

## Calculate phone\_app base address?

---

### Calculate phone\_app base address?

by [zlayaa](#) on 02 Mar 2010 19:23

Hello! Is it possible to calculate the base address of a phone\_app in a hex editor? If so, how do you do it?

Thanks in advance! :grin:

---

by [madfish](#) on 02 Mar 2010 20:20

open phone\_app.cxc(C510\_R1FA035) in IDA(with help of den\_po's ida cxcloder) and at beggining u can see something like ORG 0x14000000 and 14000000 is the base address someone correct me if i'm wrong but think that's it

---

by [zlayaa](#) on 03 Mar 2010 12:13

Oh ok, but is there any way to calculate phone\_app base addresses with an hex editor alone? :P  
Thanks in advance :pitty:

---

by [Shadow Player](#) on 03 Mar 2010 12:40

Look for the bytes

Code: [Select all](#)

18 F0 9F E5 18 F0 9F E5 18 F0 9F E5 18 F0 9F E5
---

<http://www.se-developers.net/showthread.php?tid=216>

<http://www.se-developers.net/showthread.php?tid=535>

---

where i can get the ida cxc loader?

<http://justdanpo.cheb.ru/>

**ida\_cxcldr.rar**

---

**DB3210 Base Address:**

**phone\_app.cxc = 0x10000000**

**phone\_app\_emp.cxc(or something like that) = 0x14000000**

---

## A2 Porting & Patch Bases

by [Shadow Player](#) on 04 Jan 2009 16:40

Hi Everyone!

In this thread I want to solve doubts about the A2 base confusion and how to port patches for A2.

This topic is not to learn how to port, just explain what has changed from the older platform porting.

**So, words by den\_po:**

- 1) open phone\_app.cxc in any hex-editor
  - 2) scroll down until you see something like 18 F0 9F E5 18 F0 9F E5
  
  - 3a) remove all data before address you've found
  - 4a) use 0x10000000 as base address
  
  - 3b) use 0x10000000-address you've found as base address  
(for example 0x10000000-0x2000=0xFFFE000 for w760 and  
0x10000000-0x1000=0xFFFF000 for k850)

---

## RE: The DB3210 Base Address

by [ndt](#) on 15 Sep 2009 11:33

what repeat bytes are? :?

for me 0x14000000 is good enough ;]

```
ROM:14000000 @ Segment type: Pure code
ROM:14000000      .section ROM@ CODE
ROM:14000000      @ .org 0x14000000
ROM:14000000      .code 16
ROM:14000000
ROM:14000000 @ ===== S U B R O U T I N E
=====
ROM:14000000
ROM:14000000
ROM:14000000 sub_14000000:
ROM:14000000      MOVS  R0, #0
ROM:14000002      BX    LR
ROM:14000002 @ End of function sub_14000000
ROM:14000002
ROM:14000004
ROM:14000004 @ ===== S U B R O U T I N E
=====
ROM:14000004
ROM:14000004
ROM:14000004 sub_14000004:
ROM:14000004      ADDS  R7, #0x24
```