

Εργασία στο TinyOS

ΜΕΡΟΣ ΔΕΥΤΕΡΟ

Edward Epure – 2013030015
Ευτυχία Σεισάκη - 2013030031

Στο 2^ο μέρος της εργασίας ζητείται η υλοποίηση δύο επεκτάσεων του 1^{ου} προγράμματος. Αρχικά πρέπει να επεκτείνουμε τη λειτουργικότητα του **TAG** σε περισσότερες συναθροιστικές συναρτήσεις και στη συνέχεια, να υλοποιήσουμε τη λειτουργικότητα του **TINA**.

Ακολουθεί περιγραφή της υλοποίησης με την ακόλουθη δομή:

- A) Αλλαγές/Προσθήκες κώδικα
- B) Επεξηγήσεις σημείων του κώδικα
- C) Εξήγηση περιεχομένου μηνυμάτων
- D) Παραδείγματα εκτέλεσης προγράμματος
- E) Σχόλια

Αλλαγές/Προσθήκες κώδικα

- **Αλλαγές** στις υλοποιημένες συναρτήσεις:

- *RoutingMsgTimer.fired()* :
- *epochTimer.Fired()*:
- *ReceiveNotifyTask()*:

Επεξηγήσεις σημείων του κώδικα

A) Η ρίζα **επιλέγει** την λειτουργικότητα που θα εκτελεστεί (TAG ή TINA) και τις **συναθροιστικές συναρτήσεις**. Κωδικοποιεί τις παραπάνω πληροφορίες σε μία μεταβλητή (*msg_type*) των 8 bits.

Η **κωδικοποίηση** γίνεται με τον εξής τρόπο:

e.g. *msg_type* : **153**

aggr1	aggr2	type
1:MIN	1:MIN	1:(8bits)
2:MAX	2:MAX	2:(16bits)
	3:COUNT	3:(24bits)
	4:SUM_COUNT	4:(32bits)
	5:SUM	5:(8bits-TiNA)
		6:(16bits-TiNA)
		7:(56bits-VAR)
		8:(64bits-VAR)

Ο **συνδυασμός** SUM,COUNT λαμβάνεται ως **μία συνάρτηση**, επειδή εμφανίζεται στις AVG,VAR.

Κανένας συνδυασμός **δεν ξεπερνάει** την μέγιστη **τιμή** που μπορεί να αναπαραστήσει η μεταβλητή (255).

Εάν **στέλνεται μία** μόνο συνάρτηση το *aggr1* γίνεται 0.

Το **type** περιέχει πληροφορία για το **μέγεθος** του **πακέτου** αποστολής δεδομένων και για την **επιλογή** TAG/Tina.

Η αποκωδικοποίηση του msg_type στα παιδιά γίνεται ως εξής:

1 ψηφίο: msg_type / 100

2 ψηφίο: (msg_type % 100) / 10

3 ψηφίο: msg_type % 10

Παράδειγμα κωδικοποίησης για την παραπάνω περίπτωση MIN+SUM:

```
484 if(aggr1 == SUM || aggr2 == SUM)
485 {
486     if(aggr1 == MIN || aggr2 == MIN)
487         msg_type = MIN1+SUM2+TYPE_24;
488     else if(aggr1 == MAX || aggr2 == MAX)
489         msg_type = MAX1+SUM2+TYPE_24;
490     else if(aggr1 == VAR || aggr2 == VAR)
491         msg_type = SC2+TYPE_56;
492     else
493         msg_type = SC2+TYPE_24;
494 }
```

msg_type = 1x100 + 5x10 + 3 = 153

Β)Επιλογή **μεγέθους πακέτου** αποστολής:

```
If (TAG)
    Enqueue packet with length: packet_type
else if (TINA)
    if( val > TCT and size of val == 1 byte)
        Enqueue packet with length: 1 byte
    else if( val > TCT and size of val == 2 bytes)
        Enqueue packet with length: 2 bytes
    else if( val < TCT )
        //do nothing
```

Ο συγκεκριμένος **ψευδοκώδικας** περιγράφει την λειτουργικότητα του σημείου 345-390 (**EpochTimer.Fired()**) στον κώδικα.

Με αυτόν τον τρόπο γίνεται **έλεγχος** πριν την προσθήκη ενός μηνύματος στην **ουρά αναμονής**.

Εξήγηση περιεχομένου μηνυμάτων

```
57  typedef nx_struct RoutingMsg
58  {
59      nx_uint8_t depth;
60      nx_uint8_t aggr;
61  } RoutingMsg;
62
63  typedef nx_struct Msg_64
64  {
65      nx_uint8_t var8;
66      nx_uint8_t var8_2;
67      nx_uint16_t var16;
68      nx_uint32_t var32;
69  } Msg_64;
```

Το **RoutingMsg** στέλνεται πάντα **ολόκληρο** (16 bits).

8 bits για το **βάθος** του δέντρου

8 bits για τις **συναθροιστικές** συναρτήσεις και τη λειτουργικότητα του προγράμματος (TAG/TiNA).

Το **Msg_64** περιέχει **όλες** τις δυνατές μεταβλητές και **επιλέγεται** κάθε φορά ο **συνδυασμός** που θα **σταλεί**.

Συνδυασμοί μεταβλητών σε ένα πακέτο δεδομένων:

var8	(8 bits)	: MIN/MAX/COUNT
var8 + var8_2	(16 bits)	: συνδυασμοί (MIN/MAX/COUNT)
var16	(16 bits)	: SUM
var32	(32 bits)	: SUM(X^2)
var8 + var16	(24 bits)	: συνδυασμός SUM με (MIN/MAX/COUNT) ή AVG
var8 + var16 + var32	(56 bits)	: VAR σε συνδυασμό με (SUM/COUNT/AVG)
var8 + var8_2 + var 16 + var32	(64 bits)	: συναδυασμός VAR με (MIN/MAX)

Παραδείγματα εκτέλεσης προγράμματος

Παράδειγμα εκτέλεσης TiNA με 9 κόμβους

```
0:0:11.000000010 DEBUG (0): TINA
0:1:11.000000010 DEBUG (0):
_____ EPOCH__0
0:1:11.000000010 DEBUG (0): MAX: 48
0:2:11.000000010 DEBUG (0):
_____ EPOCH__1
0:2:11.000000010 DEBUG (0): MAX: 48
0:3:11.000000010 DEBUG (0):
_____ EPOCH__2
0:3:11.000000010 DEBUG (0): MAX: 48
0:4:11.000000010 DEBUG (0):
_____ EPOCH__3
0:4:11.000000010 DEBUG (0): MAX: 49
0:5:11.000000010 DEBUG (0):
_____ EPOCH__4
0:5:11.000000010 DEBUG (0): MAX: 39
.
.
.
```

Παράδειγμα εκτέλεσης TAG με 9 κόμβους για μία συνάρτηση

```
0:0:11.000000010 DEBUG (0): TAG
0:1:11.000000010 DEBUG (0):
_____ EPOCH__0
0:1:11.000000010 DEBUG (0): COUNT: 9
0:2:11.000000010 DEBUG (0):
_____ EPOCH__1
0:2:11.000000010 DEBUG (0): COUNT: 9
0:3:11.000000010 DEBUG (0):
_____ EPOCH__2
0:3:11.000000010 DEBUG (0): COUNT: 9
0:4:11.000000010 DEBUG (0):
_____ EPOCH__3
0:4:11.000000010 DEBUG (0): COUNT: 9
0:5:11.000000010 DEBUG (0):
_____ EPOCH__4
0:5:11.000000010 DEBUG (0): COUNT: 9
.
.
.
```

Παράδειγμα εκτέλεσης TAG με 9 κόμβους για δύο συναρτήσεις

```
0:0:11.000000010 DEBUG (0): TAG
0:1:11.000000010 DEBUG (0):
_____ EPOCH__0
0:1:11.000000010 DEBUG (0): MAX: 50
0:1:11.000000010 DEBUG (0): SUM: 220
0:2:11.000000010 DEBUG (0):
_____ EPOCH__1
0:2:11.000000010 DEBUG (0): MAX: 44
0:2:11.000000010 DEBUG (0): SUM: 235
0:3:11.000000010 DEBUG (0):
_____ EPOCH__2
0:3:11.000000010 DEBUG (0): MAX: 34
0:3:11.000000010 DEBUG (0): SUM: 149
0:4:11.000000010 DEBUG (0):
_____ EPOCH__3
0:4:11.000000010 DEBUG (0): MAX: 47
0:4:11.000000010 DEBUG (0): SUM: 267
0:5:11.000000010 DEBUG (0):
_____ EPOCH__4
0:5:11.000000010 DEBUG (0): MAX: 44
0:5:11.000000010 DEBUG (0): SUM: 233
```

.
.
.

Σχόλια/Διευκρινίσεις

Εκτέλεση βοηθητικού προγράμματος: ***python topo.py 3 1.5*** για $D=3$ και $C=1.5$.

Εκτέλεση κυρίου προγράμματος: ***python mySimulation.py***

Πρέπει να **αλλάξει** ο αριθμός των **nodes** για μεγαλύτερο topology.

Για μηνύματα debugging πρέπει να γίνουν **uncomment** τα αντίστοιχα κανάλια.

Το πρόγραμμα εκτελείται **επιτυχώς** για μέγεθος **έως 8x8**

Στον φάκελο περιέχονται 3 αρχεία εξόδου:

TiNA με μέγεθος 3x3

TAG για μία μεταβλητή και μέγεθος 3x3

TAG για δύο μεταβλητές και μέγεθος 3x3

Η υλοποίηση της εργασίας έγινε από κοινού από τους δυο φοιτητές.

Η εργασία παραδόθηκε **εντός** της αρχικής προθεσμίας (14/12/2018).

Επιλογή εξέτασης: **πριν** τις γιορτές (17/12/2018)