



Richard's C64 game projects:

Time for the flight of your life



The Up in The Air Team

Richard Bayliss - Programming, compiling, disk/tape mastering, sound effects

Richard Bayliss / Andrew Fisher - Music

Wayne Womersley - Graphics and game project head gaffer

[PAGE 1](#) / [PAGE 2](#) / [PAGE 3](#) / [PAGE 4](#)
 / [PAGE 5](#) / [Page 6](#)

Project cancelled

18th January 2011 - The Bird is the Word (Wayne)

Apparently my Bird Graphics were not good enough ! So Monday morning before work I fired

up the trusty old Sprite Editor and tweaked the birds - there looking even more sexy now (erm can sprite birds be sexy ?). I also need to go through the levels and look at how my new colour mixing affects the colours. This does not involve any additional work for Richard, we have one character set for the in game Graphics and although the screen is split the only colour changes are for the status "LOG" panel. I am such a clever boy sometimes (and others I am a numb skull but here you go).

20th January 2011 - Blame the programmer - Episode 1 (Richard)

So I'm here at home, not just looking after the pooch, but getting on with some programming. Well, I received a new email from Wayne, which is all about the colour scheme for clouds for the particular levels 3 and 7. I checked the colour table and KAPOW, the suggested were already there. Heheheh :o). Anyway, it does not mean that I should not do any more programming to this project. I have that dreaded level 5 to work on. Where the game has thunder clouds. It has been probably just before Christmas since I last took a look at this routine - and then was banging my head on the table trying to resolve the bug. It seems that nothing worked at the time. Are we going to scrap the cloud stage for level 5? ... NO ... that'd be giving up to quick. Looking at the test level I noticed that the lightning bolts were appearing from nowhere. So I took some time to take a look at the code, to see what was causing this mass of stupidity - all right then. Blame the flippin' programmer why don't you? :o)

After browsing through the code, I think to myself "What a complete mess it is for level 5". It looks as if I'm going have to reprogram level 5's lightning drops - unless ... I treat them like level 7's bombers stage, where the lightning will be dropped through a timer, just like in level 7. It might be a possible solution, but I shall take a look and see if this will resolve the problem. :) After altering the unwanted code (Simply by commenting it out with the ; command). I made some changes to the enemy type, but I also discovered another problem, which needs dealing with. I have clouds shooting out clouds - across. No that's silly. So I have a cunning plan. Good, the bullet is now a lightning bolt. Now let's change the direction of the bullet so that it will go down, as well as shoot across like a bullet. Hopefully the problem's solved after this. Awesome. The clouds are now dropping lightning. However, I need to alter the code a little to make sure the clouds will not damage the player's plane. Only the lightning can. Then fingers crossed, level 5 is ready :) - Yes. at last, it's done. I'm happy for now that is :o) - It's 2:14pm and I better have my lunch now, and the dog's waiting for me as well.

5th February 2011 - More level designs (Wayne)

Well last week I went back to the Levels and did one final big push to improve the graphics, most of what I had done was pretty good but I decided to spruce up levels 7, 10 and 15 I think (giggle !). Anyway I have a super treat for you all when you reach Level 15, I have really spruced up the graphics on this level, it took me 4 goes but ultimately it was worth it I think! (Richard can you please show an early version of my level 15 graphics please to give everyone an idea of what we are doing - thanks!). I also did one final improvement on the sprite graphics, nice, I am very pleased with the final result !



5th Febraury 2011 - Into the compactor it goes (Richard)

Today I threw this project into the compactor and watched it get mashed up and crunched into a cube :). No, wait. I was only kidding. Wayne gave me some updated level graphics, therefore I have another compression job to attend to. So bit by bit, I loaded in all the frozen screen captures of all of the level graphics. Then I loaded in the newest charset data and imported it into all of the level screens. I noticed a couple of hiccups in some of the levels, but I could easily fix those by finding a suitable char and place it into where the incorrect char lied. I used WinVice's Monitor to capture the level screen and colour data, went to the Action Replay MC monitor, and saved the screen and colour data for each level, which Wayne provided me with.

My next job was to compress all of the levels I saved to disk, using the latest version of Exomizer V2.0. I used the same level packing method as I have done before, where I can compress each colour/screen data to a specified address, where it cannot interrupt the main code. Once the job was done, I tested levels 1 to 18 to see if they decompressed correctly. Success, although I will need to do some bug fixing in my main code, as the bottom layer of the pyramids were scrolling incorrectly. I corrected this problem. I also put the raster split into the correct place. There are still some glitches with colour scroll routine. I'll have to thoroughly investigate my code to find out what is causing this problem. Not today though. To end this session, I imported Wayne's updated sprites into the game code.

6th Febraury 2011 - Blasted raster time (Richard)

After finishing the session with yesterday's work and trying to fix the colour bugs, I tried the game out on my real C64, to see what happens. Unfortunately I have come across another major problem. One which I came across before - related to the raster time. It seems that I don't have enough raster time for everything to run smoothly. I also am aware of the parallax scroll colour bugs as well. I'm guessing that I will need to rework this. I noticed that there's some routines which wasn't required. So I deleted the cloud scroll colour wrap routine. As when starting a new level, I can easily set the level to fill all clouds as white, to avoid the colour bug at the top. I reworked the colour scroll routine for the bottom layer of the screen. Hooray, no colour bugs. Now I shall make a .tap file and test the game to see if it will work 100% correct on a real C64. If it has then my problem's solved. Otherwise I'll have to work harder to find how to preserve more raster time when using the screen wrap routine. Pah!

Well, I just tested the updated UITA project on my real C64 and DC2N and it worked perfectly. So thankfully I will not have to keep mucking around with the parallax scroll routine. The mountains on level 3 and 7 also looked pretty nice on the real machine as well. I guess I'm ready

to do level 8 some time soon.

20th February 2011 - Character alterations - 4 times. (Wayne)

Hiya peeps! I stayed up late Tuesday Night (3am ish) with the Character Set, The latest Demo version of UP IN THE AIR I received off Richard seems to be missing some Characters very strange! I ended up putting 3 different Character Sets together to make the final one but it should be done now! I asked Richard a Question about the Whereabouts in the C64s Memory for the in game character set cos I wanted to load my Character set in and test it myself before I send it off so I can check for graphic bugs! Richard very kindly sent me the required information over, Today (Sunday 20th February) I loaded up the Character set into the game memory and ran it- OH NO! there is some Characters that are wrong! Switched everything off and loaded the Character set up and added the missing Characters, so that makes 4 alterations I have had to make for the Game Character Set this week. Set 1 was the Set Andrew sent over, Set 2 was the Colour Mixing I did recently on the mountains, Set 3 was the Characters for the helicopter that will represent the helicopter lives and Balloon that will represent the Balloon Quota per level instead of the word "BALLOON" (Trivia Fans might like to know I was inspired by the SUB HUNTER STATUS PANEL, I thought tiny SUBS for Lives looked great and decided to do BALLOONS to represent the BALLOON Quota as previously mentioned) and finally Set 4 I added the character that was missing from the ground and added some colour mixing to a character that formed part of the mountains that had somehow got forgotten!? So all appears to be well. Richard is now asking for information about Level 8, I guess I had better give him some help.

6th June 2011 - Going yellow (Richard)

Last week I was browsing through the game code to see how much memory was left for this game, fortunately, I have memory for a few more levels, although Wayne's graphics data took over the majority of this. So being that this week I am currently on holiday from work. I decided to kill some time to make a start and work on level 8 before I work on a new small C64 puzzle game project. So I set-up level 8's screen, I think the background colour's correct. I also altered the movement speed and enemy frames for this level. Wayne suggested that I should have yellow helicopters for this stage. As I was discussing about this level, he gave me to go ahead to add bullets to those enemies. Mainly because originally the helicopters were *not* going to shoot back. Now I added all frames, updated the helicopter's speed and tested it. The level looks ok . However, I came across a small issue. The sprite multicolour clashed with the background. So I altered the colour of the outline to medium grey, while the background was light grey. Looked much better. Just the boss to work on next, but that can be later on this week. :)

27th June 2011 - A series of ups and downs (Richard)

I thought Wayne has been waiting a bit too long for an update of Up in the Air. Since I had nothing else planned for today (Although that is quite unusual for a Tuesday), I decided to plod on a little more with this project and work on the level 8 boss stage. Oh wait a minute, this isn't going to take long is it? Well, actually it did not that that long, but there was a series of ups and downs - literally :o). I saw the SEUCK example of Wayne's level 8 boss, but it was something different. A bi-plane (bomber plane), but I was very sure the plan was changed to have an expanded Chinook helicopter as the boss, as well as the small yellow Chinooks. Well, that's something I have guessed. I shall wait for Wayne to get back to me about the level 8 boss. I have made this stage slightly harder, where the enemy will move up or down and the bullets move faster, compared to level 2's bird and level 6's Gyrocopter. The level itself is already difficult, and having to face the boss at the end could be quite a challenge as well.

Okay, now I think level 8's probably finished. It looks as if (now that I am aware of what's happening with the rest of the levels, and also checking out Wayne's SEUCK examples. It looks as if I can get this project done quicker than I thought. Actually, it would probably be MUCH better if I did that, and then work on improving the levels once the main game / levels engine's finished. However, there are bound to be delays in between this project. Well, seriously I shouldn't really delay this game, but when I feel that I have to. I do exactly that and move on to something else - just every other bored programmer :o).

As well as sorting out level 8, I proceeded on to level 9, in which the player has to face spinning stunt planes. This level's similar to level 7, except for that no planes shoot. Also, they require 4 hits before being blown to bits and pieces. What also makes level 9 harder, is the fast movement of some of the planes. You had better move out the way, unless you want to fall 16000 feet from the sky back on to the ground and smash into little pieces. Heheheheheh! :) Well, that'll do for today.



Time for a cup of tea now. - Put the kettle on Wayne :o)

11th July 2011 The land of confusion (Wayne)

It appears I have confused Richard, I sent him some SEUCK demos that contain the Level Numbers and Enemy Movement Patterns, then I decided to change the order of the Attack Waves to Improve the Variety within the game, I did send Richard the Update of how I want the Enemies to appear, eg LEVEL 1 Birds, LEVEL 2 Different Birds etc. BUT I didnt really explain this to him hence there has been a LOT of confusion. It would be so easy if we could meet up and talk, then all these misunderstandings wouldnt happen - Oh well, my bad! :-<

30th August 2011 Project Frozen (Richard)

I got fed up with this game project and heard about the RGCD 16KB cartridge game compo. So I shall hold this project back a bit and work on Woolly Jumper. A 16KB cartridge game project.

25th February 2012 What a mess (Richard)

While there was nothing but crap on television for an hour or so. I decided to look at the Up in the Air source code which I produced. Yeuch! what a huge mess I made for this game - programming wise. There's so much going on in the level source code. It desperately needs to be optimized. Well, I shall take a look at reducing the size of the code and producing more low/hi byte data tables to set up the levels. If this works, then I will have preserved more memory space for enhanced game code.

26th February 2012 A clean up operation (Richard)

After browsing through the code last night and find it to be a complete and utter mess. I decided to get started on rewriting some of the routines today, so that the code looks and feels more tidier and much easier to handle. For a start off, I will need to make some major alterations to the level setup code. I'm just constantly repeating the level build procedure too much for each level. Here's what the old level setup method I done looks like

```

;SETTING UP THE COLOURS FOR THE PLAYER SPRITE MULTICOLOURS

.SetupLevel1          lda #$0f
                      sta DefaultColourSet1+1
                      lda #$0b
                      sta DefaultColourSet2+1

;SETTING UP THE ENEMY COLOUR TABLE

                      ldx #<.Level1EnemyColourTable
                      ldy #>.Level1EnemyColourTable
                      jsr .BuildColourTable

;SETTING UP THE ENEMY OBJECT TYPE

                      ldx #<.EnemyBird1
                      ldy #>.EnemyBird1
                      stx .EnemyType1+1
                      sty .EnemyType1+2
                      stx .EnemyType2+1
                      sty .EnemyType2+2
                      stx .EnemyType3+1
                      sty .EnemyType3+2
                      stx .EnemyType4+1
                      sty .EnemyType4+2
                      stx .EnemyType5+1
                      sty .EnemyType5+2

;DEFAULT LEVEL SPRITE OUTLINE COLOURS

                      lda #$0f
                      sta LevelSprCol+1

;Enemy Speed Tables
                      ldx #<.Level1XSpeedTable
                      ldy #>.Level1XSpeedTable
                      stx LevelSpeedStore+1
                      sty LevelSpeedStore+2
;SPRITE = BULLET? #$00 = NO, #$01 = YES

                      lda #$00
                      sta .Sprite5IsBullet

;Setting the limit for enemy lives

```

```
lda #$01
sta .DefaultEnemy1Life
sta .DefaultEnemy2Life
sta .DefaultEnemy3Life
sta .DefaultEnemy4Life
sta .DefaultEnemy5Life
sta .Enemy1Life
sta .Enemy2Life
sta .Enemy3Life
sta .Enemy4Life
sta .Enemy5Life
```

```
;Setting up and decrunching the screen data via Exomizer
decrunch
;source.
```

```
ldx #<ScreenL1
ldy #>ScreenL1
stx StoreHi
sty StoreLo
jsr Decrunch
```

```
;Enemy control tables
```

```
ldx #<.Level1UControlTable1
ldy #>.Level1UControlTable1
stx StoreControlUp1+1
sty StoreControlUp1+2
ldx #<.Level1DControlTable1
ldy #>.Level1DControlTable1
stx StoreControlDown1+1
sty StoreControlDown1+2
ldx #<.Level1UControlTable2
ldy #>.Level1UControlTable2
stx StoreControlUp2+1
sty StoreControlUp2+2
ldx #<.Level1DControlTable2
ldy #>.Level1DControlTable2
stx StoreControlDown2+1
sty StoreControlDown2+2
ldx #<.Level1UControlTable3
ldy #>.Level1UControlTable3
stx StoreControlUp3+1
sty StoreControlUp3+2
ldx #<.Level1DControlTable3
ldy #>.Level1DControlTable3
stx StoreControlDown3+1
sty StoreControlDown3+2
ldx #<.Level1UControlTable4
ldy #>.Level1UControlTable4
stx StoreControlUp4+1
sty StoreControlUp4+2
ldx #<.Level1DControlTable4
ldy #>.Level1DControlTable4
```



```

stx StoreControlDown4+1
sty StoreControlDown4+2
ldx #<.Level1UControlTable5
ldy #>.Level1UContralTable5
stx StoreControlUp5+1
sty StoreControlUp5+2
ldx #<.Level1DControlTable5
ldy #>.Level1DControlTable5
stx StoreControlDown5+1
sty StoreControlDown5+2
jsr .RemoveLine ;Remove unwanted charline
jsr .SetUpEnemyYControl ;Setup the enemy control
rts

```

(Whole routine repeated each time for each level).

Now, I repeated this for 9 levels which is just too much unnecessary code. There will be a solution to this problem though. I know there is one, and will be reducing the size of THIS level setup code by creating a table which will store the low/hi-byte of each function via a level counter, and a loop. To reduce this mess I could use something like:

```

SetupLevelData
    ldx LevelCounter
;Read the default colour setting table

    lda DefaultColourSetTable1,x
    sta DefaultColourSet1+1
    lda DefaultColourSetTable2,x
    sta DefaultColourSet2+1

;Read the enemy colour setting table

    lda LevelEnemyColourTableLow,x
    sta ResetColours+1
    lda LevelEnemyColourTableHi,x
    sta ResetColours+2

;Setting up the enemy object type

    lda EnemyObjectTableLow1,x
    sta EnemyType1+1
    lda EnemyObjectTableLow2,x
    sta EnemyType2+1
    lda EnemyObjectTableLow3,x
    sta EnemyType3+1
    lda EnemyObjectTableLow4,x
    sta EnemyType4+1
    lda EnemyObjectTableLow5,x
    sta EnemyType5+1
    lda EnemyObjectTableHi1,x
    sta EnemyType1+2
    lda EnemyObjectTableHi2,x

```



```
    sta EnemyType2+2
    lda EnemyObjectTableHi3,x
    sta EnemyType3+2
    lda EnemyObjectTableHi4,x
    sta EnemyType4+2
    lda EnemyObjectTableHi5,x
    sta EnemyType5+2

;Default sprite outline colours

    lda OutLineColourTable,x
    sta LevelSprCol+1

;Enemy Speed setup

    lda LevelSpeedTableLow,x
    sta LevelSpeedStore+1
    lda LevelSpeedTableHi,x
    sta LevelSpeedStore+2

;Sprite bullet? #$00 = NO, #$01 = YES

    lda Sprite5IsBulletTable,x
    sta Sprite5IsBullet

;Setting up the number of lives an enemy can have before
;it gets killed

    lda Enemy1LivesTable,x
    sta DefaultEnemy1Life
    sta Enemy1Life
    lda Enemy2LivesTable,x
    sta DefaultEnemy2Life
    sta Enemy2Life
    lda Enemy3LivesTable,x
    sta DefaultEnemy3Life
    sta Enemy3Life
    lda Enemy4LivesTable,x
    sta DefaultEnemy4Life
    sta Enemy4Life
    lda Enemy5LivesTable,x
    sta DefaultEnemy5Life
    sta Enemy5Life

;Level screen decrunching phase

    lda ScreenDecrunchLow,x
    sta StoreLo
    lda ScreenDecrunchHi,x
    sta StoreHi

;Enemy control tables

    lda Enemy1UpControlLow,x
```

```

sta StoreControlUp1+1
lda Enemy1UpControlHi,x
sta StoreControlUp1+2
lda Enemy1DownControlLow,x
sta StoreControlDown1+1
lda Enemy1DownControlHi,x
sta StoreControlDown1+2
lda Enemy2UpControlLow,x
sta StoreControlUp2+1
lda Enemy2UpControlHi,x
sta StoreControlUp2+2
lda Enemy2DownControlLow,x
sta StoreControlDown2+1
lda Enemy2DownControlHi,x
sta StoreControlDown2+2
lda Enemy3UpControlLow,x
sta StoreControlUp3+1
lda Enemy3UpControlHi,x
sta StoreControlUp3+2
lda Enemy3DownControlLow,x
sta StoreControlDown3+1
lda Enemy3DownControlHi,x
sta StoreControlDown3+2
lda Enemy4UpControlLow,x
sta StoreControlUp4+1
lda Enemy4UpControlHi,x
sta StoreControlUp4+2
lda Enemy4DownControlLow,x
sta StoreControlDown4+1
lda Enemy4DownControlHi,x
sta StoreControlDown4+2
lda Enemy5UpControlLow,x
sta StoreControlUp5+1
lda Enemy5UpControlHi,x
sta StoreControlUp5+2
lda Enemy5DownControlLow,x
sta StoreControlDown5+1
lda Enemy5DownControlHi,x
sta StoreControlDown5+2

;Let the level value add one (increment it)
    inx ;
    cpx #21 ;Level 20 exceeded, jump to end
    beq .EndSequence
    rts

.EndSequence inc $d020
             jmp *-3 ;Silly flashy border thingy

;A subroutine for incrementing the level then jumping back ;to
the main game.

.LevelComplete
    inc .LevelCounter

```

```

        jsr .RemoveLine
        jsr .SetUpEnemyYControl
        jsr .Decruncher
        jmp .GameRefresh

.GameRefresh (Main loop)

        jsr SetupLevelData

... Rest of program ...

```

This will shorten the main level setup code a lot. Simply by adding a table in which can deal with handling the default settings for each level. I feel that this is the method I should use and it would save memory as well. Imagine what the previous code would have been like if it was 20 levels long? It would leave no room for additional programming or data. I guess that I am learning from my silly programming mistakes now. :)

So I removed the huge listing and replaced it with the shorter one. What's the result? Well, not good at the moment. For some reason, the decruncher hasn't done it's job properly when I try to point the low/high vectors to the decruncher's address. So I am going to have to take a look at where it had gone wrong. After making a correction to the source, I ended up having two of the same file names (levelsetup.a) and closed new version of the LevelSetup routine by accident. Darn. It took me a long time today to tidy up this source, but then it gets screwed up by for some reason getting over written by the old level setup source. That sucks bigtime. A great error in programming - and a waste of time too! Unable to recover that routine now. I'll have to rewrite the routine Saturday next week instead (As I will hardly have any spare time this week, due to the awkward shift pattern I get at work. Oh well. :)

3rd March 2012 - What a mess. (Richard)

I promised Wayne that I'd do some more on Up in the Air today, and I have. It is still the same compression source. I typed in the new level setup source from scratch, like I did before hand. After typing in the new level setup I successfully got the level data to decrunch the screen data and also setup the correct colour schemes. I managed to get somewhere but there's still something wrong in the source. For some apparant reason the enemies' movement seem to be misbehaving. Something surely has gone wrong here in the game code, but I don't know what has gone wrong, the main in game source code looks very messy and will need to be thoroughly read through the main game code. To be honest, I wish I didn't program such a large squab of mess. Hopefully one day, I should be able to fix the game code so that the enemies are moving the way they should have been, but not today :)

17th March 2012 - What a hoot (Richard)

Well, yet more work done today for an hour or so. You may have remembered 2 weeks ago I came across a problem in which involved the enemy movements going all over the place - and could not find a solution to it? Well, it turned out that only one tiny mistake caused the silly chaos (how embarrassing). I commented out the initialized counter to set the movements correctly by mistake 2 weeks ago. After I removed the comment from the initialized counter, I still found something else to be wrong. It was pretty much laughable as well. It turns out that the enemy bullets shoot across. So any enemy that was shooting down, was shooting across. Doh. Anyway the solution to this problem has been resurrected. I created a new table in which will work out which direction the bullets should shoot. Did this work? YES. It certainly did. Although, I was still

amused with the fact that I made a huge mistake last time, and spent ages searching through the code to find the problem and today I found the problem straight away. Another problem has occurred. The boss stages. The enemies are shooting, but the bullets cannot be seen. It appeared that I forgot to set up the correct colours for \$D026, \$D025 in the table. So I updated the table slightly to display the correct values of \$D026 and \$D025 (Fixed main colour 1+2 for all sprites). Now all objects are visible and correct to how they should have been. I shall work on level 8's boss next week (The tank) and try to get it right as well. :)

30th March 2012 - Confused again - boss tank - Which one?

What's happened today? Well, I didn't code anything for this game last week, due to increased work hours, and today I didn't do any programming on this game either. Mainly because I'm sort of disorganized and unsure which tank Wayne wants for the boss. Why is that? It is because there are 2 different tanks. I'm guessing the tank in which Wayne wants me to use is the one shooting diagonally. I best go through my old emails, to find out which tank Wayne wants me to use. Feeling shattered today anyway, so no programming today. Sunday afternoon's probably the best time to do it. ;)