.

jstout

Tecmo Super Champion



Members

+ 92

537 posts

Posted December 22, 2005

Report post

Defense is designed in a similiar manner to offense. It isn't tricky as much as detailed stuff you have to wander through to figure out.  
In a nut shell, the offense chooses a play and will select the defensive reaction number depending on what slot the defense chooses for that play. That number is a pointer a line of 22 hex numbers for the defense in the x6010 area. Every 2 hex numbers are the pointers for each defensive player's logic for the play (pass rush or pass coverage) in the xA010 section. If you've worked with the offense some you should understand it is basically the same process but in different areas.

+ Quote



bruddog

Down with button mashing



Moderators

+ 3,074

11,466 posts

Location: Ca

Posted December 22, 2005

Report post

I thought I remembered it being in the 6000 area. Again i thought it was only worth doing if you were going to design entirely new plays which is a signfigant undertaking.

+ Quote



Randy Moss

Starter



Posted December 22, 2005

Report post

Yeah, I've made two new offensive plays so far out of the slot formation. One is a pitch to the bottom, the other is a pass play. Updated the play graphics too.  
I started to work with defensive logic because the pass coverage wasn't as good as I wanted it to be on the one pass play I designed. The run defense seems fine for the pitch I



Members

+1

242 posts

made.

I think I follow you Jstout, but how do you explain the randomness with the defenders reactions? For example if you make all of the defensive bytes the same number, the defenders do react differently. If you call the same slot on defense they react differently. At times they cover everyone, other times they will double wrs, leaving others open, etc. Also, the A010 section? How does that relate to the 6000 section?

+ Quote



## bruddog

Down with button mashing



Moderators

+3,074

11,466 posts

Location: Ca

Posted December 22, 2005

Report post

You know how in offense at around 4400 or whatever the exact location there are 22byte strings for each offensive play. 2 bytes for each offensive. Some plays have multiple 22 string bytes depending on the play slot called. Anyways those bytes are just more pointers to additional actions.

The same exact thing applies for defense. They bytes at 6000 are just more pointers to additional defensive actions.

Ya there is some randomness to the defensive coverage and it might be tricky to figure out how that really works.

+ Quote



## jstout

Tecmo Super Champion



Members

+92

537 posts

Posted December 22, 2005

Report post

✓ Randy Moss said:

I think I follow you Jstout, but how do you explain the randomness with the defenders reactions? For example if you make all of the defensive bytes the same number, the defenders do react differently. If you call the same slot on defense they react differently. At times they cover everyone, other times they will double wrs, leaving others open, etc.

Also, the A010 section? How does that relate to the 6000 section?

The randomness is built into each player's design. For example, I created one defense for testing were the CB has one reaction to rush the QB and another to cover deep. D1 C0 28 EC (where to lineup) 27 55 A1 (My code for rushing the QB at xA165) FF 18 BE (My code for covering deep at xBE28). The computer goes through the list of commands one by one deciding to do it or not. So you can make the defensive player do the same thing everytime (like the DL pass rush) or be extremely random.

The 6010 section like the offensive play area. It goes in order of reaction # and is a direct pointer (45AA points to xAA55).

+ Quote



## bruddog

Down with button mashing



Moderators

+ 3,074

11,466 posts

Location: Ca

Posted December 24, 2005

Report post

Interesting. How many different choices does the computer go through for each defender. Is it different per play or a fixe amount per play. Like say 3 possible choices of what each individual defender.

I suppose it would be possible to make it so that picking play slots made it so that you were picking defensive formations rather than picking the specif play and have the blitzing be more random but maybe that muddies things up to much.

Like pass/run slot 1 would be a goaline d,

slots2 would be a 4/3

slots 3 would be a 3/4

slot 4 would be a dime

Could that work. OR is it a big pain since you would also have to reprogram all the reactions for each player you move

and not just there starting location? I assume you might if the reactions tell the player to move distance x from his starting spot.

+ Quote



## jstout

Tecmo Super Champion



Members

+ 92

537 posts

Posted December 24, 2005

Report post

I haven't examined every single defensive line but I've seen 2 or 3 reactions per player as pretty common. The key though is the randomness is programmed in like a % so it shouldn't hurt to have more. 20 = 99.999% of the time will run it to 2F = 99.999% of the time it will not run it. There isn't much on defense that goes on since it is line up, do something like move or pause, then do what you are supposed to be doing. Basically, the player endings are Get the Ballcarrier, Stand, m2m a specific player, or Cover deep.

Making the defensive be slot related like you detailed would actually not be that hard. Might be tough on space though, not sure on that aspect. The harder part though is you would need to have updated blocking (run and pass) on offense for each major change in defense since the 43 and 34 would be highly different though a 43 to 42 nickel or 41 dime is nearly the same.

+ Quote



## averagetsbplayer

Tecmo Legend



Posted August 20, 2009

Report post

I just want to summarize my interpretation for two reasons: 1. to make sure I'm correct and 2. to potentially help others understand.

From x1DC10-x1DE00, we have 512 (64 plays, each with 8 potential reactions) 1-byte pointers that point to 22-byte strings in the x6010-x75F9 area. When in the x1DC10-x1DE00 range, x00 would point to x6010, x01 would point to x6026, x02 would point to x603C, etc.

The x6010-x75F9 range is made up of 22-byte strings. These 22-byte strings consist of 11 pointers (each 2-byte direct pointers). Each of these 11 pointers points to the instructions for an actual position (in the normal order: RE,NT,LE,ROLB,RILB,LILB,LOLB,RCB,LCB,FS,SS).

So, here's a more concrete example:

Offense calls T FAKE SWEEP R (run slot 1, play 1)

Defense calls run slot 2

Run slot 1's play one has its 8 possible defensive reactions defined from x1DC10-x1DC17.

5B 5A 5A 58 58 58 59 59

Since the defense called run slot 2, we actually use x5A as our pointer. If the defense called run slot 1, we'd use x5B. If the defense called pass slot 1, we'd use x58.

We can then figure out where the 22-byte string in the x6010-x75F9 range is located using x5A.

$(x5A * x16) + x6010 = x67CC$  or

$x5A * 22 \text{ bytes per defensive play string} + \text{starting location} = \text{location of } x5A\text{'s defensive play}$

The 22-byte string starting at x67CC is as follows:

10 A1 1D A1 2A A1 D1 A9 41 A1 9A A9 A9 A9 7B A9 B3 A9 33 A9 C0 A9

From here, we can see that each position has the following direct pointer for this defensive play:

```

RE   - 10 A1 (xA120) [swap the two bytes, add x10 for the NES rom header
NT   - 1D A1 (xA12D)
LE   - 2A A1 (xA13A)
ROLB - D1 A9 (xA9E1)
RILB - 41 A1 (xA151)
LILB - 9A A9 (xA9AA)
LOLB - A9 A9 (xA9B9)
RCB  - 7B A9 (xA98B)
LCB  - B3 A9 (xA9C3)
FS   - 33 A9 (xA943)
SS   - C0 A9 (xA9D0)

```

At this point, I'm still not 100% sure how to interpret what is happening at xA120. I know that the following bytes are used:

D0 D8 08 EA FC 04 00

However, I'm still in the process of interpreting those. I'll update this when I get a chance.

+ Quote



2018: Green Bay - Not Good; Madison - Sweet 16 | 2017: Green Bay - Not Good; Madison - t-5th | 2016: Madison - Sweet 16 | 2015: Green Bay - 2nd Place; Madison - Elite 8, Ohio - Not Good, Iowa - Not Good | 2014: Nebraska - 2nd Place; Madison - Sweet 16; Ryder Cup - Winner (Team Madison); Iowa - Winner | 2013: Nebraska - Elite 8; Madison - Round of 32; Ohio - Sweet 16; Iowa - Final 6 | 2012: Madison - Sweet 16; Ohio - Sweet 16 | 2011: Madison - Round of 32; Ohio - 2-3 in triple-elim | 2010: Madison - Elite 8; Ohio - Sweet 16; Chicago - Final 4 | 2009: Madison - Round of 32; Ohio - Elite 8 | 2008: Madison - Round of 32

## bruddog

Down with button mashing



Moderators

+ 3,074

11,466 posts

Location: Ca

Posted August 20, 2009

Report post

That is correct. See this thread for play command detail  
[viewtopic.php?f=22&t=11941](http://viewtopic.php?f=22&t=11941)

+ Quote



## buck

DARRELL GREEN



Members

+ 2,060

6,332 posts

Location: Tecmo Super Street

Tecmo Titles: Lincoln V (2015)

Posted August 20, 2009

Report post

▼ averagetsbplayer said:

I just want to summarize my interpretation for two reasons: 1. to make sure I'm correct and 2. to potentially help others understand.

Thanks for this.

▼ averagetsbplayer said:

At this point, I'm still not 100% sure how to interpret what is happening at xA120. I know that the following bytes are used:  
D0 D8 08 EA FC 04 00

This isn't the same play example you're using but another example:

\* xD0 x36 x28 xEC xFF xAF xBE and xD0 x50 x48 xEC xFF xAF xBE would be two different (defender) lineups but both would run the play design of \$BEAF (maybe you'll make some sense of this when compared to your D0 D8 08 EA FC 04 00.)  
\* in the xA000-xBFFF area is the code for the (defender) lineup and x6010 uses the values from x1DC10 to point to them  
\*(from jstout conversation)

+ Quote



"The right to speak and the right to refrain from speaking are complementary components of ...  
'individual freedom of mind.'"

[link to change one's signature](#)

## bruddog

Down with button mashing



Moderators

+ 3,074

11,466 posts

Location: Ca

Posted August 20, 2009

Report post

From jstout's play design

D0 (D8 08)= Set Position with respect to ballplacement (Y,X) = Set position 3 yards up 1 yard back

EA= 3pt stance

FC (04 00)= Set to Grapple (0100 0000) = Set to Grapple Left Tackle

+ Quote



## buck

DARRELL GREEN



Members

+ 2,060

6,332 posts

Location: Tecmo Super Street

Tecmo Titles: Lincoln V (2015)

Posted August 20, 2009

Report post

bruddog said:

From jstout's play design

D0 (D8 08)= Set Position with respect to ballplacement (Y,X) = Set position 3 yards up 1 yard back

EA= 3pt stance

FC (04 00)= Set to Grapple (0100 0000) = Set to Grapple Left Tackle

So is there an straight-forward way to translate (D8,08) 'positioning' to yards?

+ Quote



"The right to speak and the right to refrain from speaking are complementary components of ...  
'individual freedom of mind.'"

[link to change one's signature](#)

## Yak

Tecmo Fanatic



Posted August 20, 2009

Report post

buck said:

So is there an straight-forward way to translate (D8,08) 'positioning' to yards?

I made an excel spreadsheet a while ago to help me remember. I lost it, though, on a backup of my hard drive. I usually make spreadsheets to remind me of silly shit like this.



Quote



Members

+ 1,541

5,286 posts

Location: Boston, MA

## bruddog

Down with button mashing



Moderators

+ 3,074

11,466 posts

Location: Ca

Posted August 20, 2009

Report post

you can make a table in excel...  
and then just have it spit out yardage values when you type in the values  
however you probably want the opposite for creating your own hacks..  
Type in a yardage value and have it spit out the hex via a lookup table.  
i'll try and post a spreadsheet when i get a chance.



Quote



Reply to this topic...



GO TO TOPIC LISTING

RECENTLY BROWSING 1 MEMBER

SBlueman