| 0     | Mode   | s in pr | acettor  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|-------|--------|---------|--|---|--|---|-------------------------------------|---|---------|---------------|--------------------------------|------------|----------|---------|--------|---------|-------|--|--|--|--|--|
|       |        |         | visor m  | nde   | 7.   | ave all pri   | -iledas                             |   | an ch   | nnae          | the mo                         | de hi      | re in ab | a 2010  |        |         |       |  |  |  |  |  |
|       |        |         |  | quest Mod   |  |   | 3                                   |   |         | -0-           |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  | iviledges   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| 9     |        | 100     |  | s in IRQ  |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       | → di   | Herent  | Rib, Ri  | 4 (LR.SP  | )  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| 3     | When   | IRQ     | happen   | s > proces  | sor go   | es into II  | RQ med                              | le  |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        | copies  | ' CPSR   | → SPSR  |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       | 2.     | LRIRE   | +PC  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| @     | How    | does p  | TOLESOT  | know wh   | ere to   | begin exe   | cutting                             | the inter                                     | rupt :  | ervice        | routin                         | <b>e</b> ? |          |         |        |         |       |  |  |  |  |  |
|       | → 1G   | 0. E    | ecution  | Vector" i   | . @ 0  | dolress 0:  | × 18 →                              | execute                                       | there.  | B Int         | errupt_1                       | ervice.    |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| - Thi | s 15 . | just on | e kind   | of "Exce  | otion -  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| - on  | othe   | T exa   | nple is ·  | the powe  | r on/  | reset"  |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  | adolress O  |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  | servisor mo   | de                                  |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  | into IRO  |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  | ecuted e  |                                     |   |         | V:301         | mode                           |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  | exceptio  |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
| -Wh   | en ye  | u loac  | 1 progr  | oom using   | this w   | ith monit   | or proc                             | ram - mu                                      | st sei  | ect "         | Exception                      | ons in     | Linker   | Sectio  | n of t | New Dua | ject. |  |  |  |  |  |
|       |        | ( code  | d) exce  | ption vec   | tor to   | able  |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        | addn    | 186  | . Section   | .vec   | tor , "   | Δx"                                 |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        | 0×0     |  | В   | _8h  | art   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  | .space  | 20   | l Sw  | ords .                              | skipped                                       | 1       |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        | 0x18    |  | В   | IRQ  | handler   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       |        |         |  |   |  |   |                                     |   |         |               |                                |            |          |         |        |         |       |  |  |  |  |  |
|       | // IRG | _handi  | er hon   | dles au pe  | sible  | sources o   | Interr                              | upts - Lo                                     | b 6 h   | u two         | Source                         | s - ke     | & tim    | er      |        |         |       |  |  |  |  |  |
|       |        |         |  | dles an pe  |  |   |                                     |   |         |               |                                |            |          | er      |        |         |       |  |  |  |  |  |
|       |        |         | er: Pus  | H FRO-R   | a, LRì   |   | //push                              | (3ave 0                                       | ul regi | sters o       | nto the                        | Stack)     |          | श       |        |         |       |  |  |  |  |  |
|       |        |         | er: Pus  | code fig  | u, LR\   | ut which i  | //push                              | (3ave 0                                       | ul regi | sters o       | nto the                        | Stack)     |          | er      |        |         |       |  |  |  |  |  |
|       |        |         | er, Pus<br>i   | code fig  | ures c   | ut which i  | // push                             | pt occur                                      | ul regi | sters o       | nto the                        | stack)     |          | er      |        |         |       |  |  |  |  |  |
|       |        |         | er: Pus<br>:<br>::#<br>BL  | code fig<br>interrupt<br>kEY-IS   | LUPES C<br>LUPES C<br>LUPES<br>SR  | ut which i  | // push                             | (3ave 0                                       | ul regi | sters o       | nto the                        | stack)     |          | er'     |        |         |       |  |  |  |  |  |
|       |        |         | er. Pus<br>i<br>nif<br>BL<br>nif   | en fro-r<br>code fig<br>interrupt<br>KEY-IS<br>interrupt  | LANGE TO SHORE THE SHORE T | ut which i  | // push                             | pt occur                                      | ul regi | sters o       | nto the                        | stack)     |          | er .    |        |         |       |  |  |  |  |  |
|       |        |         | i<br>i<br>i<br>BL<br>aif   | code fig<br>interrupt<br>key_IS<br>interrupt<br>TIMER.  | was to LISR  | ut which i  | // push                             | pt occur                                      | ul regi | sters o       | nto the                        | stack)     |          | er      |        |         |       |  |  |  |  |  |
|       |        |         | er, pug  | code fig<br>interrupt<br>keY_IS<br>interrupt<br>TIMER.  | u. LAN<br>MAS<br>GR<br>MAS<br>GR<br>MAS<br>TER   | ut which i  | // push                             | pt occur                                      | ul regi | sters o       | nto the                        | stack)     |          | er      |        |         |       |  |  |  |  |  |
|       |        |         | i cif BL CROPPOP   | code fig interrupt keY_IS interrupt TIMER. set regis  | in Len   | key   | //push<br>interru                   | pt occur<br>o specif                          | ul regi | key<br>errupt | nto the.                       | Stack)     | ine.     |         |        |         |       |  |  |  |  |  |
|       |        |         | er, Pus<br>i<br>iiif<br>BL<br>uif<br>BL<br>uRe<br>Pof  | code fig<br>interrupt<br>kEY-IS<br>interrupt<br>TIMER.<br>Set regis                                 | ures of was in a second of the | key imer  | // push interru                     | pt occur<br>o specif                          | ic int  | key errupt    | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       |        |         | er, Pus<br>i<br>iif<br>BL<br>uif<br>BL<br>v.Re   | code fig<br>interrupt<br>key_IS<br>interrupt<br>TIMER.<br>Set regis<br>iRo-RIA<br>back to           | was on was the man the | key imer  | I/push interru ilgo:ti              | specifications of the stores c                | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       |        |         | er, Pus<br>i<br>iif<br>BL<br>uif<br>BL<br>v.Re   | code fig<br>interrupt<br>key_IS<br>interrupt<br>TIMER.<br>Set regis<br>iRo-RIA<br>back to           | was on was the man the | key imer  | I/push interru ilgo:ti              | specifications of the stores c                | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       |        |         | er. Pus<br>i<br>iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii   | code fig<br>interrupt<br>key_IS<br>interrupt<br>TIMER.<br>Set regis<br>iRo-RIA<br>back to           | unes of was a second of the se | key imer  | I/push interru ilgo:ti              | specifications of the stores c                | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRQ.   | handi   | Pus<br>i<br>i<br>i<br>bl.<br>uif<br>bl.<br>us<br>por<br>uge  | interrupt  TIMER.  Set regis  PC+LR  PC+LR  | u. LAN  LOGS  R  LOGS  R  LOGS  LOGS | key imer  | is pushinterru                      | pt occur<br>o specifi<br>at was e             | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
| Mor   | IRG.   | handii  | i iiif BL uif BL | interrupt TIMER. Set regis \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$                 | was the nasthernal   | imer  ain programmer  R                                       | Upushinterru  Ugo to  Tom the  (res | pt occur<br>o specifi<br>at was e             | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt TIMER. Set regis \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$                 | was a was the second of the se | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>at was e             | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | was a was the second of the se | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |
|       | IRG.   | subs    | BL B   | interrupt kEY_IS interrupt TIMER. set regis iRO-Ria back to PC+LR return S PC, LR. Crauses hardware | in LRY  LRY  THE R  LRY  THE R  CPS  THE R   | iner  iner  cin programmer  R + SPSR  reterrupt in  SR + CPSS | I/pushinterne                       | pt occur<br>o specifi<br>ot was e<br>stores c | rreal - | key<br>errupt | nto the<br>timer of<br>service | stack)     | ine.     | west ha | ppene  | d       |       |  |  |  |  |  |