Okay, so this doc is , as Xeeynamo put it, is for people to look at and think “so that’s what an ANB does”.

I’ll jump into it, so first off, download the stuff in kkdf2’s gitlab for the msetDoc. Get it here: https://gitlab.com/kenjiuno/msetDoc

This is because it’s better to look back and forth between ours to get a more thorough understanding.

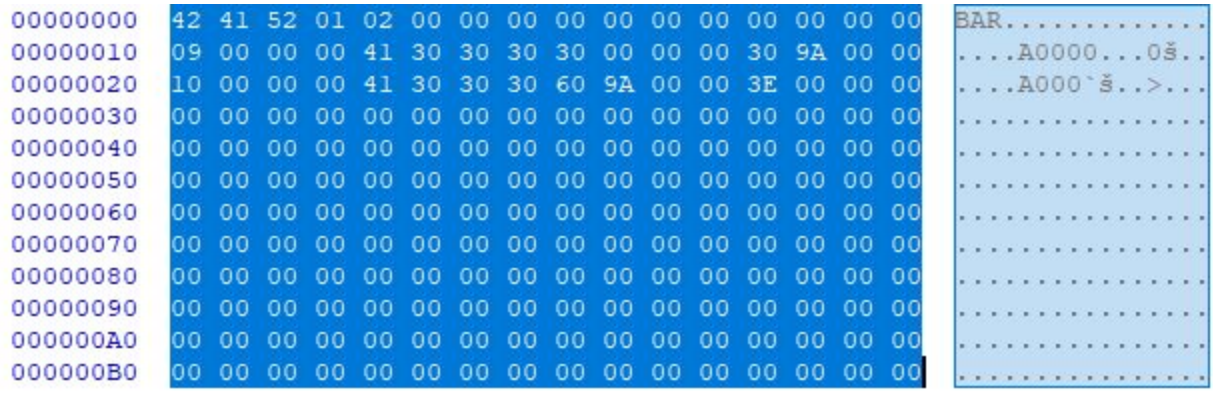
Then Download this: https://drive.google.com/file/d/1vVFq4DSImXR7wTT1eoUKIFMAoihKzVeq/view?usp=sharing, it’s a group of files to help follow along.

ANBs are stored in MSETs, as I’m sure you know, so extract one (I’ve provided you with A000, P_EX100’s Battle Idle, since it’s the one use in kkdf2’s docs, it’s easier to follow along with) and all things should apply the same

First (again), open the anb (A000) in the hex editor of your choosing, I’m using HxD, then set the bytes to 16 bytes a line. The first 3 lines should be

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	42	41	52	01	02	00	00	00	00	00	00	00	00	00	00	00	BAR.....
00000010	09	00	00	00	41	30	30	30	30	00	00	00	30	9A	00	00	....A0000...0š..
00000020	10	00	00	00	41	30	30	30	60	9A	00	00	3E	00	00	00	....A000`š..>...

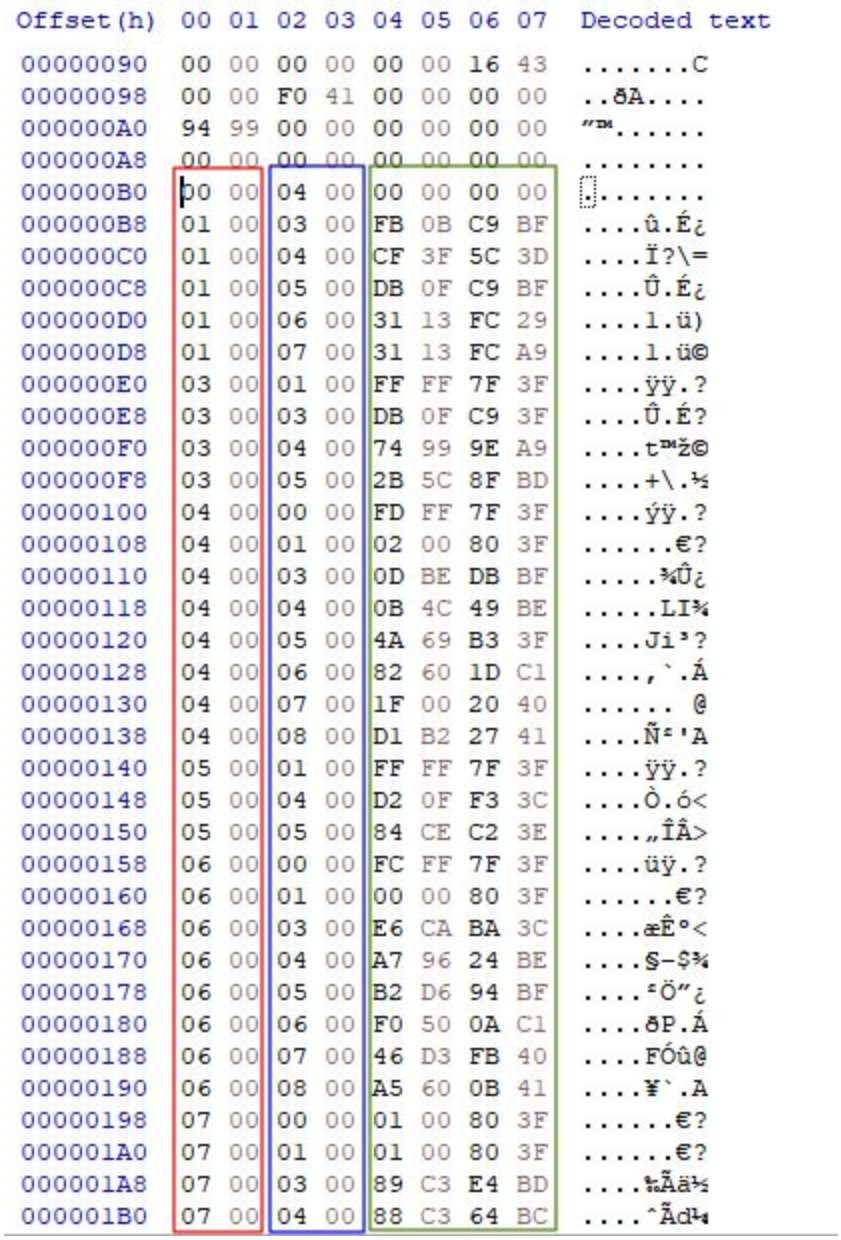
This is just telling what anb it is, nothing groundbreaking.



The first 12 lines are useless in terms of doing things in the game, in this case, I deleted them, "`A000 top chunk off so that it matchs kkdf2's formating for the first part`" is the result of this. *Note, re add the deleted lines at the end of the process, I'm simply telling you to delete lines in order to make viewing the bytes more manageable, and to follow the initial docs.

Now, change the bytes per line to 8, and then go to offset 000000B0. This is where the fun begins.

Scroll Down.



Offset(h)	00	01	02	03	04	05	06	07	Decoded text
00000090	00	00	00	00	00	00	16	43	C
00000098	00	00	F0	41	00	00	00	00	..ðA....
000000A0	94	99	00	00	00	00	00	00	"™.....
000000A8	00	00	00	00	00	00	00	00	....
000000B0	00	00	04	00	00	00	00	00	....
000000B8	01	00	03	00	FB	0B	C9	BF	....û.É¿
000000C0	01	00	04	00	CF	3F	5C	3D	....Ï?\=
000000C8	01	00	05	00	DB	0F	C9	BF	....Û.É¿
000000D0	01	00	06	00	31	13	FC	29	....1.ü)
000000D8	01	00	07	00	31	13	FC	A9	....1.ü©
000000E0	03	00	01	00	FF	FF	7F	3F	....ÿÿ.?
000000E8	03	00	03	00	DB	0F	C9	3F	....Û.É?
000000F0	03	00	04	00	74	99	9E	A9	....t™ž©
000000F8	03	00	05	00	2B	5C	8F	BD	....+\.½
00000100	04	00	00	00	FD	FF	7F	3F	....ýÿ.?
00000108	04	00	01	00	02	00	80	3F	.....€?
00000110	04	00	03	00	0D	BE	DB	BF	.....¾Û¿
00000118	04	00	04	00	0B	4C	49	BE	.....LI¾
00000120	04	00	05	00	4A	69	B3	3F	....Ji³?
00000128	04	00	06	00	82	60	1D	C1	....‚`.Á
00000130	04	00	07	00	1F	00	20	40	..... @
00000138	04	00	08	00	D1	B2	27	41	....Ñ²'A
00000140	05	00	01	00	FF	FF	7F	3F	....ÿÿ.?
00000148	05	00	04	00	D2	0F	F3	3C	....Ò.ó<
00000150	05	00	05	00	84	CE	C2	3E	....„ÎÂ>
00000158	06	00	00	00	FC	FF	7F	3F	....üÿ.?
00000160	06	00	01	00	00	00	80	3F	.....€?
00000168	06	00	03	00	E6	CA	BA	3C	....æÊº<
00000170	06	00	04	00	A7	96	24	BE	....§–$¾
00000178	06	00	05	00	B2	D6	94	BF	....²Ö"¿
00000180	06	00	06	00	F0	50	0A	C1	....ðP.Á
00000188	06	00	07	00	46	D3	FB	40	....FÓû@
00000190	06	00	08	00	A5	60	0B	41	....¥`.A
00000198	07	00	00	00	01	00	80	3F	.....€?
000001A0	07	00	01	00	01	00	80	3F	.....€?
000001A8	07	00	03	00	89	C3	E4	BD	....‰Ãä½
000001B0	07	00	04	00	88	C3	64	BC	....ˆÃd¼

The Red is the bones number

The Blue is the type of modification (translate, scale, rotate)

The Green is the extent to which is is modified

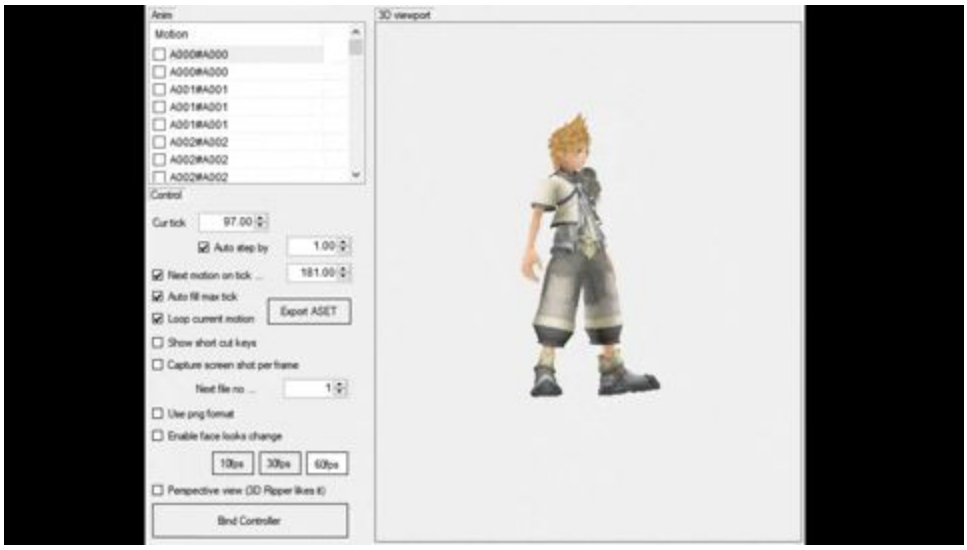
Topic	Description
joint#	Apply fixed value for ax in joint specified by joint#.
joint channel	0 = Modify Scale.x 1 = Modify Scale.y 2 = Modify Scale.z 3 = Modify Rotate.x 4 = Modify Rotate.y 5 = Modify Rotate.z 6 = Modify Translate.x 7 = Modify Translate.y 8 = Modify Translate.z Others = ?
value	A decimal value to define initial pose.

*The Chart above displays the values that correspond to the action of the bone. Full credits to kkdf2 for the chart.

P_EX100 has 227 bones, most if not all of which are present, and the bones are labeled in hex, so bone 1 is “01” and bone 227 is “E3”.

As kkdf2 (from now on referred to as Mr. Kenjiuno) discovered, the bytes in each line “destroy the mdlx”, in other words, it poses the model into the desired shape bone by bone until you get the main pose wanted.

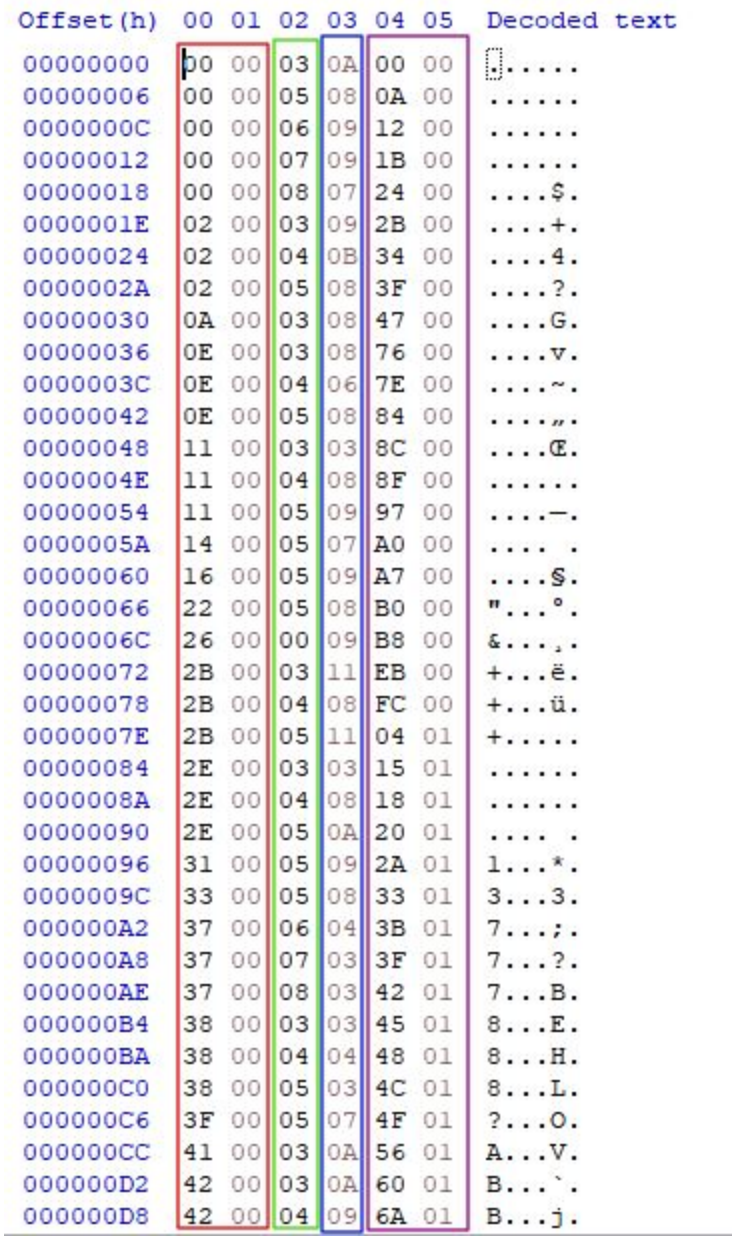
The above image is an example of me messing with what is the X Axis Rotate Value on bone 2.



All other bones not paired to bone 2 will be unaffected.

Now is where things get a lot more tricky to understand, but bare with me.
Go to line offset 00000F88 (remember, we have it set to 8 bytes per line) in "`A000 top chunk off so that it matchs kkdf2's formating for the first part,`" **the bytes should read as follows** "`00 00 03 0A 00 00 00 00`" **once** you're here, delete all bytes above this for simplicity sake, and save as a new file (re add the cut files once the process is done). Once all bytes above are gone, set bytes per line to 6.

Keep Reading



Offset(h)	00	01	02	03	04	05	Decoded text
00000000	00	00	03	0A	00	00	.....
00000006	00	00	05	08	0A	00	.....
0000000C	00	00	06	09	12	00	.....
00000012	00	00	07	09	1B	00	.....
00000018	00	00	08	07	24	00	....$.
0000001E	02	00	03	09	2B	00	....+.
00000024	02	00	04	0B	34	00	....4.
0000002A	02	00	05	08	3F	00	....?.
00000030	0A	00	03	08	47	00	....G.
00000036	0E	00	03	08	76	00	....v.
0000003C	0E	00	04	06	7E	00	....~.
00000042	0E	00	05	08	84	00	....„.
00000048	11	00	03	03	8C	00	....Œ.
0000004E	11	00	04	08	8F	00	.....
00000054	11	00	05	09	97	00	....—.
0000005A	14	00	05	07	A0	00	.... .
00000060	16	00	05	09	A7	00	....§.
00000066	22	00	05	08	B0	00	"...°.
0000006C	26	00	00	09	B8	00	&...¸.
00000072	2B	00	03	11	EB	00	+...ë.
00000078	2B	00	04	08	FC	00	+...ü.
0000007E	2B	00	05	11	04	01	+.....
00000084	2E	00	03	03	15	01	.....
0000008A	2E	00	04	08	18	01	.....
00000090	2E	00	05	0A	20	01	.... .
00000096	31	00	05	09	2A	01	1...*.
0000009C	33	00	05	08	33	01	3...3.
000000A2	37	00	06	04	3B	01	7...;.
000000A8	37	00	07	03	3F	01	7...?.
000000AE	37	00	08	03	42	01	7...B.
000000B4	38	00	03	03	45	01	8...E.
000000BA	38	00	04	04	48	01	8...H.
000000C0	38	00	05	03	4C	01	8...L.
000000C6	3F	00	05	07	4F	01	?...O.
000000CC	41	00	03	0A	56	01	A...V.
000000D2	42	00	03	0A	60	01	B...`.
000000D8	42	00	04	09	6A	01	B...j.

The Red is the bones number

The Blue is the type of modification (translate, scale, rotate)

The Green is from what I can gather, a pointer that specifies what kind of interpolation occurs and for how long

The Purple is unknown from what I've read.

*Refer to Mr. Kenjiuno's docs about this section to understand more, the explanation I've given will help to further understand what he's said: "t2" in his doc

Most if not all data below this section is dedicated to interpolation and the process of it, however, there is one section in which the IK (Inverse Kinematics) is called on. Because I can't figure this section of his docs, and because Mr. Kenjuino stated that they were speculations as he goes further into the ANB, I strongly recommend following the rest of his docs, this doc's main purpose was to give basic info to help understand the more complex stuff.

**This Doc needs to be improved, if you have any insights, please revise any errors, the goal is to have the most comprehensive, yet understandable explanation.