black hat® ASIA 2021
May 6-7, 2021
BRIEFINGS

Apple Neural Engine Internal

From ML Algorithm to HW Registers

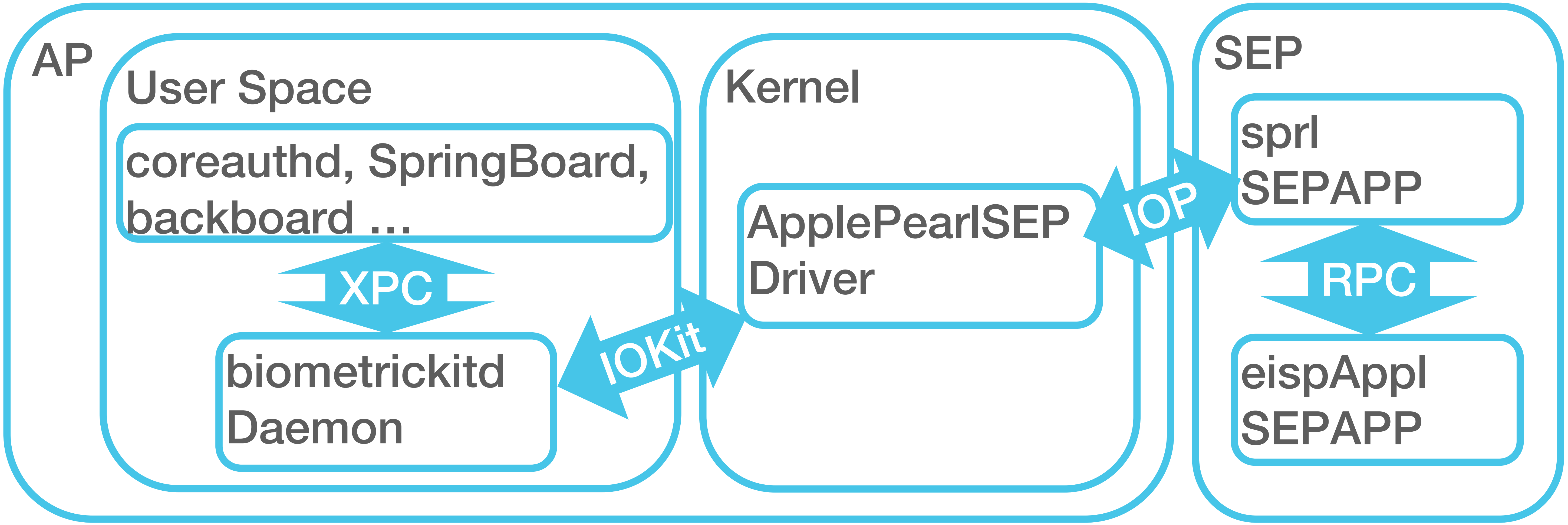
Wish Wu

Security Expert, Tian Qiong Security Lab of Ant Group

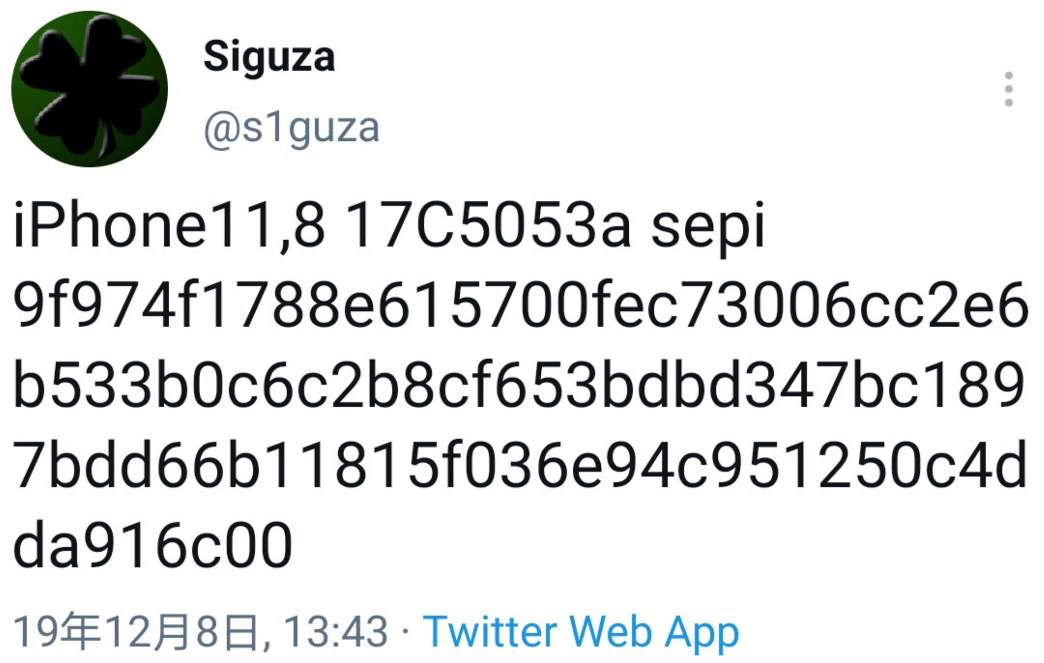
蚂蚁集团 ANT GROUP | 蚂蚁安全实验室 ANT SECURITY LAB

Contents

FaceID and Secure Neural Engine



Thanks


Siguza
@s1guza

iPhone11,8 17C5053a sepi 9f974f1788e615700fec73006cc2e6b533b0c6c2b8cf653bdbd347bc1897bdd66b11815f036e94c951250c4dda916c00

19年12月8日, 13:43 · Twitter Web App

Tools by @xerub

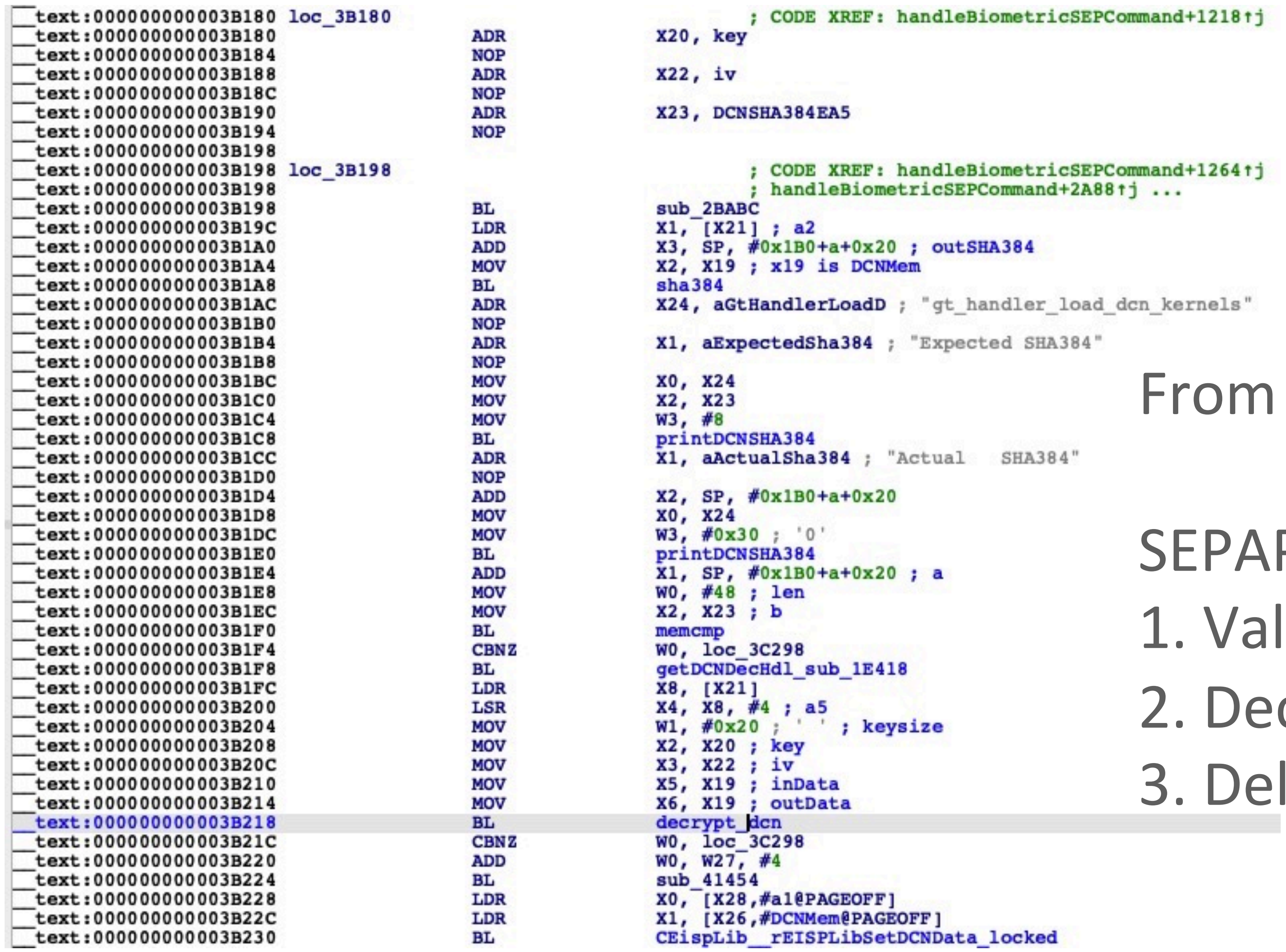
img4lib : decrypt im4p file

sepsplit : extract SEPAPP… from firmware

AES iv and key by @s1guza

decrypt SEP firmware 'sep-firmware.n841.RELEASE.im4p' black hat ASIA 2021

SEP load FaceID weights(?) from AP



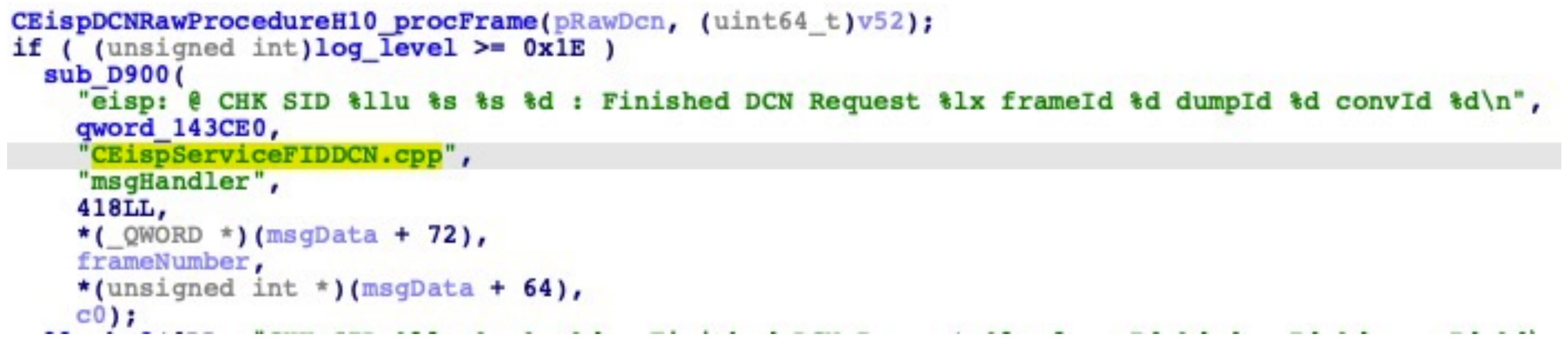
```
__text:000000000003B180 loc_3B180                               ; CODE XREF: handleBiometricSEPCommand+1218↑j
__text:000000000003B180                 ADR             X20, key
__text:000000000003B184                 NOP
__text:000000000003B188                 ADR             X22, iv
__text:000000000003B18C                 NOP
__text:000000000003B190                 ADR             X23, DCNSHA384EA5
__text:000000000003B194                 NOP
__text:000000000003B198
__text:000000000003B198 loc_3B198                               ; CODE XREF: handleBiometricSEPCommand+1264↑j
__text:000000000003B198                                         ; handleBiometricSEPCommand+2A88↑j ...
__text:000000000003B198                 BL              sub_2BABC
__text:000000000003B19C                 LDR             X1, [X21] ; a2
__text:000000000003B1A0                 ADD             X3, SP, #0x1B0+a+0x20 ; outSHA384
__text:000000000003B1A4                 MOV             X2, X19 ; x19 is DCNMem
__text:000000000003B1A8                 BL              sha384
__text:000000000003B1AC                 ADR             X24, aGtHandlerLoadD ; "gt_handler_load_dcn_kernels"
__text:000000000003B1B0                 NOP
__text:000000000003B1B4                 ADR             X1, aExpectedSha384 ; "Expected SHA384"
__text:000000000003B1B8                 NOP
__text:000000000003B1BC                 MOV             X0, X24
__text:000000000003B1C0                 MOV             X2, X23
__text:000000000003B1C4                 MOV             W3, #8
__text:000000000003B1C8                 BL              printDCNSHA384
__text:000000000003B1CC                 ADR             X1, aActualSha384 ; "Actual   SHA384"
__text:000000000003B1D0                 NOP
__text:000000000003B1D4                 ADD             X2, SP, #0x1B0+a+0x20
__text:000000000003B1D8                 MOV             X0, X24
__text:000000000003B1DC                 MOV             W3, #0x30 ; '0'
__text:000000000003B1E0                 BL              printDCNSHA384
__text:000000000003B1E4                 ADD             X1, SP, #0x1B0+a+0x20 ; a
__text:000000000003B1E8                 MOV             W0, #48 ; len
__text:000000000003B1EC                 MOV             X2, X23 ; b
__text:000000000003B1F0                 BL              memcmp
__text:000000000003B1F4                 CBNZ            W0, loc_3C298
__text:000000000003B1F8                 BL              getDCNDecHdl_sub_1E418
__text:000000000003B1FC                 LDR             X8, [X21]
__text:000000000003B200                 LSR             X4, X8, #4 ; a5
__text:000000000003B204                 MOV             W1, #0x20 ; ' ' ; keysize
__text:000000000003B208                 MOV             X2, X20 ; key
__text:000000000003B20C                 MOV             X3, X22 ; iv
__text:000000000003B210                 MOV             X5, X19 ; inData
__text:000000000003B214                 MOV             X6, X19 ; outData
__text:000000000003B218                 BL              decrypt_dcn
__text:000000000003B21C                 CBNZ            W0, loc_3C298
__text:000000000003B220                 ADD             W0, W27, #4
__text:000000000003B224                 BL              sub_41454
__text:000000000003B228                 LDR             X0, [X28,#a1@PAGEOFF]
__text:000000000003B22C                 LDR             X1, [X26,#DCNMem@PAGEOFF]
__text:000000000003B230                 BL              CEispLib__rEISPLibSetDCNData_locked
```

From /System/Library/Pearl/DCNKernels/

SEPAPP sprl load DCNKernel.bin…

1. Validate with hard code SHA384
2. Decrypt with hard code AES iv and key
3. Deliver to SEPAPP esipAppl

FaceID use Secure Neural Engine



```
CEispDCNRawProcedureH10_procFrame(pRawDcn, (uint64_t)v52);
if ( (unsigned int)log_level >= 0x1E )
  sub_D900(
    "eisp: @ CHK SID %llu %s %s %d : Finished DCN Request %lx frameId %d dumpId %d convId %d\n",
    qword_143CE0,
    "CEispServiceFIDDCN.cpp",
    "msgHandler",
    418LL,
    *(_QWORD *)(msgData + 72),
    frameNumber,
    *(unsigned int *)(msgData + 64),
    c0);
```

Call Stack:

CEispServiceFIDDCN

-> CEispDCNRawProcedureH10

-> CEispRawOperation

-> CEISPPIODMAH10

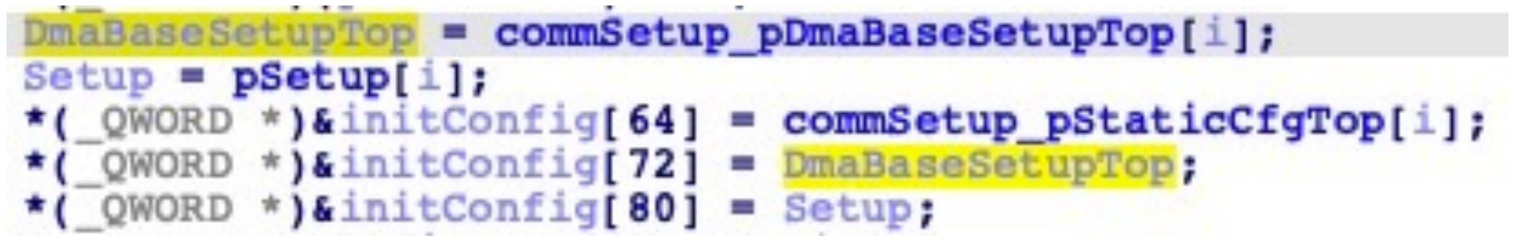
-> ffwCommon_writeReg32(addr, value)

-> ffwCommon_writeReg32(addr, value)

SEPAPP eispAppl operate hardware

FaceID use Secure Neural Engine

CEispServiceFIDDCN initialize function



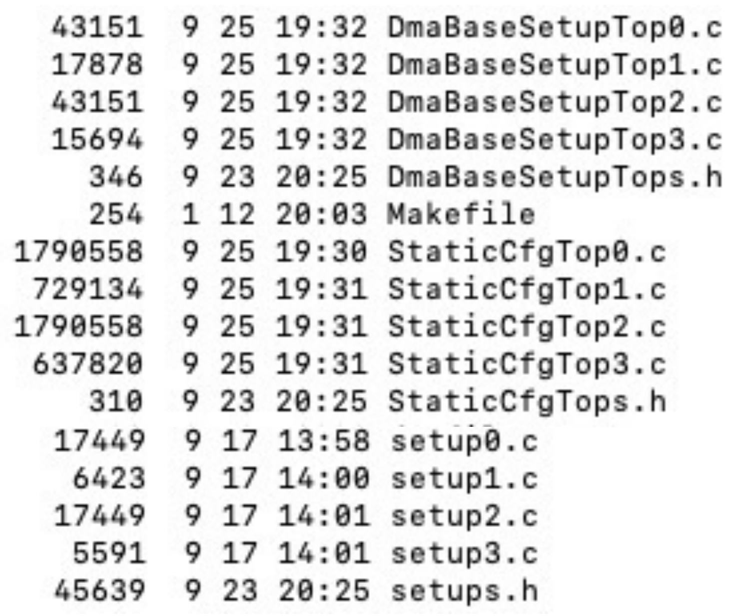
```
DmaBaseSetupTop = commSetup_pDmaBaseSetupTop[i];
Setup = pSetup[i];
*(_QWORD *)&initConfig[64] = commSetup_pStaticCfgTop[i];
*(_QWORD *)&initConfig[72] = DmaBaseSetupTop;
*(_QWORD *)&initConfig[80] = Setup;
```

[(addr, value), …] = Interpreter(DmaBaseSetup, Setup, StaticCfg)

ffwCommon_writeReg32(addr, value)

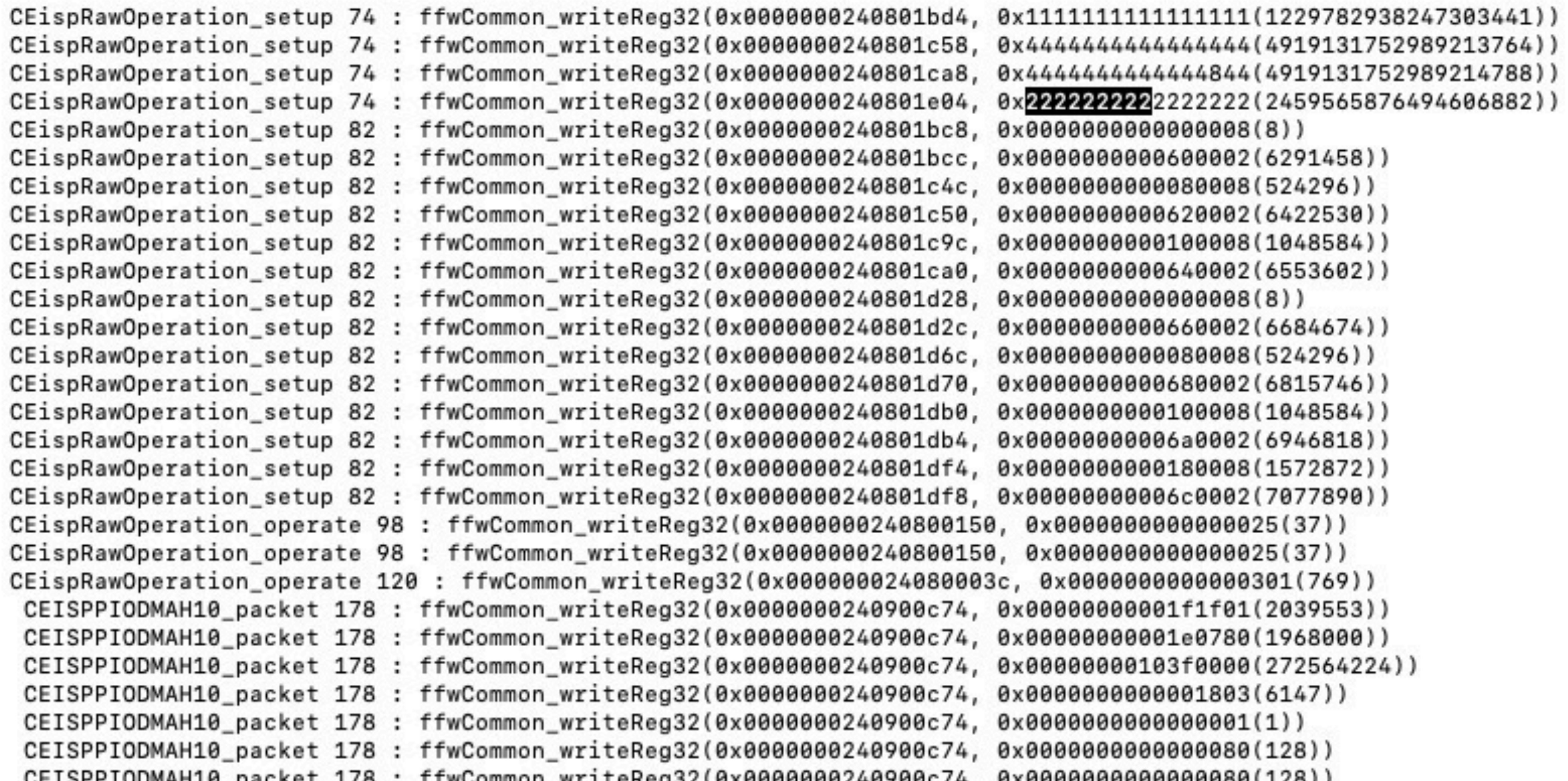
WHAT DO addr AND value MEAN?????

Print Registers Addr and Value



```
  43151  9 25 19:32 DmaBaseSetupTop0.c
  17878  9 25 19:32 DmaBaseSetupTop1.c
  43151  9 25 19:32 DmaBaseSetupTop2.c
  15694  9 25 19:32 DmaBaseSetupTop3.c
    346  9 23 20:25 DmaBaseSetupTops.h
    254  1 12 20:03 Makefile
1790558  9 25 19:30 StaticCfgTop0.c
 729134  9 25 19:31 StaticCfgTop1.c
1790558  9 25 19:31 StaticCfgTop2.c
 637820  9 25 19:31 StaticCfgTop3.c
    310  9 23 20:25 StaticCfgTops.h
  17449  9 17 13:58 setup0.c
   6423  9 17 14:00 setup1.c
  17449  9 17 14:01 setup2.c
   5591  9 17 14:01 setup3.c
  45639  9 23 20:25 setups.h
```

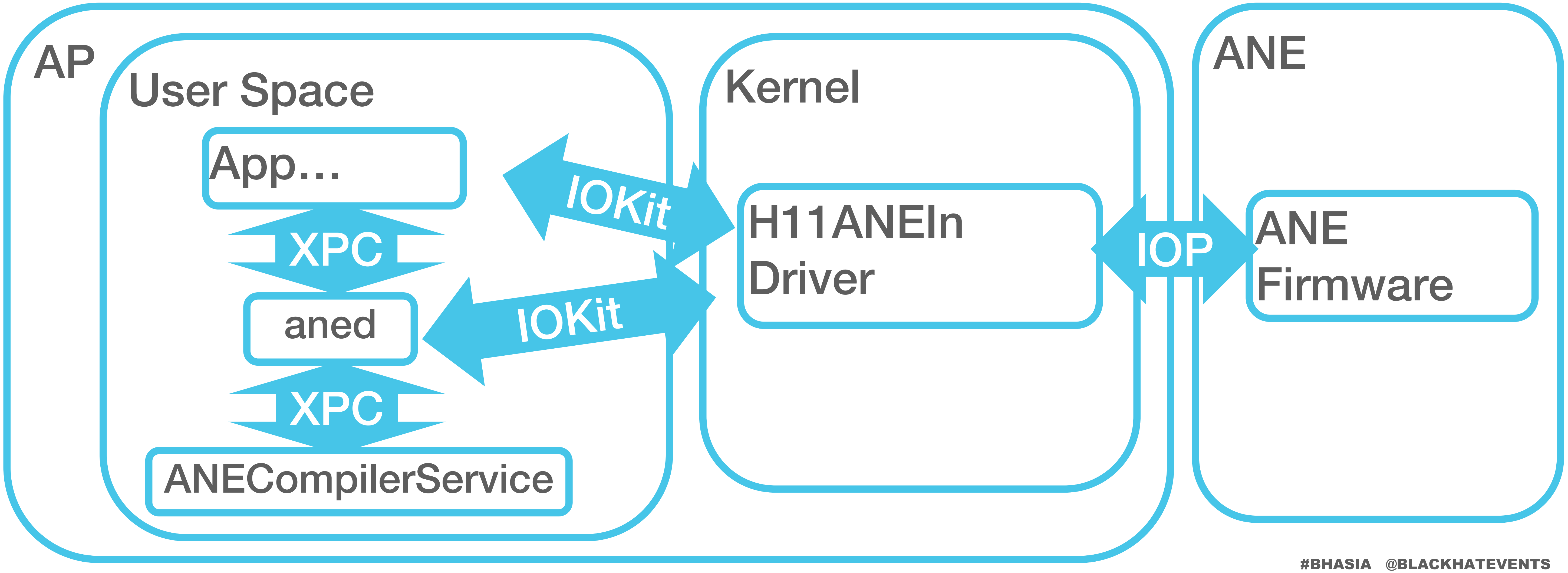
dump the data from SEPAPP eispAppl, parse it in Host



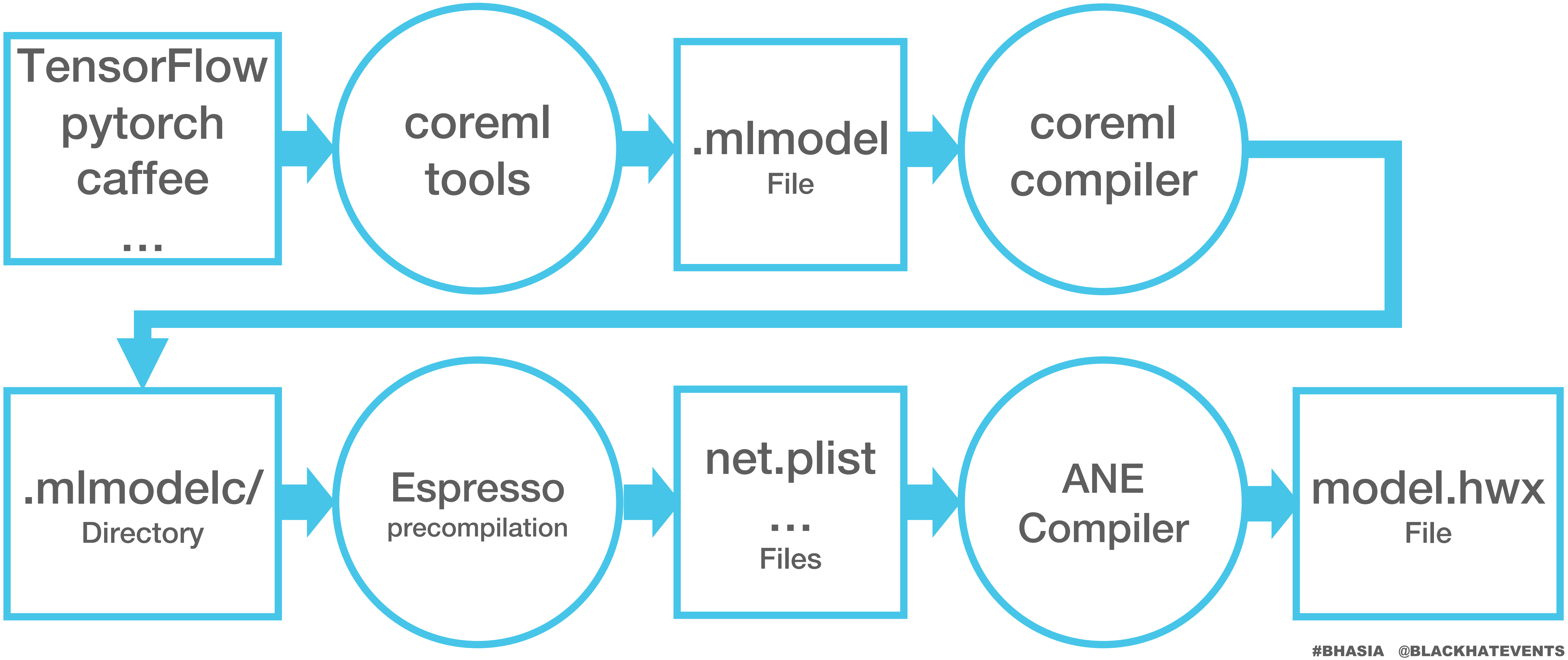
```
CEispRawOperation_setup 74 : ffwCommon_writeReg32(0x0000000240801bd4, 0x1111111111111111(1229782938247303441))
CEispRawOperation_setup 74 : ffwCommon_writeReg32(0x0000000240801c58, 0x4444444444444444(4919131752989213764))
CEispRawOperation_setup 74 : ffwCommon_writeReg32(0x0000000240801ca8, 0x4444444444444844(4919131752989214788))
CEispRawOperation_setup 74 : ffwCommon_writeReg32(0x0000000240801e04, 0x2222222222222222(2459565876494606882))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801bc8, 0x0000000000000008(8))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801bcc, 0x0000000000600002(6291458))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801c4c, 0x0000000000080008(524296))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801c50, 0x0000000000620002(6422530))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801c9c, 0x0000000000100008(1048584))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801ca0, 0x0000000000640002(6553602))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801d28, 0x0000000000000008(8))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801d2c, 0x0000000000660002(6684674))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801d6c, 0x0000000000080008(524296))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801d70, 0x0000000000680002(6815746))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801db0, 0x0000000000100008(1048584))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801db4, 0x00000000006a0002(6946818))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801df4, 0x0000000000180008(1572872))
CEispRawOperation_setup 82 : ffwCommon_writeReg32(0x0000000240801df8, 0x00000000006c0002(7077890))
CEispRawOperation_operate 98 : ffwCommon_writeReg32(0x0000000240800150, 0x0000000000000025(37))
CEispRawOperation_operate 98 : ffwCommon_writeReg32(0x0000000240800150, 0x0000000000000025(37))
CEispRawOperation_operate 120 : ffwCommon_writeReg32(0x000000024080003c, 0x0000000000000301(769))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x00000000001f1f01(2039553))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x00000000001e0780(1968000))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x00000000103f0000(272564224))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x0000000000001803(6147))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x0000000000000001(1))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x0000000000000080(128))
 CEISPPIODMAH10_packet 178 : ffwCommon_writeReg32(0x0000000240900c74, 0x0000000000000080(128))
```

STILL
UNKOWN

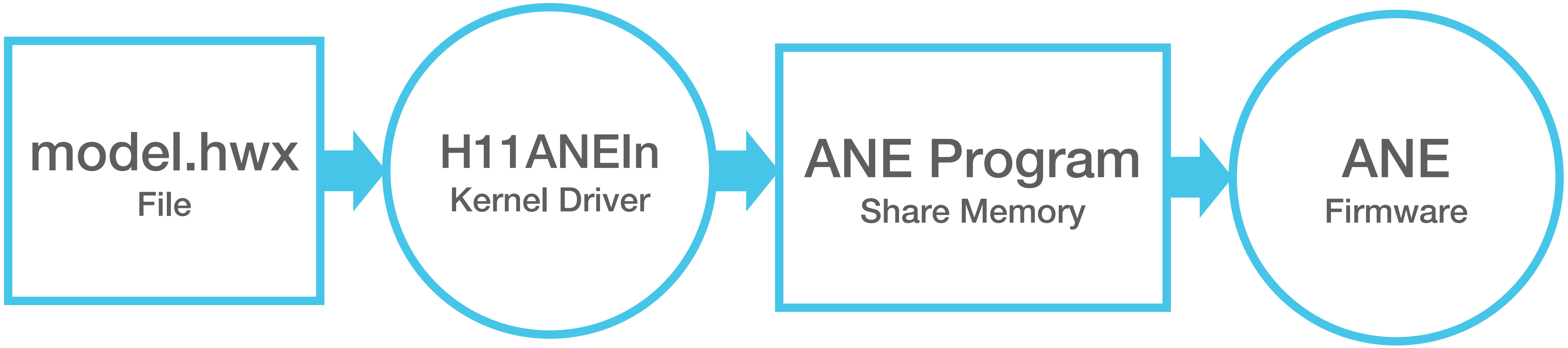
Apple Nerual Engine for App



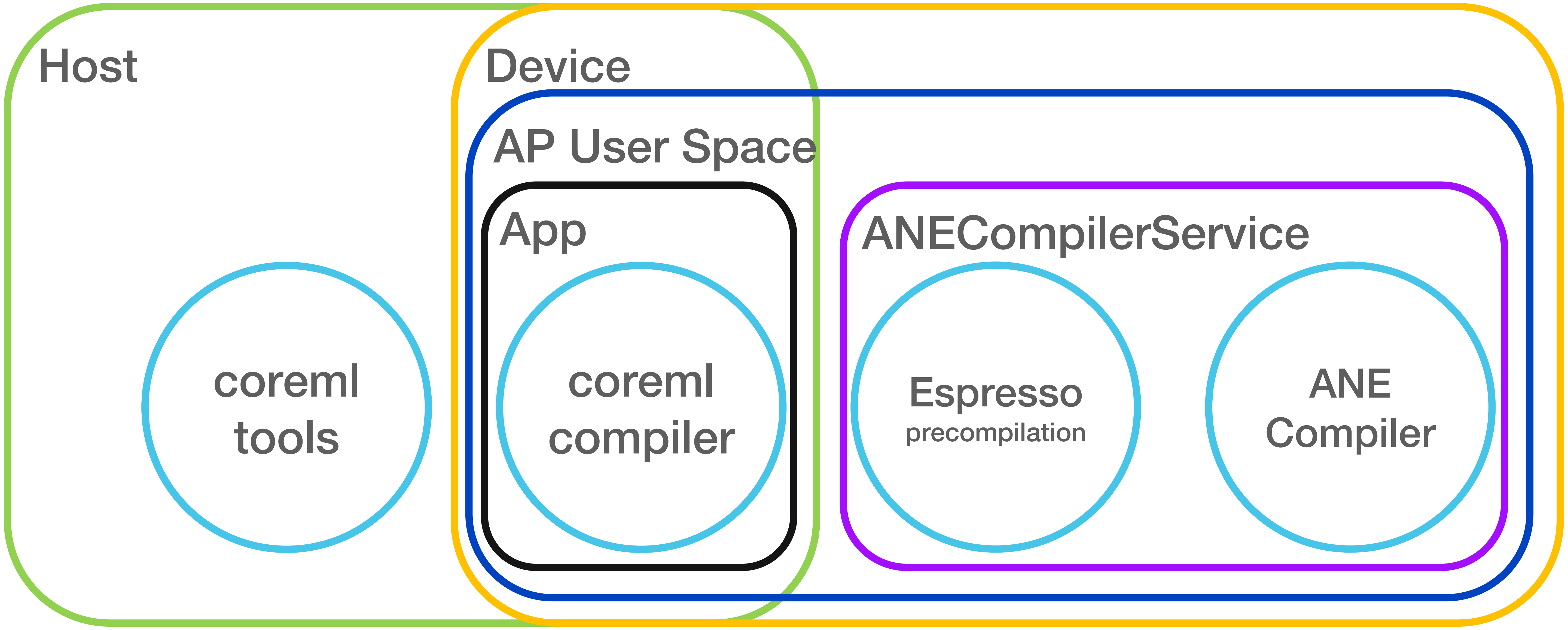
ANE Data Process Flow



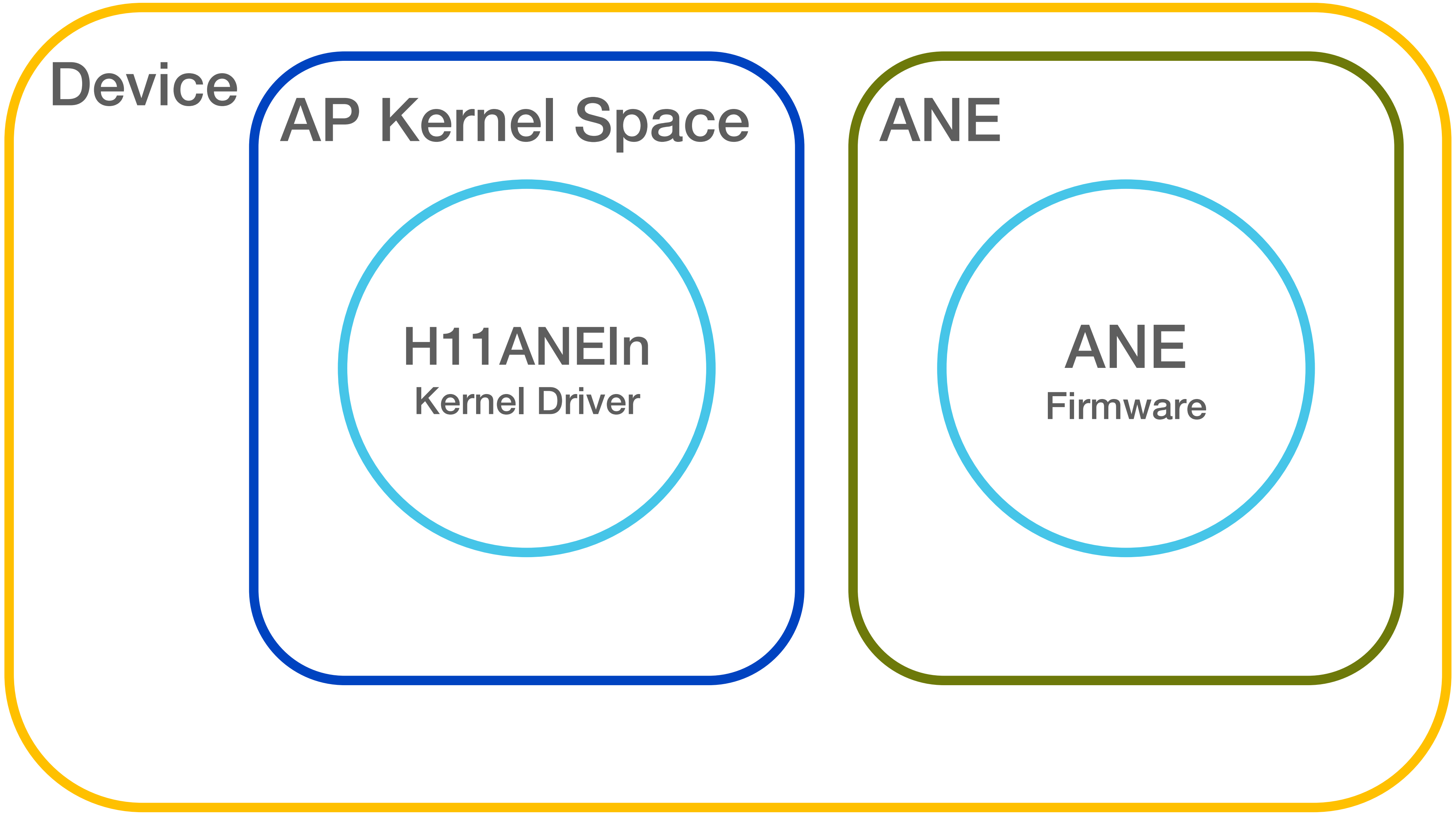
ANE Data Process Flow



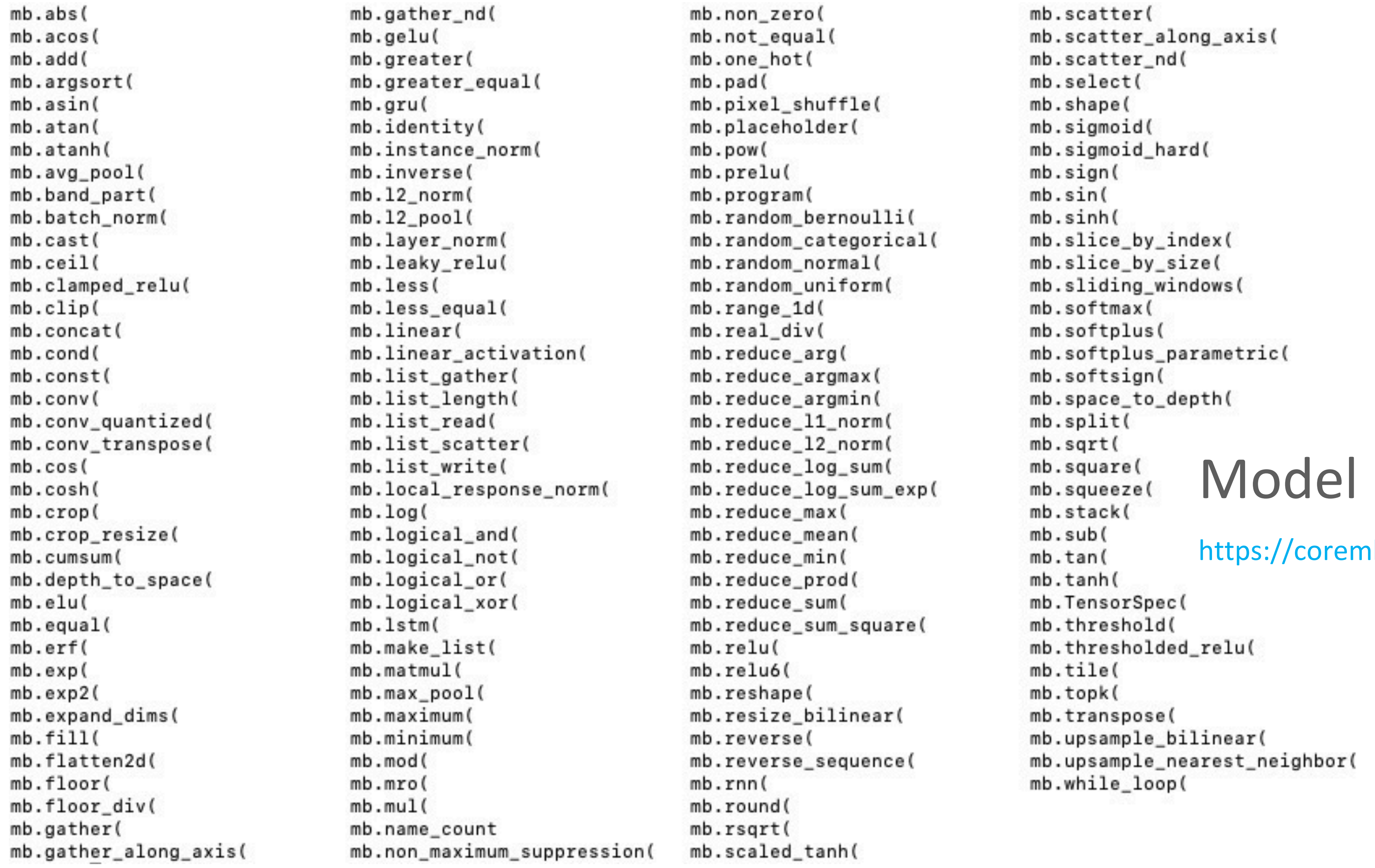
ANE Data Process Places



ANE Data Process Places



ML Algorithm by coremltools



```
mb.abs(
mb.acos(
mb.add(
mb.argsort(
mb.asin(
mb.atan(
mb.atanh(
mb.avg_pool(
mb.band_part(
mb.batch_norm(
mb.cast(
mb.ceil(
mb.clamped_relu(
mb.clip(
mb.concat(
mb.cond(
mb.const(
mb.conv(
mb.conv_quantized(
mb.conv_transpose(
mb.cos(
mb.cosh(
mb.crop(
mb.crop_resize(
mb.cumsum(
mb.depth_to_space(
mb.elu(
mb.equal(
mb.erf(
mb.exp(
mb.exp2(
mb.expand_dims(
mb.fill(
mb.flatten2d(
mb.floor(
mb.floor_div(
mb.gather(
mb.gather_along_axis(
mb.gather_nd(
mb.gelu(
mb.greater(
mb.greater_equal(
mb.gru(
mb.identity(
mb.instance_norm(
mb.inverse(
mb.l2_norm(
mb.l2_pool(
mb.layer_norm(
mb.leaky_relu(
mb.less(
mb.less_equal(
mb.linear(
mb.linear_activation(
mb.list_gather(
mb.list_length(
mb.list_read(
mb.list_scatter(
mb.list_write(
mb.local_response_norm(
mb.log(
mb.logical_and(
mb.logical_not(
mb.logical_or(
mb.logical_xor(
mb.lstm(
mb.make_list(
mb.matmul(
mb.max_pool(
mb.maximum(
mb.minimum(
mb.mod(
mb.mro(
mb.mul(
mb.name_count
mb.non_maximum_suppression(
mb.non_zero(
mb.not_equal(
mb.one_hot(
mb.pad(
mb.pixel_shuffle(
mb.placeholder(
mb.pow(
mb.prelu(
mb.program(
mb.random_bernoulli(
mb.random_categorical(
mb.random_normal(
mb.random_uniform(
mb.range_1d(
mb.real_div(
mb.reduce_arg(
mb.reduce_argmax(
mb.reduce_argmin(
mb.reduce_l1_norm(
mb.reduce_l2_norm(
mb.reduce_log_sum(
mb.reduce_log_sum_exp(
mb.reduce_max(
mb.reduce_mean(
mb.reduce_min(
mb.reduce_prod(
mb.reduce_sum(
mb.reduce_sum_square(
mb.relu(
mb.relu6(
mb.reshape(
mb.resize_bilinear(
mb.reverse(
mb.reverse_sequence(
mb.rnn(
mb.round(
mb.rsqrt(
mb.scaled_tanh(
mb.scatter(
mb.scatter_along_axis(
mb.scatter_nd(
mb.select(
mb.shape(
mb.sigmoid(
mb.sigmoid_hard(
mb.sign(
mb.sin(
mb.sinh(
mb.slice_by_index(
mb.slice_by_size(
mb.sliding_windows(
mb.softmax(
mb.softplus(
mb.softplus_parametric(
mb.softsign(
mb.space_to_depth(
mb.split(
mb.sqrt(
mb.square(
mb.squeeze(
mb.stack(
mb.sub(
mb.tan(
mb.tanh(
mb.TensorSpec(
mb.threshold(
mb.thresholded_relu(
mb.tile(
mb.topk(
mb.transpose(
mb.upsample_bilinear(
mb.upsample_nearest_neighbor(
mb.while_loop(
```

Model Intermediate Language

https://coremltools.readme.io/docs/model-intermediate-language

coremlcompiler

Host:

/Applications/Xcode.app/Contents/Developer/usr/bin/coremlc

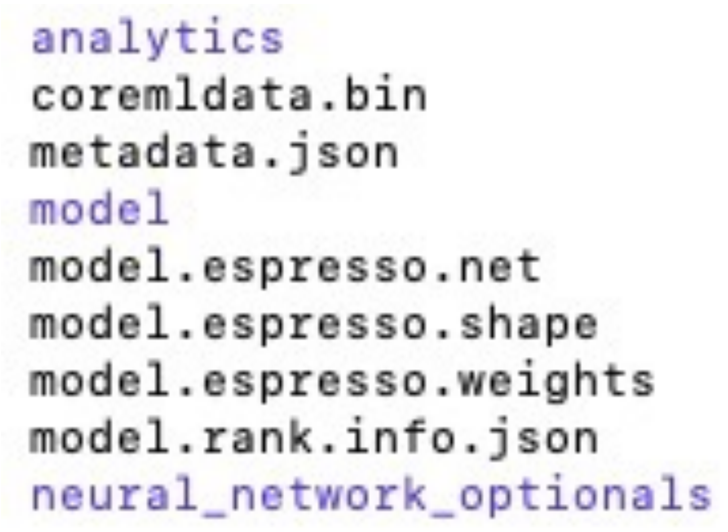
/Applications/Xcode.app/Contents/Developer/Toolchains/XcodeDefault.xctoolchain/usr/bin/coremlcompiler

Device:

CoreML class

Input: *.mlmodel file

Output: *.mlmodelc/ directory



```
analytics
coremldata.bin
metadata.json
model
model.espresso.net
model.espresso.shape
model.espresso.weights
model.rank.info.json
neural_network_optionals
```

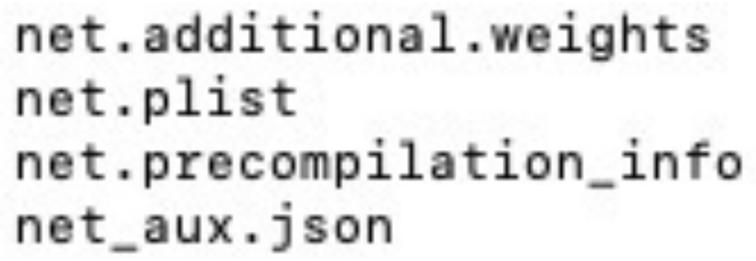
Espresso precompilation

Run in ANECompilerService process invoked by aned deamon process

```
0x1b0009cb0 Foundation!-[NSDictionary(NSDictionary) writeToFile:atomically:]
0x1dd6316d0 Espresso!Espresso::ANECompilerEngine::compiler::dump_ir(std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char>
0x1dd5a78dc Espresso!espresso_dump_ir
0x1047f1c0c ANECompilerService!0x9c0c +[_ANEEspressoIRTranslator translateModelAt:key:outputPath:error:]
0x1047f2ac0 ANECompilerService!0xaac0 +[_ANECoreMLModelCompiler compileModelAt:csIdentity:key:optionsFilename:tempDirectory:outputURL:ok:error:]
```

Input : directory which has file ‘model.espresso.net’

Output : net.plist files…

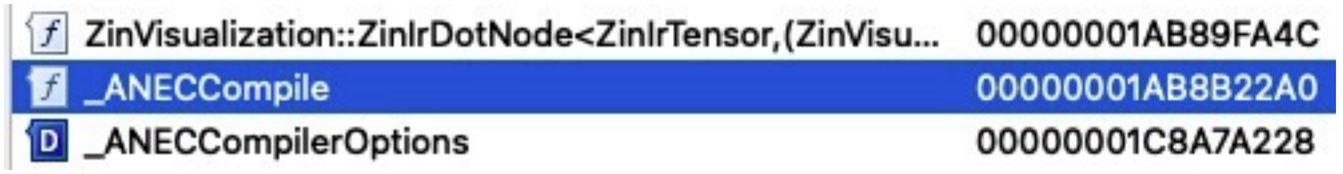


```
net.additional.weights
net.plist
net.precompilation_info
net_aux.json
```

ANECompiler

Run in ANECompilerService process invoked by aned deamon process

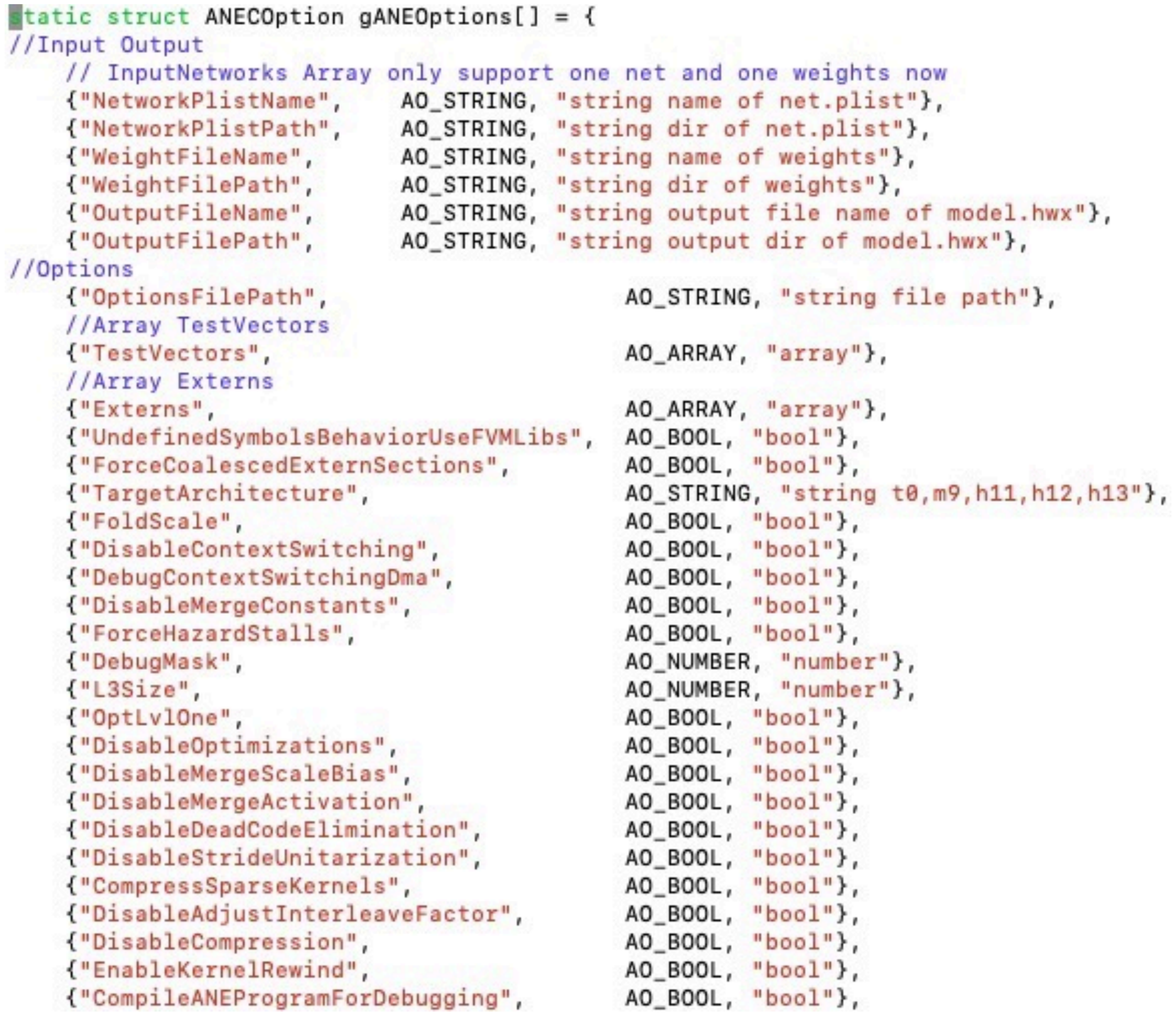
```
/System/Library/Caches/com.apple.dyld/dyld_shared_cache_arm64e
      -> /System/Library/PrivateFrameworks/ANECompiler.framework/ANECompiler
            -> int ANECCompile(CFDictionaryRef ios, CFDictionaryRef opts, uintptr_t a3);
```



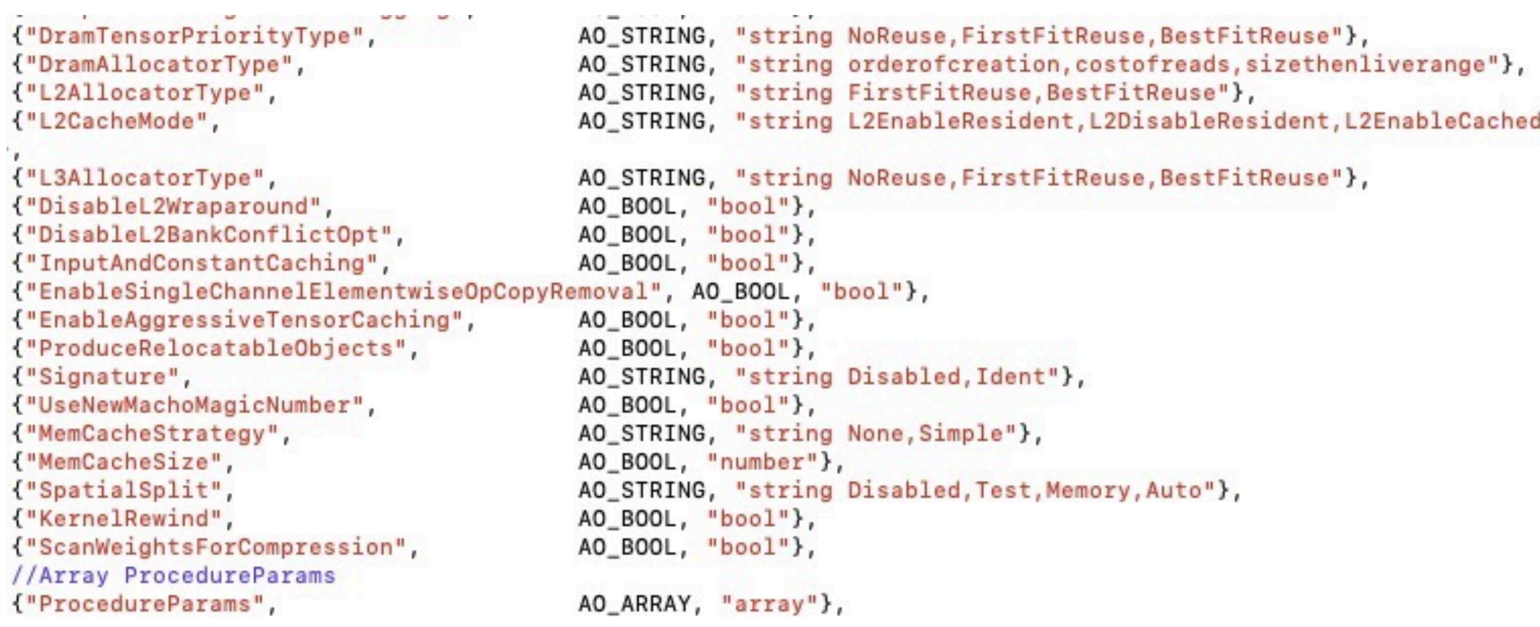
```
ZinVisualization::ZinIrDotNode<ZinIrTensor,(ZinVisu...   00000001AB89FA4C
_ANECCompile                                             00000001AB8B22A0
_ANECCompilerOptions                                     00000001C8A7A228
```

Input : net.plist file, weights file…
Output : model.hwx file

ANECompiler Internal Options



```
static struct ANEOption gANEOptions[] = {
//Input Output
    // InputNetworks Array only support one net and one weights now
    {"NetworkPlistName",    AO_STRING, "string name of net.plist"},
    {"NetworkPlistPath",    AO_STRING, "string dir of net.plist"},
    {"WeightFileName",      AO_STRING, "string name of weights"},
    {"WeightFilePath",      AO_STRING, "string dir of weights"},
    {"OutputFileName",      AO_STRING, "string output file name of model.hwx"},
    {"OutputFilePath",      AO_STRING, "string output dir of model.hwx"},
//Options
    {"OptionsFilePath",                     AO_STRING, "string file path"},
    //Array TestVectors
    {"TestVectors",                         AO_ARRAY, "array"},
    //Array Externs
    {"Externs",                             AO_ARRAY, "array"},
    {"UndefinedSymbolsBehaviorUseFVMLibs",  AO_BOOL, "bool"},
    {"ForceCoalescedExternSections",        AO_BOOL, "bool"},
    {"TargetArchitecture",                  AO_STRING, "string t0,m9,h11,h12,h13"},
    {"FoldScale",                           AO_BOOL, "bool"},
    {"DisableContextSwitching",             AO_BOOL, "bool"},
    {"DebugContextSwitchingDma",            AO_BOOL, "bool"},
    {"DisableMergeConstants",               AO_BOOL, "bool"},
    {"ForceHazardStalls",                   AO_BOOL, "bool"},
    {"DebugMask",                           AO_NUMBER, "number"},
    {"L3Size",                              AO_NUMBER, "number"},
    {"OptLvlOne",                           AO_BOOL, "bool"},
    {"DisableOptimizations",                AO_BOOL, "bool"},
    {"DisableMergeScaleBias",               AO_BOOL, "bool"},
    {"DisableMergeActivation",              AO_BOOL, "bool"},
    {"DisableDeadCodeElimination",          AO_BOOL, "bool"},
    {"DisableStrideUnitarization",          AO_BOOL, "bool"},
    {"CompressSparseKernels",               AO_BOOL, "bool"},
    {"DisableAdjustInterleaveFactor",       AO_BOOL, "bool"},
    {"DisableCompression",                  AO_BOOL, "bool"},
    {"EnableKernelRewind",                  AO_BOOL, "bool"},
    {"CompileANEProgramForDebugging",       AO_BOOL, "bool"},
```



```
{"DramTensorPriorityType",           AO_STRING, "string NoReuse,FirstFitReuse,BestFitReuse"},
{"DramAllocatorType",                AO_STRING, "string orderofcreation,costofreads,sizethenliverange"},
{"L2AllocatorType",                  AO_STRING, "string FirstFitReuse,BestFitReuse"},
{"L2CacheMode",                      AO_STRING, "string L2EnableResident,L2DisableResident,L2EnableCached
,
{"L3AllocatorType",                  AO_STRING, "string NoReuse,FirstFitReuse,BestFitReuse"},
{"DisableL2Wraparound",              AO_BOOL, "bool"},
{"DisableL2BankConflictOpt",         AO_BOOL, "bool"},
{"InputAndConstantCaching",          AO_BOOL, "bool"},
{"EnableSingleChannelElementwiseOpCopyRemoval", AO_BOOL, "bool"},
{"EnableAggressiveTensorCaching",    AO_BOOL, "bool"},
{"ProduceRelocatableObjects",        AO_BOOL, "bool"},
{"Signature",                        AO_STRING, "string Disabled,Ident"},
{"UseNewMachoMagicNumber",           AO_BOOL, "bool"},
{"MemCacheStrategy",                 AO_STRING, "string None,Simple"},
{"MemCacheSize",                     AO_BOOL, "number"},
{"SpatialSplit",                     AO_STRING, "string Disabled,Test,Memory,Auto"},
{"KernelRewind",                     AO_BOOL, "bool"},
{"ScanWeightsForCompression",        AO_BOOL, "bool"},
//Array ProcedureParams
{"ProcedureParams",                  AO_ARRAY, "array"},
```

Options in Module ANECompiler

H11ANEIn Kernel Driver

$$\begin{tikzcd}
\text{App} \arrow[rr, leftrightarrow] \arrow[d, leftrightarrow] & & \text{aned} \arrow[d, leftrightarrow] & \\
\text{H11ANEInDirectPathClient} \arrow[d, leftrightarrow] & & \text{H11ANEInUserClient} \arrow[d, leftrightarrow] & \\
\text{ANE\_ProgramSendRequest()} & & \text{ANE\_ProgramCreate()} & \\
\dots & & \dots & \\
& & & \text{ZinComputeProgramMake()}
\end{tikzcd}$$

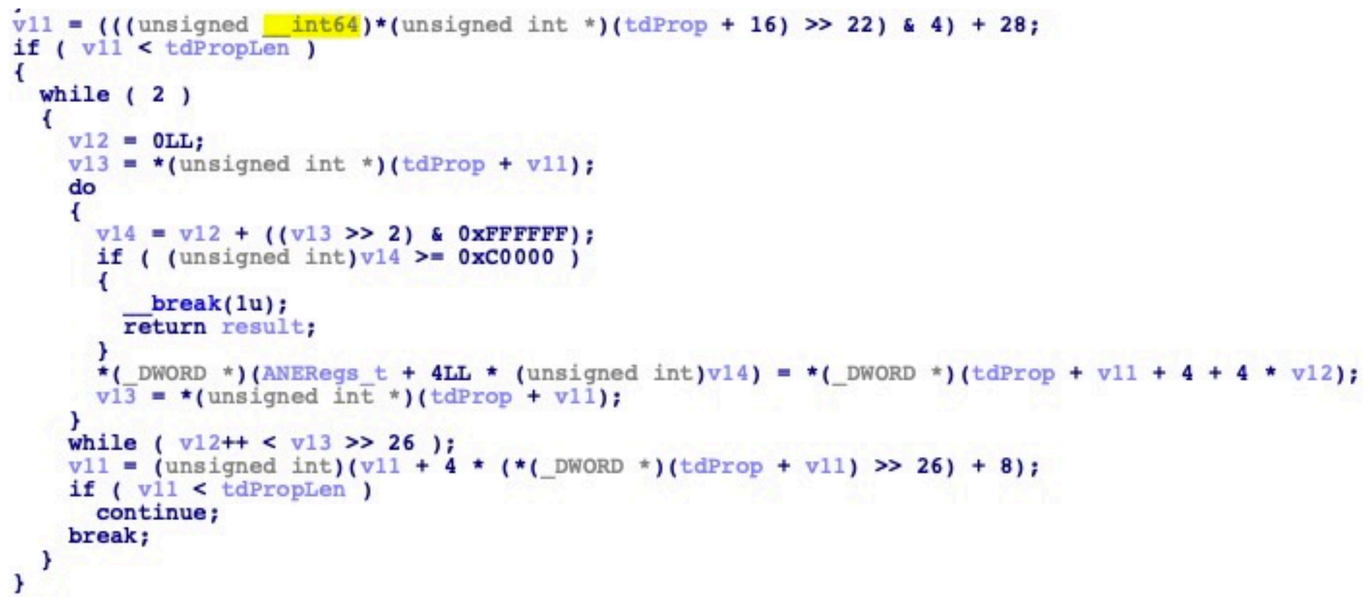
```
IOReturn H11ANEIn::aneCmdSend(void *cmd, uint32_t cmdSize, uint32_t *, uint32_t, bool, IOPhysicalAddress, uint32_t, bool, bool, bool)
```

ANE Firmware

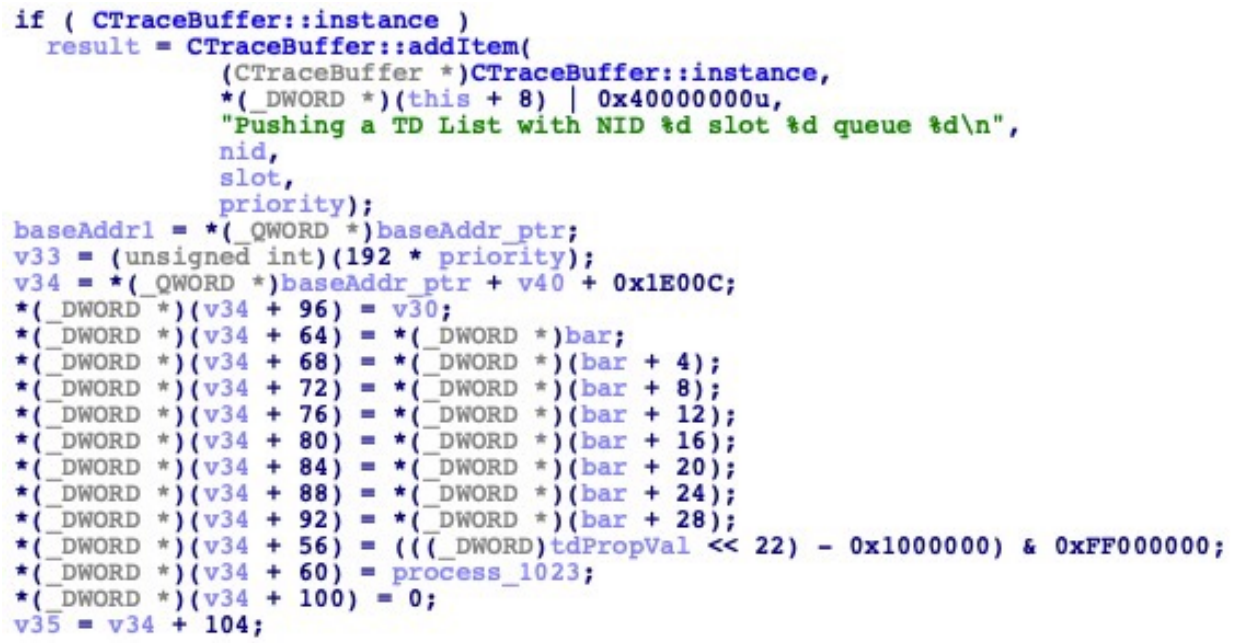
File Path : Firmware/ane/h11_ane_fw_quin.im4p

```
Load Program:
    CSneTMDrv::ParseTD(void const*,ulong,ANERegs_t *,ane_TD_HEADER_t *,bool)

Execute:
    CSneTMDrv::AddTDList(void const*,ulong,ulong,ulong,uint,uint *,uint,ulong volatile*,_rtk_timer_call *,bool)
```



```
v11 = (((unsigned __int64)*(unsigned int *)(tdProp + 16) >> 22) & 4) + 28;
if ( v11 < tdPropLen )
{
  while ( 2 )
  {
    v12 = 0LL;
    v13 = *(unsigned int *)(tdProp + v11);
    do
    {
      v14 = v12 + ((v13 >> 2) & 0xFFFFFF);
      if ( (unsigned int)v14 >= 0xC0000 )
      {
        __break(1u);
        return result;
      }
      *(_DWORD *)(ANERegs_t + 4LL * (unsigned int)v14) = *(_DWORD *)(tdProp + v11 + 4 + 4 * v12);
      v13 = *(unsigned int *)(tdProp + v11);
    }
    while ( v12++ < v13 >> 26 );
    v11 = (unsigned int)(v11 + 4 * (*(_DWORD *)(tdProp + v11) >> 26) + 8);
    if ( v11 < tdPropLen )
      continue;
    break;
  }
}
```



```
if ( CTraceBuffer::instance )
  result = CTraceBuffer::addItem(
             (CTraceBuffer *)CTraceBuffer::instance,
             *(_DWORD *)(this + 8) | 0x40000000u,
             "Pushing a TD List with NID %d slot %d queue %d\n",
             nid,
             slot,
             priority);
baseAddr1 = *(_QWORD *)baseAddr_ptr;
v33 = (unsigned int)(192 * priority);
v34 = *(_QWORD *)baseAddr_ptr + v40 + 0x1E00C;
*(_DWORD *)(v34 + 96) = v30;
*(_DWORD *)(v34 + 64) = *(_DWORD *)bar;
*(_DWORD *)(v34 + 68) = *(_DWORD *)(bar + 4);
*(_DWORD *)(v34 + 72) = *(_DWORD *)(bar + 8);
*(_DWORD *)(v34 + 76) = *(_DWORD *)(bar + 12);
*(_DWORD *)(v34 + 80) = *(_DWORD *)(bar + 16);
*(_DWORD *)(v34 + 84) = *(_DWORD *)(bar + 20);
*(_DWORD *)(v34 + 88) = *(_DWORD *)(bar + 24);
*(_DWORD *)(v34 + 92) = *(_DWORD *)(bar + 28);
*(_DWORD *)(v34 + 56) = (((_DWORD)tdPropVal << 22) - 0x1000000) & 0xFF000000;
*(_DWORD *)(v34 + 60) = process_1023;
*(_DWORD *)(v34 + 100) = 0;
v35 = v34 + 104;
```

My ANETools

CLI Tools For Apple Neural Engine

https://github.com/antgroup-arclab/ANETools.git

coremlc.sh

use xcode coremlc tool to compile .mlmodel file to mlmodelc folder.

MLModelCToANECompiler

convert neural network files under coremlc folder to net.plist(....) files for ANECompiler.

ANECompiler

compile net.plist and weights files to .hwx file.

ANEDisassembler

disassemble .hwx file, print all registers , values and bits.

ANEDisassembler

Thank George Hotz's https://github.com/geohot/tinygrad.git

Reference Function：

In ANECompiler :

```
ZinIrRegBitPrintOutDebug(unsigned int, ZinIrCodegenTd_v5 *, int, std::ostream &)
```

In ANE Firmware :

```
CSneTMDrv::ParseTD(void const*,ulong,ANERegs_t *,ane_TD_HEADER_t *,bool)
```

https://github.com/antgroup-arclab/ANETools/tree/main/ANEDisassembler

ANEDisassembler

```c
const struct aneTDItem aneRegsTileDMADst_v5[] = { //+268
    {0, 0,  1,  "aneRegs.TileDMADst.DMAConfig.En"},
    {0, 4,  1,  "aneRegs.TileDMADst.DMAConfig.CrH"},
    {0, 12, 4,  "aneRegs.TileDMADst.DMAConfig.CacheHint"},
    {0, 26, 1,  "aneRegs.TileDMADst.DMAConfig.L2BfrMode"},
    {0, 27, 1,  "aneRegs.TileDMADst.DMAConfig.BypassEOW"},
    {1, 6,  26, "aneRegs.TileDMADst.BaseAddr.Addr"},
    {2, 6,  24, "aneRegs.TileDMADst.Stride.Stride"},
    {3, 6,  24, "aneRegs.TileDMADst.PlaneStride.PlaneStride"},
    {4, 0,  2,  "aneRegs.TileDMADst.Fmt.FmtMode"},
    {4, 4,  2,  "aneRegs.TileDMADst.Fmt.Truncate"},
    {4, 8,  1,  "aneRegs.TileDMADst.Fmt.Shift"},
    {4, 12, 2,  "aneRegs.TileDMADst.Fmt.MemFmt"},
    {4, 16, 3,  "aneRegs.TileDMADst.Fmt.OffsetCh"},
    {4, 20, 1,  "aneRegs.TileDMADst.Fmt.ZeroPadLast"},
    {4, 21, 1,  "aneRegs.TileDMADst.Fmt.ZeroPadFirst"},
    {4, 22, 1,  "aneRegs.TileDMADst.Fmt.CmpVecFill"},
    {4, 24, 4,  "aneRegs.TileDMADst.Fmt.Interleave"},
    {4, 28, 4,  "aneRegs.TileDMADst.Fmt.CmpVec"},
    {5, 0,  32, "aneRegs.TileDMADst.Spare0.Spare"},
    {6, 0,  32, "aneRegs.TileDMADst.Spare1.Spare"},
};
const struct aneTDItemInt aneRegsTileDMADstInt_v5[] = {
    {&aneRegsTileDMADst_v5[0], 5},
    {&aneRegsTileDMADst_v5[5], 1},
    {&aneRegsTileDMADst_v5[6], 1},
    {&aneRegsTileDMADst_v5[7], 1},
    {&aneRegsTileDMADst_v5[8], 10},
    {&aneRegsTileDMADst_v5[18], 1},
    {&aneRegsTileDMADst_v5[19], 1},
};
```

```
idx 8 off 128 addr 0000000000 count 10 :
    0000000000  0x000d000e  851982
        -----------------000000000001110       14       0xe aneRegs.Common.InDim.Win
        -000000000001101----------------       13       0xd aneRegs.Common.InDim.Hin
    0x00000004  0x00000022  34
        ------------------------------10        2       0x2 aneRegs.Common.ChCfg.InFmt
        --------------------------10----        2       0x2 aneRegs.Common.ChCfg.OutFmt
    0x00000008  0x00000004  4
        ---------------00000000000000100        4       0x4 aneRegs.Common.Cin.Cin
    0x0000000c  0x00000001  1
        ---------------00000000000000001        1       0x1 aneRegs.Common.Cout.Cout
    0x00000010  0x00050003  327683
        -----------------000000000000011        3       0x3 aneRegs.Common.OutDim.Wout
        -000000000000101----------------        5       0x5 aneRegs.Common.OutDim.Hout
    0x00000014  0x6006024b  1611006539
        ---------------------------01011       11       0xb aneRegs.Common.ConvCfg.Kw
        ---------------------01001------        9       0x9 aneRegs.Common.ConvCfg.Kh
        -----------------000------------        0         0 aneRegs.Common.ConvCfg.OCGSize
        --------------10----------------        2       0x2 aneRegs.Common.ConvCfg.Sx
        ------------01------------------        1       0x1 aneRegs.Common.ConvCfg.Sy
        --------0000--------------------        0         0 aneRegs.Common.ConvCfg.Px
        ----0000------------------------        0         0 aneRegs.Common.ConvCfg.Py
        --10----------------------------        2       0x2 aneRegs.Common.ConvCfg.Ox
        01------------------------------        1       0x1 aneRegs.Common.ConvCfg.Oy
    0x00000018  0x00010001  65537
        -------------------0000000000001        1       0x1 aneRegs.Common.GroupConvCfg.NumGroups
        -----------------0--------------        0         0 aneRegs.Common.GroupConvCfg.UnicastEn
        ----------------0---------------        0         0 aneRegs.Common.GroupConvCfg.ElemMultMode
        0000000000000001----------------        1       0x1 aneRegs.Common.GroupConvCfg.UnicastCin
    0x0000001c  0x00000005  5
        -----------------000000000000101        5       0x5 aneRegs.Common.TileCfg.TileHeight
    0x00000020  0x04044400  67388416
        -----------------------------0--        0         0 aneRegs.Common.Cfg.SmallSourceMode
        ---------------------100--------        4       0x4 aneRegs.Common.Cfg.ShPref
        -----------------100------------        4       0x4 aneRegs.Common.Cfg.ShMin
        -------------100----------------        4       0x4 aneRegs.Common.Cfg.ShMax
        -----------00-------------------        0         0 aneRegs.Common.Cfg.ActiveNE
        ---------0----------------------        0         0 aneRegs.Common.Cfg.ContextSwitchIn
        -------0------------------------        0         0 aneRegs.Common.Cfg.ContextSwitchOut
        -----1--------------------------        1       0x1 aneRegs.Common.Cfg.AccDoubleBufEn
```

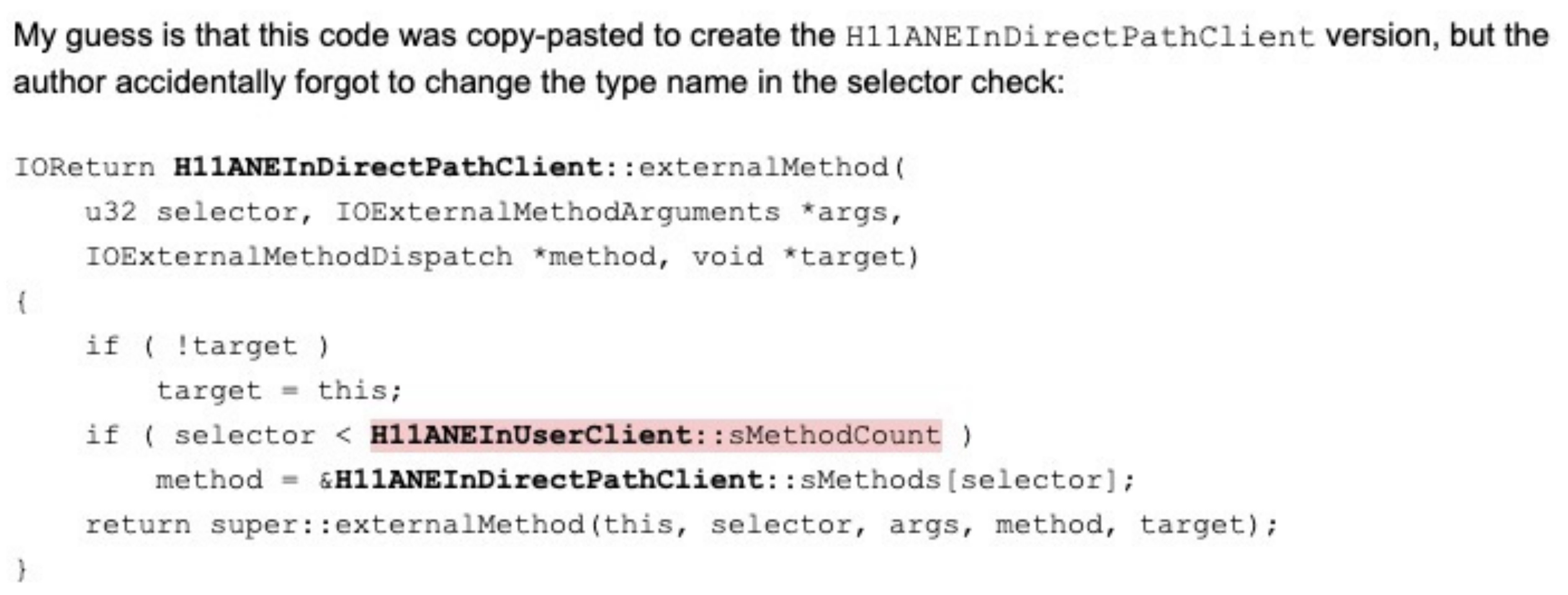
Attack Surfaces of ANE

1. Cloud to App, Cloud to Host
   .mlmodel file can be from internet and load dynamically
2. App to ANE daemon process
   App use XPC to talk with aned process
3. App to ANE Compiler process
   App provide files to ANECompilerService process
4. App to kernel ANE driver
   App process can access 3 H11ANEIn driver functions
5. ANE daemon process to kernel ANE driver
   aned process can access all H11ANEIn driver functions
6. kernel ANE driver and ANE firmware
   H11ANEIn driver use H11ANEIn::aneCmdSend to talk with Firmware

4. App to kernel ANE driver

https://googleprojectzero.blogspot.com/2020/11/oops-i-missed-it-again.html

Written by Brandon Azad, when working at Project Zero



My guess is that this code was copy-pasted to create the H11ANEInDirectPathClient version, but the author accidentally forgot to change the type name in the selector check:

```
IOReturn H11ANEInDirectPathClient::externalMethod(
    u32 selector, IOExternalMethodArguments *args,
    IOExternalMethodDispatch *method, void *target)
{
    if ( !target )
        target = this;
    if ( selector < H11ANEInUserClient::sMethodCount )
        method = &H11ANEInDirectPathClient::sMethods[selector];
    return super::externalMethod(this, selector, args, method, target);
}
```

1. Cloud to App, Cloud to Host

coremlcompiler compile <.mlmodel file> <output dir> --deployment-target 13.0 --platform ios

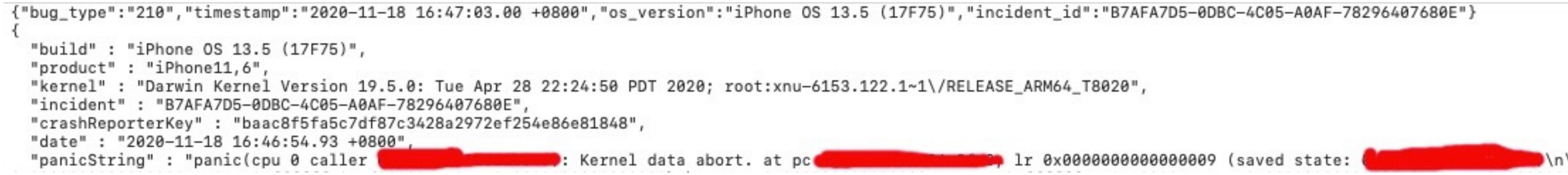
3 EXC_BAD_ACCESS crashes have been found and sent to Apple.

Sorry, bugs have not been fixed now , can not be public.

5. ANE daemon process to kernel ANE driver

I hook mmap() of aned process and modify one byte of model.hwx.

Got Kernel data abort



```
{"bug_type":"210","timestamp":"2020-11-18 16:47:03.00 +0800","os_version":"iPhone OS 13.5 (17F75)","incident_id":"B7AFA7D5-0DBC-4C05-A0AF-78296407680E"}
{
  "build" : "iPhone OS 13.5 (17F75)",
  "product" : "iPhone11,6",
  "kernel" : "Darwin Kernel Version 19.5.0: Tue Apr 28 22:24:50 PDT 2020; root:xnu-6153.122.1~1\/RELEASE_ARM64_T8020",
  "incident" : "B7AFA7D5-0DBC-4C05-A0AF-78296407680E",
  "crashReporterKey" : "baac8f5fa5c7df87c3428a2972ef254e86e81848",
  "date" : "2020-11-18 16:46:54.93 +0800",
  "panicString" : "panic(cpu 0 caller [redacted]: Kernel data abort. at pc [redacted] lr 0x0000000000000009 (saved state: [redacted]\n\
```

Bug Prediction

May locate in:

1. Passes of CoreMLCompiler or ANECompiler.
2. File format parser in kernel driver, daemon process.
3. Complex interaction between aned , kernel driver and ane firmware.

Thank you

蚂蚁集团安全响应中心
ANT GROUP SECURITY RESPONSE CENTER

蚂蚁集团 ANT GROUP | 蚂蚁安全实验室 ANT SECURITY LAB